

UNIVERSIDADE FEDERAL DO PARANÁ

VINICIUS STRUGINSKI PEREIRA

CORE INC.: UMA SOLUÇÃO DE ALGORITMOS ANIMADOS COM COGNIÇÃO
SITUADA

CURITIBA PR

2022

VINICIUS STRUGINSKI PEREIRA

CORE INC.: UMA SOLUÇÃO DE ALGORITMOS ANIMADOS COM COGNIÇÃO
SITUADA

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre em Informática no Programa de Pós-Graduação em Informática, Setor de Ciências Exatas, da Universidade Federal do Paraná.

Área de concentração: *Ciência da Computação*.

Orientador: Marcos Castilho.

CURITIBA PR

2022

DADOS INTERNACIONAIS DE CATALOGAÇÃO NA PUBLICAÇÃO (CIP)
UNIVERSIDADE FEDERAL DO PARANÁ
SISTEMA DE BIBLIOTECAS – BIBLIOTECA DE CIÊNCIA E TECNOLOGIA

Pereira, Vinicius Struginski

Core Inc. : uma solução de algoritmos animados com cognição situada /
Vinicius Struginski Pereira. – Curitiba, 2022.
1 recurso on-line : PDF.

Dissertação (Mestrado) - Universidade Federal do Paraná, Setor de
Ciências Exatas, Programa de Pós-Graduação em Informática.

Orientador: Marcos Alexandre Castilho

1. Informática – Estudo e ensino. 2. Programação (Computadores). 3.
Algoritmos. 4. Jogos digitais. 5. Animação por computador. 6. Cognição. I.
Universidade Federal do Paraná. II. Programa de Pós-Graduação em
Informática. III. Castilho, Marcos Alexandre. IV. Título.

Bibliotecário: Elias Barbosa da Silva CRB-9/1894

TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação INFORMÁTICA da Universidade Federal do Paraná foram convocados para realizar a arguição da Dissertação de Mestrado de **VINICIUS STRUGINSKI PEREIRA** intitulada: **CORE INC.: UMA SOLUÇÃO DE ALGORITMOS ANIMADOS COM COGNIÇÃO SITUADA**, que após terem inquirido o aluno e realizada a avaliação do trabalho, são de parecer pela sua APROVAÇÃO no rito de defesa.

A outorga do título de mestre está sujeita à homologação pelo colegiado, ao atendimento de todas as indicações e correções solicitadas pela banca e ao pleno atendimento das demandas regimentais do Programa de Pós-Graduação.

CURITIBA, 09 de Novembro de 2022.

Assinatura Eletrônica

16/11/2022 13:38:44.0

MARCOS ALEXANDRE CASTILHO

Presidente da Banca Examinadora

Assinatura Eletrônica

10/11/2022 10:43:07.0

LETICIA MARA PERES

Avaliador Interno (UNIVERSIDADE FEDERAL DO PARANÁ)

Assinatura Eletrônica

09/11/2022 14:50:49.0

ALEXANDER ROBERT KUTZKE

Avaliador Externo (UNIVERSIDADE FEDERAL DO PARANÁ - SEPT)

À memória de meus familiares e amigos que nos deixaram.

Aos meus pais Iremar e Roseli e irmã Gabriele, pelo apoio incondicional durante toda minha trajetória e especialmente durante a escrita desta dissertação.

AGRADECIMENTOS

Em primeiro lugar, agradeço a minha família e em especial meus pais e irmã que me apoiaram incondicionalmente por toda minha trajetória até a conclusão desta dissertação.

Aos professores que foram responsáveis pela minha formação até aqui.

Ao meu orientador Marcos Castilho, pela paciência praticamente infinita e conselhos essenciais para a conclusão deste trabalho.

Aos amigos e colegas da comunidade Nautilus, agradeço em especial a Bruno Tessaro, Henrique Antero, Lucas Zavadil, Ricardo Regis e Nelson por criarem esta comunidade e aos vários amigos Alex Alves, Anderson Rodrigo, Andre Moraes, Arthur Ramos, Breno Martins, Breno "Conta", Bruno Quadros, Eduardo Keneipp, Jonathan Carvalho, Luiz Cavalcanti, Matheus Gubler, Túlio Lobo e Vil e vários outros pelas conversas diárias ao longo dos últimos anos que me ajudaram a seguir em frente.

Aos meus amigos Alexandre da Rosa, Gabriel Vaz Igarashi, Rafael Hitoshi de Oliveira e Romualdo Viana Filho por estarem ao meu lado durante minha vida e trajetória acadêmica para que eu pudesse chegar aqui hoje.

Aos irmãos que escolhi, Iury Teixeira de Carvalho, Jonas Pereira de Cristo, Lucas Balduino, Rafael Piazzentin Freire Gonçalves e Thomas Gabriel Balduino Reckelberg por me apoiarem mais do que qualquer um poderia pedir e terem a paciência de me aturar durante todos estes anos.

*"Segunda à direita e depois direto
até amanhã de manhã".
(J.M. Barrie - Peter Pan)*

RESUMO

Esta dissertação apresenta um estudo embasado na teoria da cognição situada e elementos de jogos digitais para o auxílio no ensino introdutório de programação de computadores, especificamente no momento em que o modelo de *von Neumann* é apresentado. O estudo é validado pela apresentação de uma ferramenta de algoritmos animados denominada Core Inc. desenvolvida segundo as teorias citadas, apresentando de maneira lúdica versões antropomorfizadas das regiões essenciais de uma unidade central de processamento segundo a arquitetura de *von Neumann*. A validação do estudo foi realizada com professores de diferentes áreas de estudo através de uma metodologia qualitativa de entrevistas semiestruturadas fenomenológicas com o suporte adicional de um questionário baseado do modelo de aceitação de tecnologia. Os resultados demonstraram que os professores consideraram os conceitos e a proposta desenvolvida adequados.

Palavras-chave: Educação 1. Algoritmos 2. Animação 3.

ABSTRACT

This dissertation presents a study grounded in the situated cognition theory and elements of videogames to improve the teach-learning process on the student's first contact with programming concepts, specifically when the von Neumann architecture is presented. The study is validated by an apresentation of an animated algorithm framework named Core Inc., developed accordingly to the cited theories, showing anthropomorphized versions of the main regions from the central process unit according to von Neumann architecture. The study's validation was done by teachers and researchers from different study areas through a qualitative methodology of semi-structured phenomenological interviews with additional support from a questionnaire grounded on the technology acceptance model. The results demonstrated that the teachers considered the concepts and the developed proposal adequate.

Keywords: Education 1.Algorithm 2. Animation 3.

LISTA DE FIGURAS

2.1	Arquitetura de von Neumann.(Tanembaum, 2013)	21
2.2	Taxonomia de Software Visualization. (Price et al., 1993).	23
2.3	Tela do software BALSA. (Brown e Sedgewick, 1985).	26
2.4	Tela do software Tango. (Stasko, 1990a)	27
2.5	Tela do software Zeus. (Brown, 1991)	28
2.6	Tela do software ANIM. (Bentley e Kernighan, 1991)	29
2.7	Tela do software desenvolvido por Carlson.(Carlson et al., 1996)	30
2.8	Tela do software SICAS.(Gomes et al., 2004)	31
2.9	Tela do software JPie.(Goldman, 2004).	32
2.10	Tela do software Greenfoot.(Kölling, 2008)	32
2.11	Tela do software Jype.(Helminen e Malmi, 2010)	33
2.12	Tela do software AIDA. O Autor	34
2.13	Tela do software Scratch. (Maloney et al., 2010).	34
2.14	Tela do software de Farrugia.(Farrugia e Quigley, 2011).	35
2.15	Tela do software JAVENGA.(Baloukas, 2012)	36
2.16	Tela do software CSTutor.(Buchanan e Laviola, 2014)	36
2.17	Tela do software TBC-AED.(dos Santos e Costa, 2005)	37
2.18	Efetividade das Soluções Catalogadas. (Hundhausen et al., 2002).	39
2.19	Efetividade dos Softwares por Teorias. (Hundhausen et al., 2002).	40
2.20	Efetividade dos Softwares por Métodos de Avaliação. (Hundhausen et al., 2002).	41
2.21	Ikenfell.(Games, 2020)	43
2.22	Modelo de Aceitação de Tecnologia (Saleh, 2004)	44
2.23	Extensão do Modelo de Aceitação de Tecnologia (Saleh, 2004)	44
3.1	Logo Core Inc. O Autor	48
3.2	Ilustração Empresa Core Inc. O Autor	48
3.3	Personagem Lucy. O Autor	49
3.4	Personagem Úrsula. O Autor	50
3.5	Personagem M3M0. O Autor	51
3.6	Personagem Bus. O Autor	52
3.7	Triângulo da Esquisitice (Rogers, 2016)	54
3.8	Cenário Core Inc.	55
3.9	Relações de Objetos com Contexto	56
3.10	Tipos de Pacotes de Entrega (Variáveis)	56

3.11	Diagrama de Contexto	59
3.12	Diagrama de Fluxo de Dados	60
3.13	Protótipo de Baixa Fidelidade	60
3.14	Sprite Sheet Lucy	65
3.15	Rotas de Animação	66
3.16	Animação Linha 1.	69
3.17	Animação Linha 2.	70
3.18	Animação Linha 4 - Parte 1.	71
3.19	Animação Linha 4 - Parte 2.	71
3.20	Animação Linha 5 - Parte 1.	72
3.21	Animação Linha 5 - Parte 2.	72
3.22	Animação Linha 7.	73
4.1	Funcionalidade de Códigos Exemplo	76
4.2	Uso de ferramentas de visualização de software	107
4.3	Uso de ferramentas de algoritmos animados	107
4.4	Interesse em algoritmos animados	108
4.5	Frequência de uso algoritmos animados e visualização de software	108
4.6	Número de softwares utilizados.	109
4.7	Temas das soluções utilizadas.	109
4.8	Satisfação das experiências anteriores	110
4.9	Respostas “Em geral, é fácil usar o Core Inc.”	110
4.10	Respostas “É fácil lembrar como realizar ações no Core Inc.”	111
4.11	Respostas “Não é necessário treinamento para utilizar o Core Inc.”	111
4.12	Respostas “Aprender a usar o Core Inc. é fácil.”	112
4.13	Respostas “Utilizar o Core Inc. é satisfatório.”	112
4.14	Respostas “Consigo utilizar o Core Inc. sem o auxílio de um instrutor.”	113
4.15	Respostas “A interface do Core Inc. é de fácil entendimento.”	114
4.16	Respostas “Utilizar o Core Inc. pode melhorar o processo de ensino durante a aula.”	114
4.17	Respostas “Usar o Core Inc. auxiliaria no ensino de algoritmos e programação em sala de aula.”	115
4.18	Respostas “O Core Inc. auxiliaria durante as aulas.”	115
4.19	Respostas “Consigo atingir meus objetivos como professor mais facilmente com o auxílio do Core Inc.”	116
4.20	Respostas “Utilizar o Core Inc. pode melhorar a qualidade de meu trabalho.”	116
4.21	Respostas “Utilizar o Core Inc. pode tornar as aulas mais produtivas.”	117
4.22	Respostas “Utilizar o Core Inc. pode melhorar a eficiência do tempo gasto em sala de aula.”	117

4.23	Respostas “O Core Inc. possui um visual engajante.”	118
4.24	Respostas “Os personagens do Core Inc. são bonitos.”	118
4.25	Respostas “O cenário das animações de Core Inc. é bonito.”	119
4.26	Respostas “O cenário das animações de Core Inc. é de fácil entendimento.”	119
4.27	Respostas “As animações do Core Inc. são engajante.”	120
4.28	Respostas “As animações do Core Inc. são de fácil entendimento.”	120
4.29	Respostas “O tutorial do Core Inc. é de fácil acesso.”	121
4.30	Respostas “As instruções do tutorial são de fácil compreensão.”.	121
4.31	Respostas “Recomendo a utilização do Core Inc.”	122
4.32	Respostas “Prefiro utilizar o Core Inc. em sala de aula para auxiliar no ensino de algoritmos e programação.”.	122
4.33	Respostas “Prefiro utilizar o Core Inc. em sala de aula para auxiliar no ensino de algoritmos e programação.”.	123
4.34	Respostas “Estou motivado a utilizar o Core Inc.”	123
B.1	carta de Apresentação.	139
D.1	Questionário Professores	142
D.2	Questionário em Comum Parte 1	143
D.3	Questionário em Comum Parte 2	144
D.4	Questionário em Comum Parte 3	145
D.5	Questionário Método TAM Parte 2	145
D.6	Questionário Método TAM Parte 3	146
D.7	Questionário Método TAM Parte 4	147
D.8	Questionário Método TAM Parte 5	148
D.9	Questionário Método TAM Parte 6	149
D.10	Questionário Método TAM Parte 7	150
G.1	Processo de Manipulação do Código	184

LISTA DE TABELAS

3.1	Codificar	57
3.2	Escolher Animação	58
3.3	Pausar Animação	58
3.4	Elementos de descrição	58
3.5	Animação de Apresentação	59
3.6	Animação Instrução program	62
3.7	Animação Instrução if then else	62
3.8	Animação Declaração.	63
3.9	Animação Entrada de Dados	63
3.10	Animação Saída de Dados	64
3.11	Animação Operadores Lógicos e Aritméticos	64
4.1	Frases Entrevista Professor A.	79
4.2	Frases Entrevista Professor B.	87
4.3	Frases Entrevista Professora C	93
4.4	Frases Entrevista Professor D.	100
4.5	Sugestões de Modificação Necessárias	127
4.5	Sugestões de Modificação Necessárias	128
4.6	Sugestões Adicionais	128

LISTA DE ACRÔNIMOS

DINF	Departamento de Informática
PPGINF	Programa de Pós-Graduação em Informática
UFPR	Universidade Federal do Paraná

SUMÁRIO

1	INTRODUÇÃO	15
1.1	PROBLEMA	16
1.2	OBJETIVOS	16
1.3	JUSTIFICATIVA	17
1.4	ORGANIZAÇÃO	17
2	REFERENCIAL TEÓRICO	19
2.1	COGNIÇÃO SITUADA	19
2.2	ARQUITETURA DE VON NEUMANN.	21
2.3	ALGORITMO ANIMADO.	22
2.4	TRABALHOS RELACIONADOS A ALGORITMOS ANIMADOS	25
2.5	TRABALHOS DE VISUALIZAÇÃO DE SOFTWARE	31
2.6	EFETIVIDADE DOS ALGORITMOS ANIMADOS NO ENSINO	38
2.7	PIXEL ART.	42
2.8	MODELO DE ACEITAÇÃO DE TECNOLOGIA - TAM	43
2.9	CONSIDERAÇÕES	45
3	DESENVOLVIMENTO DA SOLUÇÃO DE ALGORITMO ANIMADO. . .	47
3.1	CORE INC.	47
3.2	CONSTRUÇÃO DA CONTEXTUALIZAÇÃO	52
3.3	ANÁLISE ESSENCIAL	57
3.4	ESTRUTURA DA ANIMAÇÃO DE ALGORITMOS	60
3.5	DESENVOLVIMENTO DA ANIMAÇÃO	65
3.6	DEMONSTRAÇÃO DE FUNCIONAMENTO	68
3.7	CONSIDERAÇÕES	73
4	EXPERIMENTOS E RESULTADOS	75
4.1	JUSTIFICATIVA DOS EXPERIMENTOS.	75
4.2	DEFINIÇÃO DO EXPERIMENTO PRINCIPAL	76
4.3	DEFINIÇÃO DO EXPERIMENTO COMPLEMENTAR	78
4.4	RESULTADOS EXPERIMENTO PRINCIPAL	79
4.4.1	Análise Temática Entrevista Professor A	79
4.4.2	Análise Temática Entrevista Professor B	87
4.4.3	Análise Temática Entrevista Professora C	93
4.4.4	Análise Temática Entrevista Professor D	100
4.5	RESULTADOS EXPERIMENTO COMPLEMENTAR	106
4.6	CONSIDERAÇÕES	121

5	CONCLUSÃO	129
	REFERÊNCIAS	132
	APÊNDICE A – PARÂMETROS DE ENTRADA DE COMANDOS CORE INC..	137
	APÊNDICE B – CARTA DE APRESENTAÇÃO	139
	APÊNDICE C – CÓDIGOS DE EXEMPLO CORE INC.	140
	APÊNDICE D – QUESTIONÁRIO EXPERIMENTO COMPLEMENTAR	142
	APÊNDICE E – ROTEIRO ENTREVISTA QUALITATIVA	151
	APÊNDICE F – TRANSCRIÇÃO ENTREVISTAS EXPERIMENTO PRINCIPAL	152
F.1	TRANSCRIÇÃO ENTREVISTA PROFESSOR A.	152
F.2	TRANSCRIÇÃO ENTREVISTA PROFESSOR B.	161
F.3	TRANSCRIÇÃO ENTREVISTA PROFESSORA C.	170
F.4	TRANSCRIÇÃO ENTREVISTA PROFESSOR D.	176
	APÊNDICE G – DEMONSTRAÇÃO DETALHADA DA IMPLEMENTAÇÃO DO CORE INC..	183
G.1	MANIPULAÇÃO DE CÓDIGO-FONTE	183
G.2	MÓDULO DE ANIMAÇÃO	188

1 INTRODUÇÃO

A compreensão do funcionamento de um algoritmo e sua ação em tempo real no computador podem ser tarefas complexas e abstratas para alunos recém introduzidos aos conceitos de programação (Brown e Sedgewick, 1985). Diversas pesquisas no campo evidenciam como tal dificuldade no aprendizado cria uma deficiência no processo, gerando a reprovação de uma quantidade considerável de alunos nas disciplinas introdutórias do tema (Campos, 2010). A quantidade de conceitos a serem absorvidos, junto de uma linguagem de programação com regras de utilização e escrita totalmente novas cria ainda mais barreiras.

Um dos muitos conceitos a serem absorvidos durante o processo de aprendizagem de algoritmos e programação é o funcionamento do computador e como ele executa as instruções que os alunos começam a aprender nesta fase inicial. A arquitetura de *von Neumann* é um modelo de funcionamento dos computadores modernos utilizado para explicar e um dos pilares desta etapa de aprendizado. Este conteúdo é abordado pelo material didático (Castilho et al., 2020) utilizado pelos alunos e público-alvo desta dissertação no curso de ciência da computação da Universidade Federal do Paraná.

Meios de solucionar esta deficiência através de métodos de ensino inovadores são uma preocupação recorrente dos pesquisadores, sendo que somente no Brasil 73 estudos relevantes para a área foram localizados no período de 2009 até 2015 (da Silva et al., 2015). Dentre estes estudos diversas abordagens estão presentes, como software de apoio, jogos educativos, soluções com robótica e novas metodologias pedagógicas. Neste meio, destaca-se a quantidade de soluções do tipo jogos, totalizando 14 durante este período.

A motivação por trás de tantas soluções do mesmo gênero se sustenta em diversos benefícios que os jogos podem trazer no processo de ensino-aprendizagem. Um destes é a capacidade dos jogos de tomarem para si um conteúdo e dar ao mesmo um senso de propósito, que torna o conhecimento relevante neste contexto (Eck, 2006). Essa relação entre conhecimento e contexto é abordada por pesquisadores como cognição situada (Brown et al., 1989) e difundiu-se amplamente ao longo dos anos.

Estabelecer uma relação entre conteúdo e um contexto torna-se ainda mais importante para um assunto tão abstrato quanto programação e algoritmos, principalmente no seu estágio inicial, quando os alunos ainda estão em seu primeiro contato com tantos conceitos novos. Contudo, jogos nem sempre se adequam a todas as situações. Isto acontece quando o professor deve apresentar junto destes conceitos uma linguagem de programação com ferramentas capazes de resolver os problemas de lógica e programação propostos ao decorrer do curso. Portanto, soluções que permitam a codificação através de linguagens de programação tradicionais são importantes.

Então buscou-se uma abordagem que permita a codificação por parte dos alunos e fosse capaz de diminuir a abstração necessária em algoritmos. Desta forma identificou-se soluções capazes de entregar representações visuais do que acontece no programa, estas chamadas de algoritmos animados (Brown e Sedgewick, 1985) (Brown, 1991) (Stasko, 1990b) (Garcia et al., 1997) (Bentley e Kernighan, 1991). Contudo, não só a representação visual é importante, mas também a forma com que o conteúdo é exibido ao aluno, tal como Demetrescu e Finocchi (Demetrescu e Finocchi, 2001) já se preocupavam com a fluidez das animações que seriam exibidas.

Ao trazer as soluções de algoritmos animados para o âmbito do contexto do dia a dia do aluno, proposta pela teoria da cognição situada, as soluções apresentadas até o momento

demonstram-se deficientes, pois não entregam um contexto para as representações visuais. As soluções atuais usam representações geométricas dos dados e estruturas presentes no programa ou representações visuais através de desenhos e animações sem uma justificativa contextual no ambiente do aluno. Por mais que os resultados de tais soluções já tenham se demonstrado benéficas, o potencial de unir-se tais soluções com a contextualização que a representação visual típica dos jogos possui é promissora para software de algoritmos animados.

Ao longo desta dissertação é apresentado os fundamentos teóricos que embasam este trabalho, além do processo desenvolvimento do Core Inc., uma solução de algoritmos animados com elementos visuais de jogos e embasada na teoria da cognição situada. Na sequência serão apresentados os experimentos realizados através de questionários e entrevistas qualitativas, onde a solução de algoritmos animados Core Inc. é avaliada em busca de verificar o potencial de aceitação de alunos e professores sobre a ferramenta.

1.1 PROBLEMA

O tema deste trabalho aborda a etapa de ensino de algoritmos e programação em seu estágio inicial. Portanto, ao sintetizar o problema em uma única pergunta, temos: Como desenvolver algoritmos animados de forma contextualizada para melhorar o processo de ensino-aprendizagem de disciplinas introdutórias de algoritmos e programação particularmente durante o momento do aprendizado da arquitetura de *von Neumann*?

Entende-se que os fatores que influenciam o processo de ensino-aprendizagem de algoritmos são vários (de L. Junior et al., 2015) e a abordagem proposta, por mais eficiente que seja, não será capaz de sanar todos os desafios relacionados ao tema, visto que diferentes estudos apontam uma diversidade de obstáculos na taxa de aprendizado dos alunos, como por exemplo, a pesquisa de Giraffa e Mora (Giraffa e da C. Mora, 2013) relatando que alunos com deficiência em conteúdos matemáticos também possuem influência direta no problema bem como pesquisas mais recentes que acabam por contrapor tal afirmação (Prat et al., 2020), levantando problemas de raciocínio fluído e memorização como os principais fatores relacionados a dificuldade de programação.

1.2 OBJETIVOS

Apresentar uma ferramenta de algoritmo animado para o estágio inicial do aprendizado de programação e transformá-los em animações em uma máquina que simula o ciclo de instrução na arquitetura de *von Neumann*, que se aproveitam de pontos fortes de jogos educativos, apresentando contextualização visual e narrativa embasada na teoria da cognição situada.

Como subproduto do objetivo principal propõe-se:

- Disponibilizar online uma ferramenta (framework) de objetos de aprendizagem com imagens e clipes de animação prontos para uso em web que sejam capazes de construir animações de programas simples de forma a abstrair o modelo de *von Neumann*.

Ao buscar atingir o objetivo principal, durante o processo de avaliação será promovido uma pesquisa com um entrevistas qualitativas e questionários com professores e pesquisadores do departamento das áreas de sistemas, arquitetura de *von Neumann* e outras especialidades, deste modo atingindo os seguintes objetivos adicionais:

- Avaliar se os professores consideram a solução viável e com potencial para ser utilizada em sala de aula.

- Identificar o histórico de utilização de soluções de algoritmos animados e visualização de software pelos professores participantes da pesquisa.
- Demonstrar um planejamento de aplicação e correções da proposta com base nos resultados obtidos com a pesquisa realizada com os professores.

1.3 JUSTIFICATIVA

O primeiro contato dos alunos com um determinado conteúdo pode reverberar no desempenho e motivação do mesmo durante toda a duração da disciplina. Levando em conta a importância do conteúdo de algoritmos como o pilar que fundamenta o ensino dos cursos voltados a computação (de L. Junior et al., 2015), uma primeira impressão desmotivadora e confusa pode afastar alunos não só desse conteúdo, mas como de todas as demais disciplinas de cursos da área de ciência da computação.

Contribuindo com atenção específica na introdução dos alunos aos algoritmos e seu funcionamento, acredita-se que seja possível promover o interesse não só daqueles alunos que possuem mais facilidade com o tema, mas também os que estão do outro lado deste espectro e possuem dificuldades com os conceitos mais básicos.

Ao abordar o conteúdo de algoritmos através das técnicas de algoritmos animados, compreende-se que entender o funcionamento do computador e a forma com que ele executa cada comando de um algoritmo é essencial para uma melhor compreensão do que ele está desenvolvendo e como cada linha de algoritmo interage com o computador. Devido a isto vê-se necessário a união do conceito de algoritmos com a arquitetura de *von Neumann* nas animações, refletindo também o método de ensino já utilizado em sala de aula na disciplina de Algoritmos e Estrutura de Dados I do curso de ciência da computação da UFPR (Castilho et al., 2020).

A arquitetura aqui entra não só para se adequar as metodologias já utilizadas, mas também para embasar em conteúdos teóricos já vistos pelos alunos ao contexto das animações e elementos apresentados em cena. Desta forma possibilitando a aproximação das animações do conteúdo ministrado pelo professor além do dia a dia e cultura do aluno.

Para se aproximar ainda mais cotidiano e dia a dia dos alunos, a opção de entregar visuais próximos a soluções de jogos digitais tem como objetivo dar as animações suporte em elementos visuais atuais aos alunos, possibilitando representações com elementos de obras de entretenimento presentes na cultura atual, visto que trabalhos anteriores demonstram o potencial que soluções que se utilizam do estilo de narrativa e visuais de jogos digitais possuem neste âmbito de contextualização (Eck, 2006).

A união dos elementos de algoritmos animados junto da contextualização fundamentada pela teoria da cognição situada e uso de narrativa e visuais de jogos digitais habilita novas possibilidades no campo. Acredita-se que as possibilidades de interação do aluno com o conteúdo de algoritmos e a arquitetura *von Neumann* do aluno através desta nova abordagem permite um novo conjunto de estímulos, motivando e facilitando a abstração necessária na disciplina, consequentemente tornando o conteúdo mais acessível.

1.4 ORGANIZAÇÃO

Os capítulos a seguir estão organizados de modo a apresentar os conteúdos relevantes a esta pesquisa. No capítulo 2 é demonstrado o referencial teórico que embasa a dissertação, contendo a explicação de conceitos e histórico de trabalhos relacionados a esta pesquisa desenvolvidos até então. O capítulo 3 apresenta o processo de desenvolvimento da ferramentas de

algoritmos animados com cognição situada denominada Core Inc. O capítulo 4 contém o processo de desenvolvimento e preparação dos experimentos realizados e a discussão dos resultados obtidos. Por fim o capítulo 5 apresenta as conclusões finais obtidas através do desenvolvimento desta dissertação.

2 REFERENCIAL TEÓRICO

Este capítulo tem como objetivo apresentar as informações teóricas relevantes para a compreensão desta dissertação. Através de diferentes sessões serão apresentados logo a seguir os conceitos de Cognição Situada, Modelo de *von Neumann* e Algoritmo Animado, junto com o histórico de desenvolvimento deste tipo de software e trabalhos similares. Além destes três pilares que fundamentam a dissertação também serão apresentados os conceitos que fundamentam as escolhas artísticas por trás do desenvolvimento da proposta de algoritmos animados no capítulo 3 e a o método de avaliação utilizado no capítulo 4.

2.1 COGNIÇÃO SITUADA

A Teoria da Cognição Situada trouxe uma nova perspectiva quanto as ciências cognitivas. Estas que buscam construir, compreender e refletir sobre como se dá e se transmite o conhecimento humano (Venâncio e Borges, 2007).

Inicialmente as chamadas abordagens físicas consideravam a informação como algo concreto, e a transmissão dessa informação como um sistema, todos estes com o objetivo de suprir uma demanda específica, relacionando-se diretamente com a teoria da informação de Shannon (Shannon, 1948), onde a semântica da mensagem não importa, e o que é relevante é desenvolver-se um sistema em que a mensagem, seja qual for, chegará com a mesma quantidade de dados ao seu destinatário. Não há um processo na comunicação neste caso, e sim um sistema que pode ser medido e calculado.

Contudo, ao longo do tempo estudos demonstravam que abordagens chamadas de cognitivistas obtinham mais sucesso quanto as físicas, causando uma mudança de foco das pesquisas. Campos e Venâncio explicam como a passagem da abordagem física para a cognitivista foi sinalizada com a criação do “objeto mental” e da representação mental do mundo externo (de B. Campos e Venâncio, 2007).

As abordagens cognitivistas passaram a considerar a existência de um mundo externo e a mente como uma máquina, capaz de utilizar-se de ferramentas para decifrar as informações vindas do mundo externo (Venâncio e Borges, 2007). Os estudos cognitivistas abordam como o destinatário da informação recebe os estímulos do mundo externo e as processa com base nas ferramentas já disponíveis em sua mente, e como tais informações são armazenadas na memória (Magro, 1999) (von Krogh e Roos, 1995).

A Teoria da Cognição Situada surgiu inicialmente com Lave (Lave, 1988b), e possui abordagens diversas e diferentes, sendo elas: o conceito de biologia do conhecimento (Maturana e Varela, 1995); Cognição Situada, escrita por Clancey (Clancey, 1997), *Enactive View* (Varela et al., 1991); e Ecologia da Mente (Bateson, 1972).

A Cognição Situada aponta que o mundo externo, e o contexto da informação não pode ser separada da informação em si, e que ao ignorar isto a educação sabota a ela mesma ao buscar atingir o objetivo de transmitir conhecimento útil (Brown et al., 1989). Enquanto em teorias cognitivistas os indivíduos são considerados sistemas abertos, capazes de receberem informações, processarem-nas e devolverem uma saída com base em suas ferramentas internas. De forma que apresenta diferenças fundamentais com a cognição situada que possui um entendimento diferente de como essas informações interagem com as pessoas.

Os estudos iniciados por Lave (Lave, 1988b) buscavam relacionar conhecimentos matemáticos ensinados em sala de aula com sua aplicação no dia a dia, estendendo a aplicação

dos conceitos para o dia a dia dos estudantes. Discutindo conceitos relacionados a cognição do ser humano, Lave levanta discussões sobre a relação da cultura, aprendizado e a forma como a cognição das pessoas funciona quanto ao aprendizado e aplicação de conhecimentos, assim como a dificuldade que áreas como psicologia e antropologia possuíam em desenvolver teorias que explicassem esta relação de forma a satisfazer os pesquisadores de suas respectivas áreas.

A partir disto, discussões sobre transferência de conhecimento através de atividades diferentes são apresentadas por Lave e demais pesquisadores. Com experimentos sendo realizados, buscou-se identificar a relação entre conceitos matemáticos ensinados em sala de aula e tarefas diárias dos docentes. Abordando situações relacionadas a decisões de compra de produtos em supermercados ou trabalhos de estoquistas entregadores e outras funções que dependam de cálculos para execução de tarefas. Ao deparar-se com dificuldades em relacionar ambos ambientes e demonstrar a transferência de conhecimento entre sala de aula e o dia a dia das pessoas, Lave atribui parte desta dificuldade ao processo de elitização do ambiente acadêmico, onde os conhecimentos passados em sala de aula passaram a ser considerados mais difíceis e importantes que conhecimentos da “vida real”. Desta forma, causando um isolamento do conhecimento ensinado na academia em relação ao contexto e vivência dos docentes, ignorando a cultura destes indivíduos.

Segundo Maturana e Varela (Maturana e Varela, 1995), os seres vivos são um sistema fechado e estruturado. Ou seja, um agente do mundo externo vai causar mudanças estruturais determinadas pelo próprio sistema fechado do indivíduo. Portanto, a cognição não é uma representação independente do mundo externo e sim um processo constante de desenvolvimento, que se dá pela vivência do indivíduo (Venâncio e Nassif, 2008). Entende-se então que o mundo externo e seus agentes não são capazes de provocar mudanças específicas no processo cognitivo durante o aprendizado e sim são capazes de provocar estímulos para estas mudanças estruturais, entretanto essas mudanças são decididas pelo próprio sistema fechado dos indivíduos, que se molda através desta interação constante com o ambiente.

Portanto, entende-se através da cognição situada que o mundo externo não é um conjunto de informações já determinado que aguardam a interação de cada ser humano, mas sim o resultado da interação constante entre nós e o ambiente. O local onde vivemos é moldado pelas nossas interações com este ambiente porque depende das próprias capacidades que nossos sistemas possuem de interagir com o mesmo.

Através destas interações constantes é que se dá a obtenção do conhecimento, Maturana e Varela explicam que o conhecimento não é um armazenamento de informações ou a obtenção de dados no meio externo e sim o resultado desta interação constante entre o meio e o indivíduo. As ações que o indivíduo toma interagem com o mundo externo e seus agentes, permitindo que ele obtenha conhecimento através desta interação, desencadeando processos cognitivos em seu sistema baseado na forma com que ele interagiu com o meio.

Essa falta de contexto, ou mais especificamente vivência, na abordagem de ensino de diversas instituições e educadores afeta negativamente o aprendizado dos alunos, separando o conteúdo acadêmico com o conhecimento do dia a dia dos alunos. É comum que estudantes de ciência da computação recebam como ferramentas algoritmos, rotinas e definições que eles não possuem conhecimento de onde aplicar e como afeta o mundo ao seu redor. Portanto muitas destas ficam paradas na mente dos alunos, criando barreiras nesta interação constante entre os agentes externos e a cognição dos alunos. Na melhor das hipóteses tais alunos são capazes de aprender a como usar a ferramenta em determinado contexto, porém ao se modificar o contexto o conhecimento também se perde (Brown et al., 1989).

Brown e seus colegas (Brown et al., 1989) ainda complementam dizendo que:

“pessoas que usam as ferramentas ativamente, ao invés de só as adquirir, constroem um entendimento implícito mais rico do mundo nos quais as ferramentas são utilizadas e do próprio mundo em si”.

No princípio dos estudos, demonstrava-se que no ensino a alunos de ensino superior de cursos de matemática já era impactado positivamente pelas metodologias que se utilizavam da cognição situada (Lave, 1988a), (Schoenfeld, 1985), (Lampert, 1986).

2.2 ARQUITETURA DE VON NEUMANN

A arquitetura ou modelo de *von Neumann* é um dos fundamentos da execução de algoritmos nas máquinas contemporâneas e conteúdo fundamental no princípio dos cursos de ciência da computação. Os materiais utilizados nestas disciplinas, em específico nos cursos do DINF-UFPR abordam inicialmente estes conteúdos (Castilho et al., 2020).

A arquitetura das máquinas atuais, desenvolvida pela equipe liderada por *von Neumann*, foi capaz no período de colocar tanto instruções de um programa quanto seus dados em memória. Este fator estabeleceu a arquitetura da figura 2.20 que perdura até hoje.

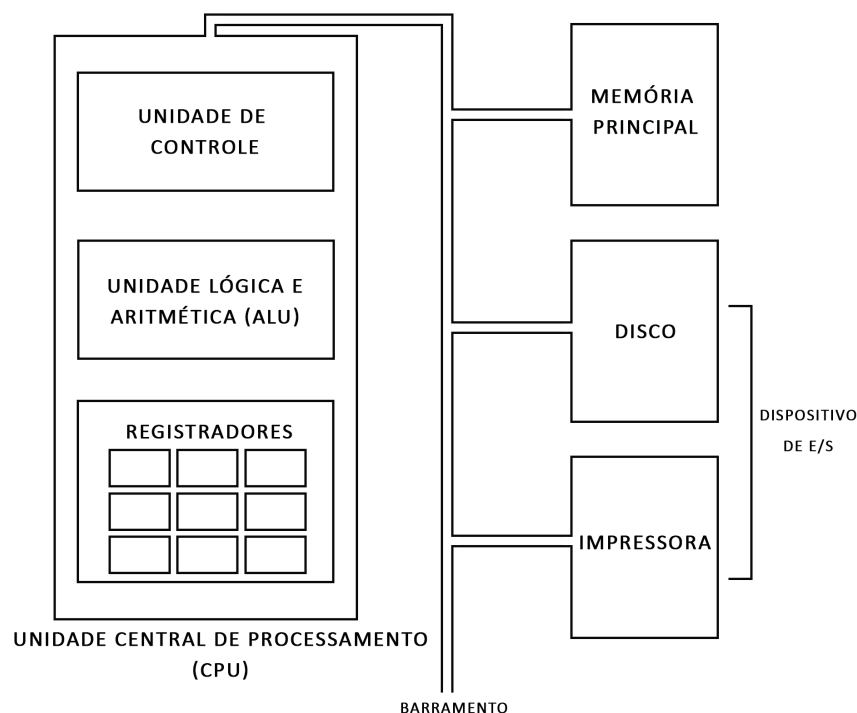


Figura 2.1: Arquitetura de von Neumann.(Tanenbaum, 2013)

A organização da arquitetura pode ser explicada passando pela função de cada componente. Primeiramente, a unidade central de processamento ou CPU tem a função de executar os programas na memória, decodificando as instruções e coletando os dados necessários. Ela possui várias partes distintas, sendo elas: unidade de controle que é responsável por buscar as instruções na memória principal e decodificá-las; a unidade lógica e aritmética efetua operações aritméticas

e booleanas; além disso, os registradores são memórias de velocidade muito alta localizados dentro da CPU, utilizadas para armazenar dados momentaneamente durante a realização e operações. Dentre os registradores disponíveis existem os responsáveis por contar a próxima instrução do programa a ser buscada e o registrador que armazena a instrução sendo executada no exato momento (Tanenbaum, 2013).

A arquitetura ao executar um programa utiliza a execução de instrução, composta por sete etapas, elas são reproduzidas ciclicamente até o fim das instruções do programa. Estas etapas são (Tanenbaum, 2013):

1. Levar a próxima instrução da memória para o registrador;
2. Modificar o contador de memória para apontar para a próxima instrução;
3. Definir o tipo de instrução trazida;
4. Se a instrução usar um dado de memória, determinar onde o dado está;
5. Trazer o dado para o registrador da CPU, caso necessário;
6. Executar a instrução;
7. Voltar a etapa 1 para começar a próxima instrução.

Estas etapas são o guia básico utilizado para entender o funcionamento de um algoritmo em uma máquina contemporânea, assim como é utilizado para fundamentar o desenvolvimento da solução proposta no capítulo 3.

2.3 ALGORITMO ANIMADO

O sistemas do tipo de algoritmo animado despontaram como uma área de interesse na década de 1980, quando a capacidade computacional cresceu o suficiente para abrir portas para abordagens visuais de algoritmos. Dois dos pioneiros neste campo de pesquisa explicam que a motivação inicial para o desenvolvimento destas técnicas foi a capacidade computacional dos equipamentos da sua universidade sendo subaproveitados em modelos tradicionais de computação (Brown e Sedgewick, 1985).

Em um trabalho posterior Stasko define na página 1 de (Stasko, 1990a): “Animação de algoritmos é o processo de abstrair dados, operações e semântica, criando representações visuais destas abstrações.”

No entanto, ele mesmo evidencia a importância de diferenciar uma solução de *software visualization*, que somente transpõe os dados de um programa para uma versão visual, do conceito de algoritmo animado.

A nomenclatura e execução de cada um desses tipos de software pode ser confusa, e suas diferenças aparentam ser mínimas entre si, contudo são essenciais para diferenciar-se as soluções desenvolvidas, e principalmente, quais delas são consideradas pesquisas com algoritmos animados e porquê.

No caso das soluções de *software visualization*, segundo a taxonomia desenvolvida por Price e colegas (Price et al., 1993), o termo engloba praticamente qualquer tipo de software que tenha algum estímulo visual em mente como visto na taxonomia da Figura 2.1. Eles definem essas soluções como o uso de tipografias, design gráfico, animação e cinematografia com interação humano-computador que busque facilitar a compreensão e efetividade do uso de software (Price et al., 1993).

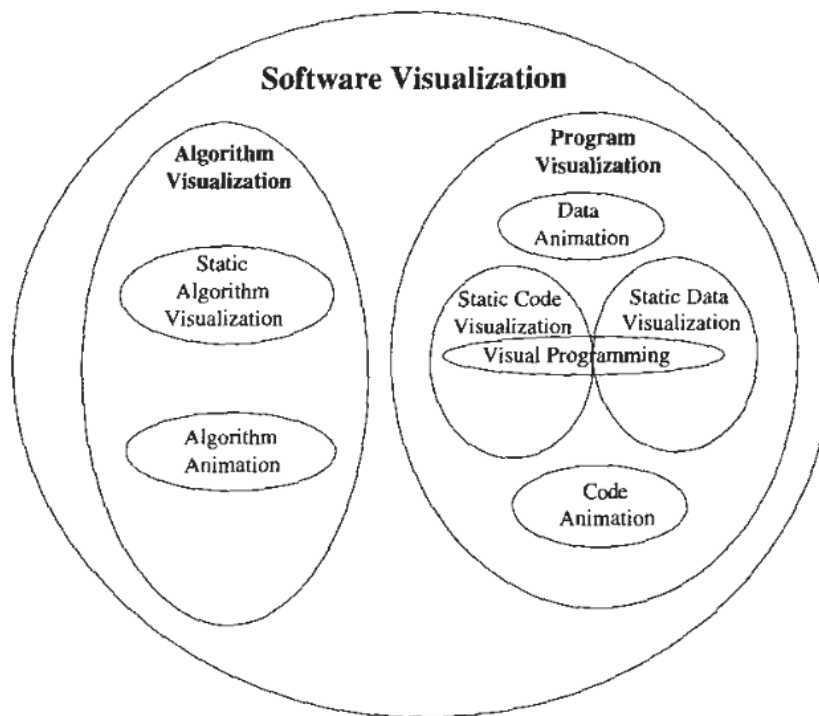


Figura 2.2: Taxonomia de Software Visualization. (Price et al., 1993)

Algoritmos animados fazem sim parte das soluções de *software visualization*, mas não compartilham todas as características entre si, inseridas na área de visualização de algoritmos. Segundo a taxonomia de Price (Price et al., 1993) a abordagem se difere da visualização de programa por apresentar abstrações de alto nível que descrevem a execução do software. Enquanto isso a visualização de programa tem a preocupação de demonstrar o fluxo do código e os dados (Price et al., 1993), de forma semelhante ao que os depuradores fazem atualmente. Quanto à diferenciação de abordagens dentro da subárea de visualização de algoritmos, separam-se em dois tipos:

- *Visualização de Algoritmos Estática*: representações de alto nível que se utilizam de imagens e gráficos estáticos;
- *Algoritmos Animados*: representações de alto nível de um software alvo que usam animações fluídas.

Stasko (Stasko, 1990b) ainda lista alguns pontos importantes onde os algoritmos animados têm capacidade de ajudar alunos e programadores, sendo estes:

1. *Entender programas*. Ou seja, visualizar as soluções propostas em código fonte agirem em tempo real, auxiliando na compreensão das ações tomadas internamente pelo código, como na manipulação de dados;
2. *Avaliar programas existentes*. A visualização gráfica e em tempo real das ações do código permitem ao autor identificar mais facilmente falhas e pontos de melhora;
3. *Desenvolver novos programas*. Ao se deparar com novos comandos, funções e métodos disponíveis na linguagem de programação, a representação gráfica auxilia na

identificação do que cada uma das novas funcionalidades é capaz de fazer e como é seu comportamento durante estas ações.

Quanto as abordagens na execução destes software, elas diferem-se de duas formas distintas:

Abordagem Passiva: Aquelas em que o usuário é um espectador passivo, ou seja, ele somente assiste à execução de uma animação de um algoritmo específico (Garcia et al., 1997), normalmente preparado pelo educador. Exemplos próximos a esta abordagem que também podem ser citados são atividades como o aluno assistir um filme, ou uma apresentação de slides. Alguns dos trabalhos pioneiros como por exemplo de Brown e Sedgewick (Brown e Sedgewick, 1985) sobre o tema tinham essa abordagem, que ainda é utilizada em trabalhos mais recentes, tais como (dos Santos e Costa, 2005) e (Falkembach et al., 2003), trabalhos estes característicos por entregar um conjunto de algoritmos, imagens ou animações estáticas, não possibilitando interatividade com o aluno.

Porém, há barreiras óbvias nesta abordagem. O número de exemplos possíveis é limitado pelo que foi desenvolvido pelo educador, além da capacidade de interação baixa do usuário com o software, o que é contra intuitivo quando se olha para os principais pontos em que os algoritmos animados são capazes de auxiliar o usuário. Sabendo disto, abordagens que beneficiaram a interatividade não demoraram a surgir.

Abordagem Ativa: Características por permitir a interação direta do aluno com o desenvolvimento da animação, possibilitando que ele a modifique e adapte aos programas que ele desejar (Brown, 1991), (Stasko, 1990b), (Garcia et al., 1997), (Roman et al., 1992), de forma que o aluno não só assista as animações, mas também seja capaz participar do processo de desenvolvimento das mesmas.

Nos anos de 1990, Lawrance e seus colegas (Lawrance et al., 1994) em seus experimentos na utilização de algoritmos animados de abordagem ativa identificaram um impacto mais positivo dos alunos que utilizaram soluções ativas ao invés de métodos passivos ou aulas tradicionais. Este tipo de resultado favorecendo abordagens que permitam uma maior interatividade dos alunos fez com que mais soluções de algoritmos animados ativas fossem desenvolvidas ao longo dos anos ao invés de métodos passivos e os estudos se voltassem mais a este tipo de solução interativa.

Ao estudar a literatura nota-se também que não só o ato de animar foi considerado importante. Alguns trabalhos na literatura passaram a levar em conta no desenvolvimento destas soluções a importância das animações fluídas (Roman et al., 1992) e (Stasko, 1990a), em contraste com representações pouco polidas ou simples demais.

Animações suaves são mais eficazes para manter o usuário dentro do contexto proposto, enquanto imagens mais claras das modificações que ocorrem durante a execução do programa são capazes de manter melhor a atenção do usuário (Demetrescu e Finocchi, 2001). Sabendo disto Stasko desenvolveu o paradigma da transição de caminho (*the path-transition paradigm*) (Stasko, 1990b) como um método para produzir animações coloridas, bidimensionais e em tempo real. Ao abordar quatro tipos diferentes de dados abstratos, sendo eles: localização, imagem, caminho e transição; ele também definiu como desejáveis quatro elementos para uma ferramenta de animação:

- Os métodos e procedimentos do software deveriam ser orientados a entrada de dados que definem como a animação se move e modifica;
- A animação deve ser realizada em tempo real;
- A tela deve ser constantemente atualizada, evitando animações que parecem fotografias intercaladas;

- Fácil de aprender e implementar.

Ao atualizar as animações com base nesses quatro elementos o paradigma se adequa bem às soluções que lidam com abordagens que utilizam identificação de eventos relevantes no código, estes sendo aqueles que alteram significativamente os dados do programa. Esse tipo de solução acaba por ser mais eficiente quando se trata de entregar animações e transições mais suaves do que sistemas que confiam exclusivamente na exibição das estruturas de dados do algoritmo (Demetrescu e Finocchi, 2001).

2.4 TRABALHOS RELACIONADOS A ALGORITMOS ANIMADOS

A seguir são listados alguns dos trabalhos que se dedicam ao estudo de algoritmos animados e são mais relevantes na literatura, junto de uma descrição de suas abordagens. Estão presentes software que dão suporte à criação de animações tanto ativas quanto passivas. Soluções que funcionam em tempo de execução ou não. Além de frameworks, linguagens e ambientes que facilitam o desenvolvimento de algoritmos animados.

Os autores Brown e Sedgewick (Brown e Sedgewick, 1985) foram pioneiros quando se trata de algoritmos animados. Os autores desenvolveram o software Balsa (*Brown Algorithm Simulator and Animator*) que, segundo os próprios autores, permite a construção e customização de “filmes” com o propósito de ilustrar programas e suas propriedades.

Desenvolver os scripts que geram as animações se dá por diversas etapas, envolvendo profissionais de diferentes áreas, onde o responsável pelo desenvolvimento do algoritmo define os momentos em que a animação precisa ser atualizada para que o programador em cargo da animação desenvolva o script necessário. No fim, o produto final contém o algoritmo original e o script responsável pela animação.

O software funcionava de forma que ambos algoritmo e scripts de animação sejam armazenados e exibidos para os alunos em sala de aula, de forma passiva, ou seja, sem interação direta dos alunos. Ele permitia a execução dos scripts individualmente pelos alunos, e também um modo de transmissão, onde todas as máquinas espelham as ações do computador do professor, dando mais controle ao mesmo sob o conteúdo ministrado. Os algoritmos demonstrados neste trabalho possuem foco nas disciplinas de algoritmos e estrutura de dados e foram todos desenvolvidos previamente pelos autores.

Na Figura 2.2 é demonstrado o software Balsa apresentando a construção e balanceamento de uma árvore binária e seu histórico de construção, começando do topo e indo até o fim da tela. É perceptível como a interface do Balsa dá um enfoque nas estruturas de dados que ele busca ensinar. Tanto no funcionamento de algoritmos ou estruturas de dados como árvores e grafos ele busca se manter próximo das representações mais tradicionais destas estruturas.

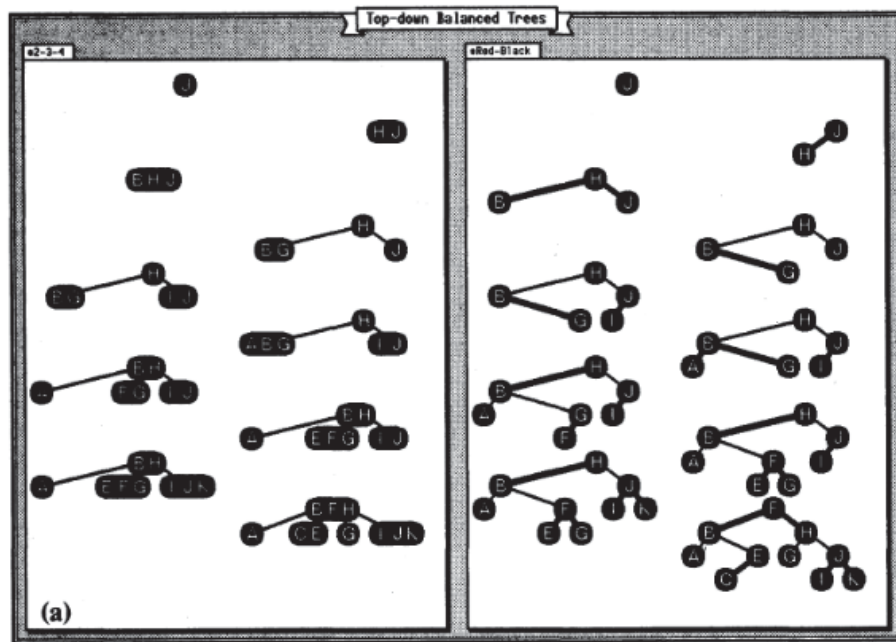


Figura 2.3: Tela do software BALSA. (Brown e Sedgewick, 1985)

Brown (Brown, 1988) desenvolveu a segunda versão de seu software como parte de sua tese de doutorado, chamado BALSA-II, o software é uma versão aprimorada de seu predecessor e foi utilizado em sala de aula a partir de 1987, permitindo a animação de algoritmos na linguagem Pascal desde que fossem construídos dentro do software de animação seguindo três passos:

1. Dividir o programa em três componentes, sendo eles o próprio algoritmo em Pascal, as entradas que contém os dados a serem manipulados e imagens que representam o algoritmo em ação;
2. Implementar cada componente no BALSA-II;
3. Identificar cada componente visual e de entrada de dados que o BALSA-II deve utilizar e demonstrar.

Stasko (Stasko, 1990b) desenvolveu um framework e sistema chamado Tango. O software ajuda definir quais são as operações abstratas de um algoritmo, desenvolver animações para estes tópicos, e relacionar os trechos do código com as animações desenvolvidas. Este processo de animação se inicia com o sistema lendo um arquivo controlador contendo as especificações para a animação, em seguida os arquivos de código fonte da animação desenvolvidos previamente por um designer são conectados ao controlador da animação, por último o Tango aguarda a execução do algoritmo e suas operações, animando as ações pré definidas conforme elas são chamadas.

Esta mesma pesquisa também desenvolveu o paradigma da transição de caminho (*path-transition paradigm*) (Stasko, 1990a), que dá suporte ao software para animações utilizando localização, imagens, caminhos e transições. O autor procura facilitar a construção de animações no Tango por parte do designer, mantendo o software compartimentalizado em suas funcionalidades de algoritmo e animação. Apesar da parte gráfica do Tango ser mais robusta em comparação aos trabalhos de sua época, como observado na Figura 2.3 que demonstra a visualização de um algoritmo de ordenação bubblesort, o processo de desenvolvimento de uma animação ainda é complexo e trabalhoso.

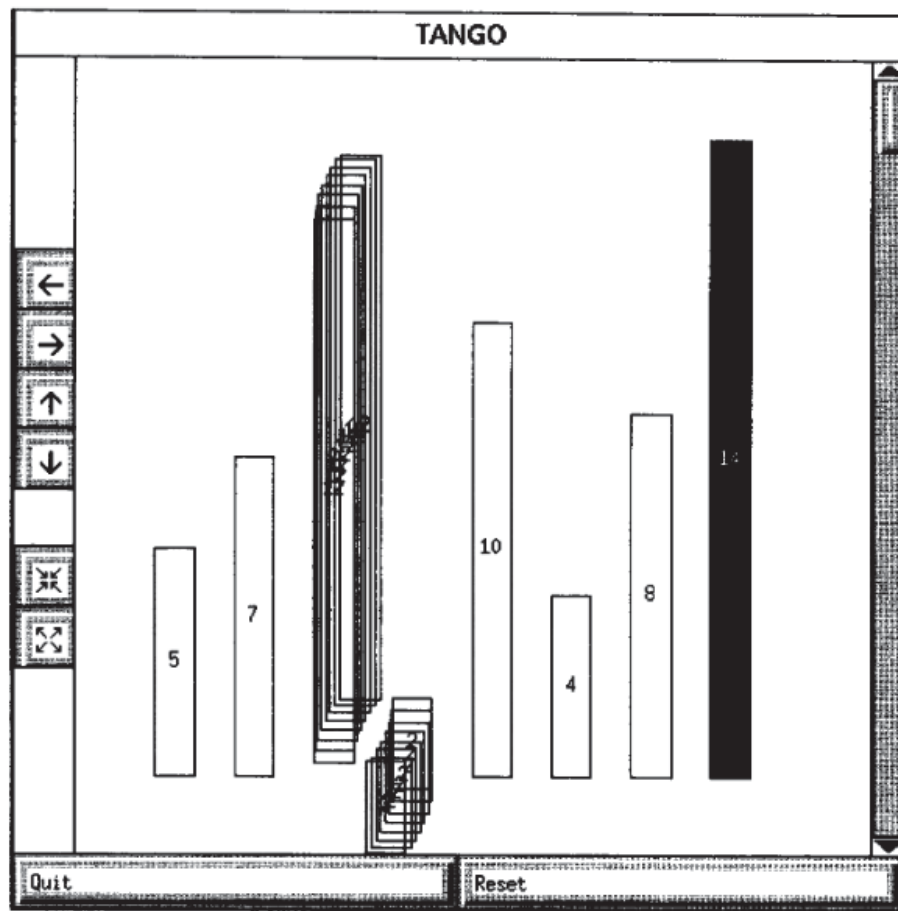


Figura 2.4: Tela do software Tango. (Stasko, 1990a)

Brown (Brown, 1991) prosseguiu com os estudos na área após BALSAs com um novo software chamado Zeus. Ele é um framework para FormsEdit, que é um software para construção de interfaces de usuário com múltiplas telas. Um dos diferenciais de Zeus em comparação ao seu trabalho prévio é que ele permite controlar a velocidade da animação, assim como pausar, reproduzir ou executar passo a passo o algoritmo.

Com base na experiência dos autores com o BALSAs foi dado um grande foco ao desenvolvimento da interface do usuário através de um sistema com opções e janelas mais organizado, ao contrário do trabalho prévio. As telas são representadas na Figura 2.4 que, apesar de difícil visualização hoje, demonstra o painel de controle do software, onde é permitido selecionar o algoritmo a ser executado e a definição e seus parâmetros. Já as animações ainda são geradas por uma linguagem própria para a animação, onde eventos importantes nos algoritmos são identificados com uma forte tipificação, resultando na modificação das animações em momentos relevantes.

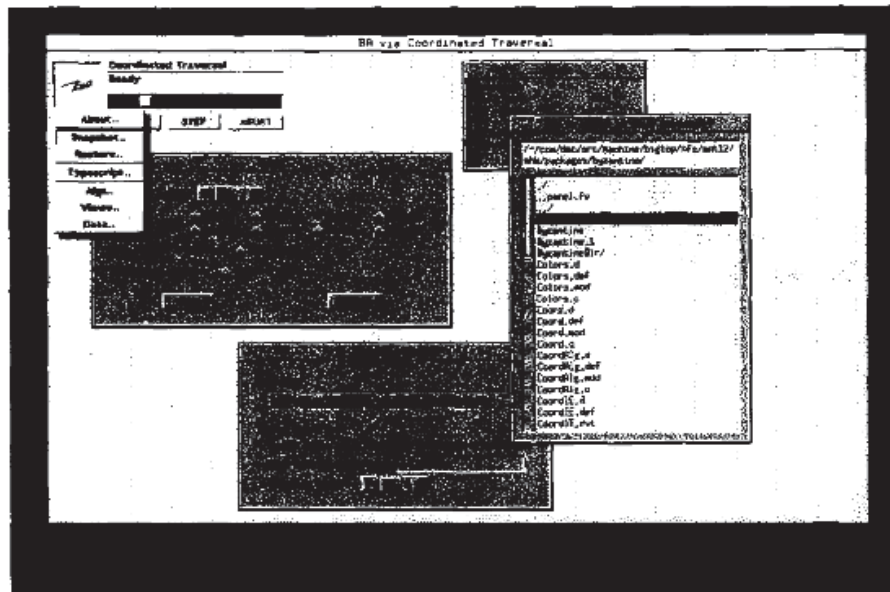


Figura 2.5: Tela do software Zeus. (Brown, 1991)

Os autores Bentley e Kernighan (Bentley e Kernighan, 1991) desenvolveram um sistema chamado ANIM, um software de algoritmos animados focado na facilidade de uso. Segundo os autores o ANIM buscava atingir um espectro diferente em relação aos trabalhos desenvolvidos na época ao sacrificar a qualidade de suas animações para obter facilidade de uso. Este é capaz de realizar animações de caráter semelhante aos trabalhos listados anteriormente, porém com um grau de fidelidade inferior, ainda que use as representações com linhas, pontos e formas geométricas como na Figura 2.5 onde um algoritmo de problema de empacotamento (*bin packing*) é demonstrado. O ANIM também é capaz de suportar o desenvolvimento de animações de grafos e algoritmos de ordenação. Seu diferencial é a rapidez com que as animações são geradas e facilidade de aprendizado da linguagem responsável pela animação, sendo que, segundo os autores, um usuário novo levaria de uma a duas horas para animar um algoritmo. O software também se comporta tanto em modo filme, como em fotografia, onde o usuário seleciona momentos específicos para que imagens do algoritmo previamente animado sejam geradas.

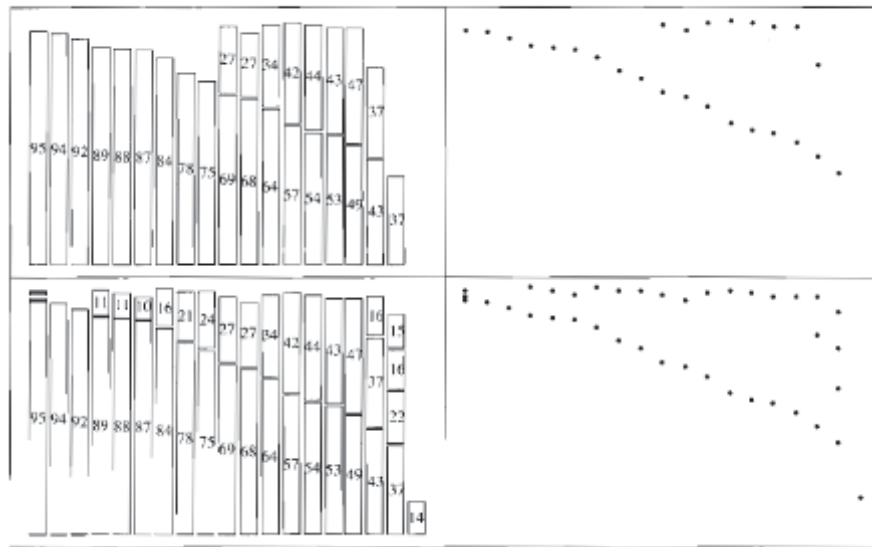


Figura 2.6: Tela do software ANIM. (Bentley e Kernighan, 1991)

Amorim e Rezende (Amorim e de Rezende, 1993) desenvolveram o software Anima, que diferente de seus antecessores, permite uma fácil reutilização de códigos, com um sistema de bibliotecas que permite implementações complexas serem armazenadas, reutilizadas e combinadas posteriormente. Desenvolvido em C++, a principal característica do Anima é a possibilidade de desenvolvimento concorrente entre diversos usuários, onde cada um possui seu ambiente de trabalho individual, apesar de compartilharem o mesmo arquivo executável. O desenvolvimento de algoritmos animados é dividido em três componentes, sendo estes: o algoritmo, as entradas do programa e o módulo responsável pela parte visual.

Carlson e demais (Carlson et al., 1996) tomam uma abordagem diferente dos trabalhos apresentados até o momento, ao dar um foco maior na linguagem responsável por gerar as animações dos algoritmos. Se apoderando do paradigma desenvolvido por Stasko (Stasko, 1990a), eles trazem a soluções de algoritmo animado para a própria linguagem responsável por criar animações. Através de uma interface amigável exibida na Figura 2.6, o usuário pode inserir elementos e animações na interface, junto de parâmetros que a regulam. As animações são capazes de lidar com situações de ordenação de elementos, como demonstrado na figura e matrizes, manipulação e células e dados dentro destas estruturas.

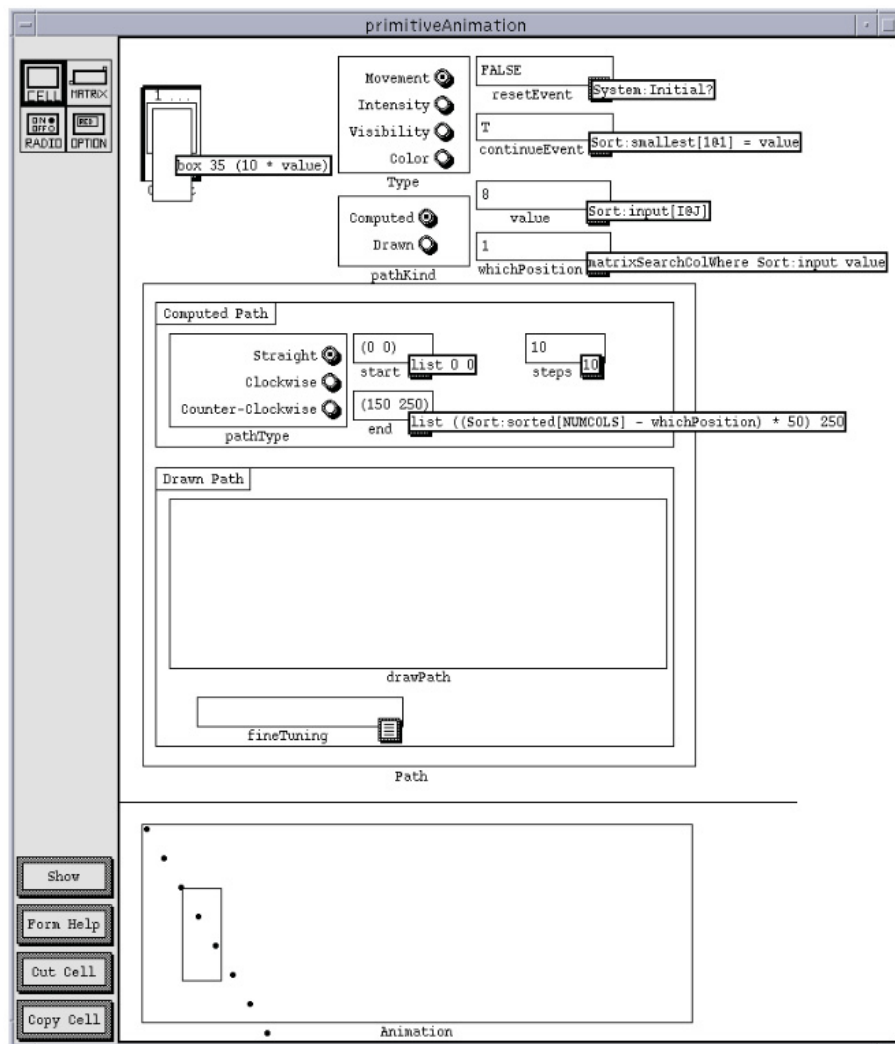


Figura 2.7: Tela do software desenvolvido por Carlson.(Carlson et al., 1996)

Garcia e seus colegas (Garcia et al., 1997) apresentaram o ambiente Astral, visando auxiliar na implementação de animações gráficas para exercícios de estrutura de dados. Os autores demonstram o software lidando com animações sobre algoritmos de ordenação, estruturas de árvores e grafos. Desenvolvido para Macintosh e linguagem Pascal, o software possui uma arquitetura dividida em camadas, cada uma destas responsável por um tipo de estrutura de dados que deve ser animada. Sua utilização se dá pelo usuário interagir com as interfaces disponibilizadas pelo ambiente enquanto projeta seu algoritmo. Utilizando-se da abordagem construtivista de ensino, o Astral reflete o código implementado pelo aluno sempre que as interfaces responsáveis pela animação são chamadas no código, servindo como uma boa ferramenta de depuração para os alunos.

Nos anos subsequentes aos trabalhos de algoritmos animados apresentados, o enfoque neste tipo de software diminuiu, motivados por estudos que são abordados nas próximas seções desta dissertação. Entretanto, esta atenção continuou na área de software de visualização por meio de outros tipos de software similares, entre eles destaca-se o campo de visualização de software. Ainda semelhantes em sua abordagem em relação aos algoritmos animados, eles abrem mão de parte da abstração dos algoritmos animados, buscando uma abordagem mais direta ao código escrito pelo aluno. Trabalhos deste campo de estudo são apresentados na seção seguinte.

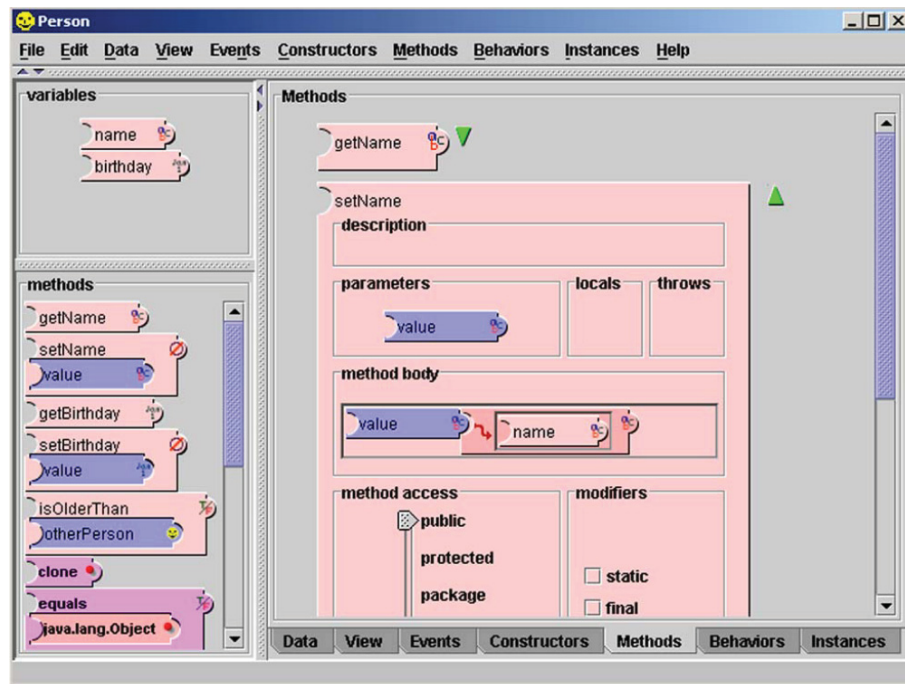


Figura 2.9: Tela do software JPie.(Goldman, 2004)

Os autores Henriksen e Kolling (Kölling, 2008) abordaram o ensino de programação orientada a objetos para alunos ainda no ensino médio. Para isto foi desenvolvido o Greenfoot, utilizando da ferramenta BlueJ, outro software com o objetivo educacional de auxiliar no ensino de programação orientada a objetos. O Greenfoot busca unir os conceitos abordados com o mundo real, dando a eles uma representação gráfica, como é observável na Figura 2.9. Os comandos são realizados através de interações com os objetos em tela ou adição de novos métodos, também através da interface.

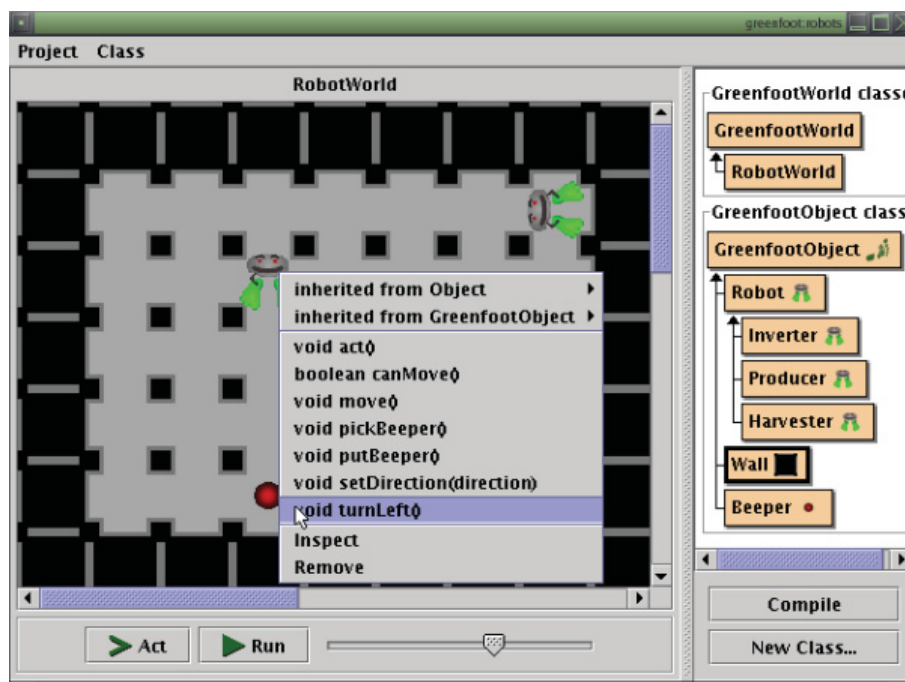


Figura 2.10: Tela do software Greenfoot.(Kölling, 2008)

Ambos Helminen e Malmi (Helminen e Malmi, 2010) desenvolveram o Jype, uma ferramenta web que permite a execução linha por linha de programas na linguagem Python, com enfoque no aprendizado dos conceitos fundamentais de programação. Caracterizado pelos autores como um depurador visual, ele busca auxiliar os estudantes a resolver problemas em seus códigos fonte permitindo um feedback visual e mais veloz como demonstrado na Figura 2.10, permitindo os estudantes até mesmo executarem o programa de trás para a frente.

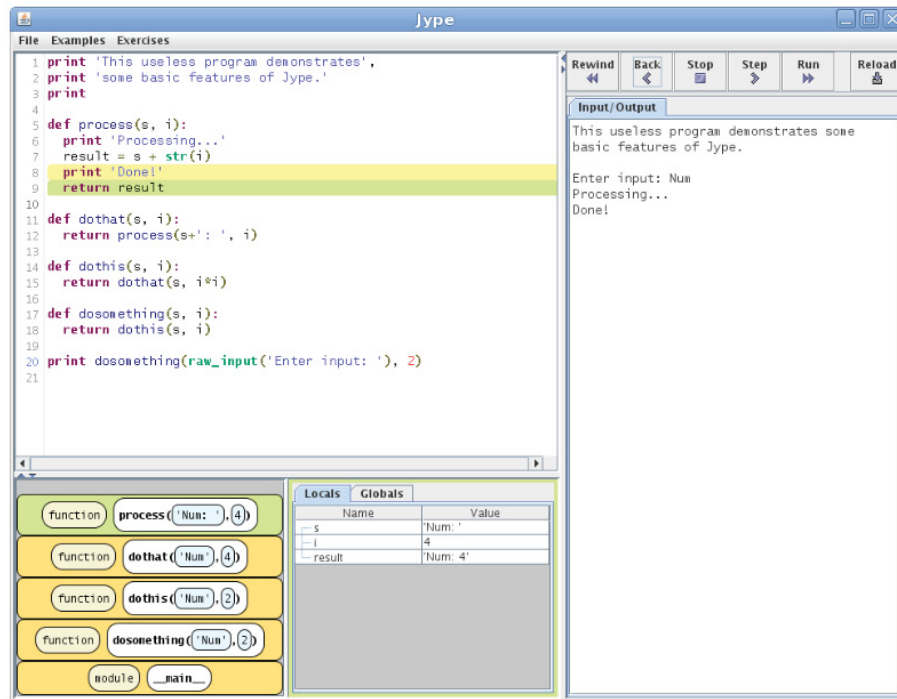


Figura 2.11: Tela do software Jype.(Helminen e Malmi, 2010)

Madow e seus colegas (Madow et al., 2010) demonstraram uma ferramenta gráfica animada de simulação, para o ensino de algoritmos de busca heurística em inteligência artificial. Chamado de AIDA, a ferramenta tem uma interface demonstrada na Figura 2.11, que permite os estudantes editarem um tabuleiro quadriculado, definindo obstáculos e os nodos de início e fim, após isto é possível executar uma diversidade de algoritmos de busca passo a passo, permitindo compará-los e verificar seus rendimentos. Além disto, o software busca sanar com mais facilidade as dúvidas mais frequentes dos alunos sobre os algoritmos.

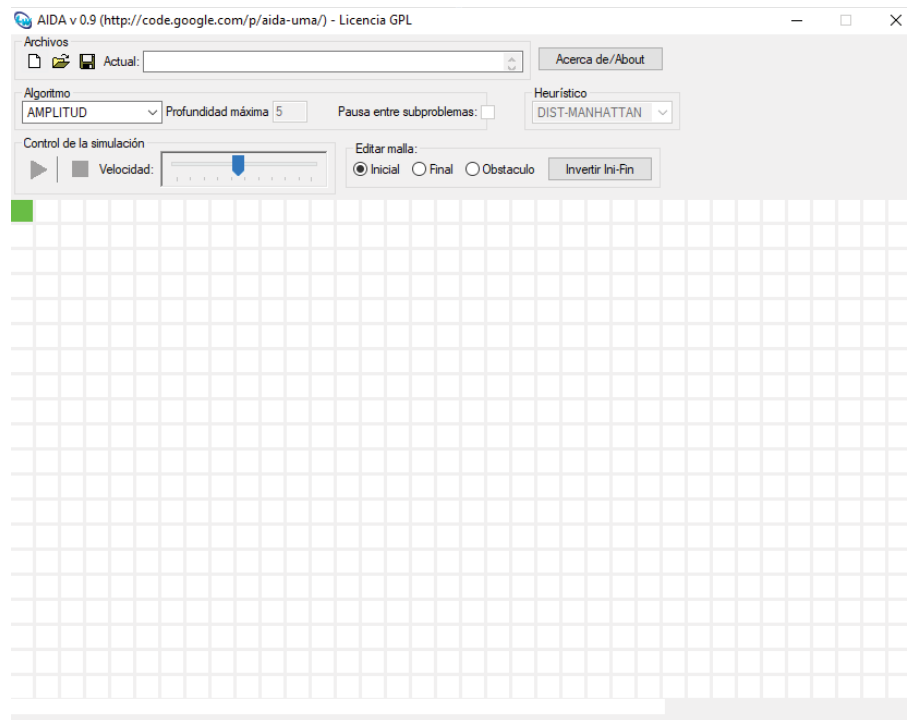


Figura 2.12: Tela do software AIDA. O Autor

Maloney e demais (Maloney et al., 2010) desenvolveram um ambiente de programação visual para web e desktop com o público alvo de jovens entre 8 e 16 anos, permitindo o aprendizado de conceitos de programação através de uma interface de comandos de puxar e arrastar blocos, com um grau de interatividade alto com objetos gráficos em 2D como demonstrado na figura 2.12. O projeto tem o propósito de dar suporte ao auto aprendizado através da colaboração com outros usuários e o desenvolvimento de projetos que sejam de interesse dos usuários, permitindo uma ampla variedade de aplicações, indo de programas tradicionais chegando até mesmo em jogos e histórias animadas.

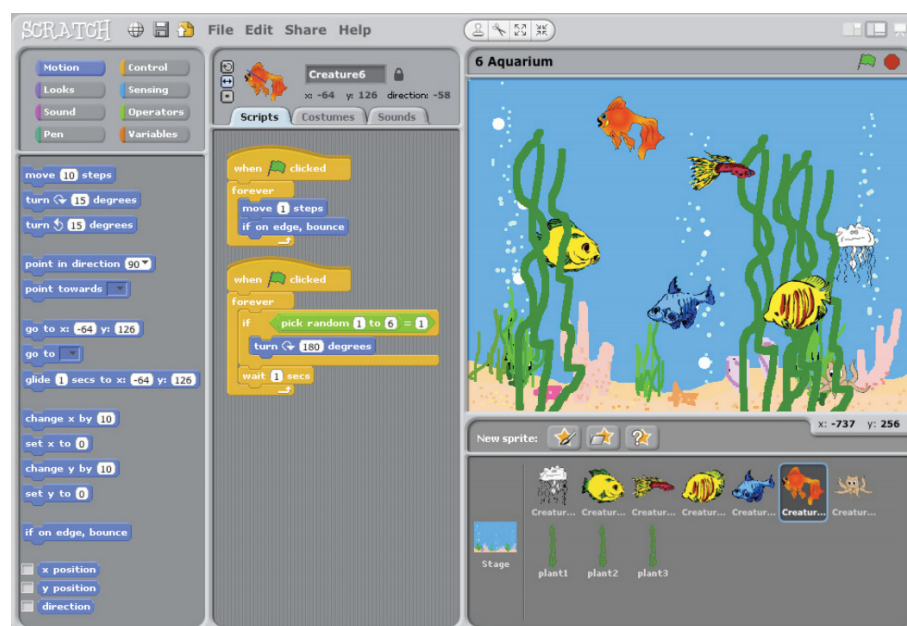


Figura 2.13: Tela do software Scratch. (Maloney et al., 2010)

Farrugia e Quigley (Farrugia e Quigley, 2011) ao realizarem experimentos comparando a eficiência do aprendizado de grafos através de exemplos estáticos em relação a animação de grafos, vieram a desenvolver uma plataforma web com Flash e Javascript, capaz de tocar vídeos de animações, permitindo os participantes do experimento pausarem e alterarem a velocidade da animação, como é possível observar na Figura 2.13.

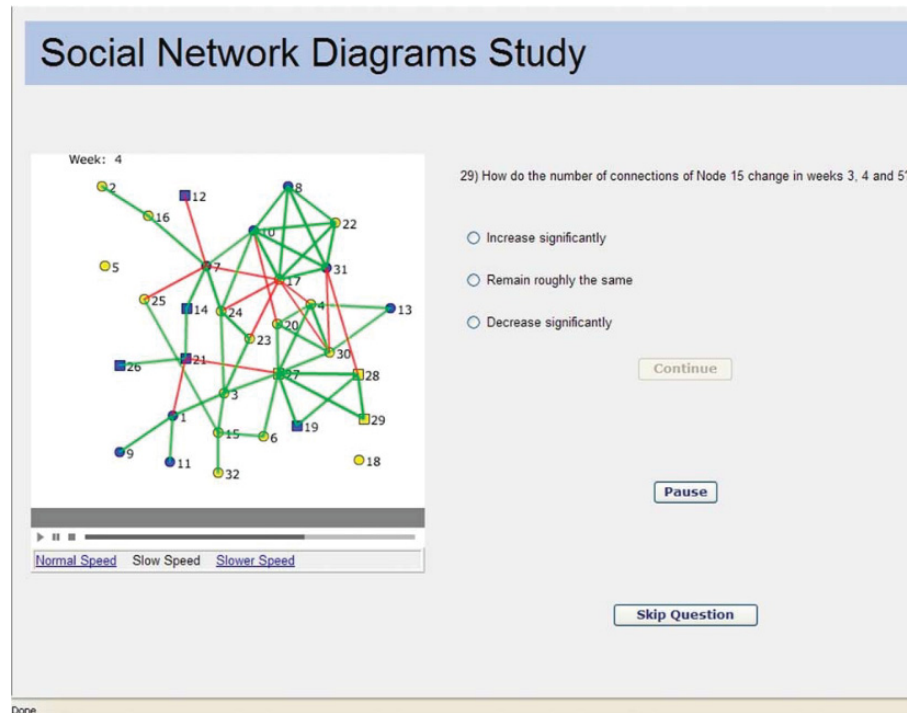


Figura 2.14: Tela do software de Farrugia.(Farrugia e Quigley, 2011)

Baloukas (Baloukas, 2012) apresentou o JAVENGA, um ambiente de visualização de algoritmos para propósitos educacionais, com enfoque no ensino de grafos e algoritmos de rede. Permitindo visualização de algoritmos tradicionais de grafos e redes, além de uma representação complexa do algoritmo *Network Primal Simplex*. O software utiliza-se tanto de imagens estáticas quanto de visualização dinâmica. Através da interface demonstrada na Figura 2.14, o usuário é capaz de criar direta e indiretamente grafos e escolher um algoritmo para resolver o problema desejado.

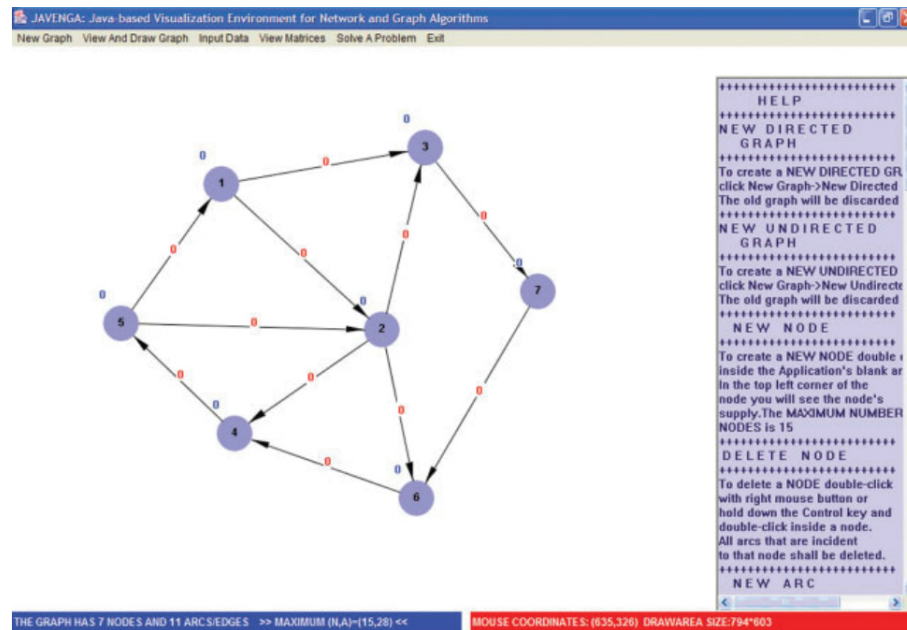


Figura 2.15: Tela do software JAVENGA.(Baloukas, 2012)

Ambos Buchanan e Laviola Jr. (Buchanan e Laviola, 2014) demonstraram o CSTutor, uma interface baseada em rascunhos, desenvolvida para auxiliar estudantes a entenderem estruturas de dados, sendo elas: listas ligadas, árvores binárias, árvore AVL, e heaps. O software permite que os alunos desenhem em um quadro branco dentro do software seus diagramas, que são analisados em tempo real, gerando um código fonte editável correspondente, como demonstrado pela figura 2.15. Junto disto, qualquer mudança no código fonte se reflete no desenho do diagrama também. O objetivo dos autores com esta solução é diminuir a distância entre diagramas conceituais e o código fonte dos programas.

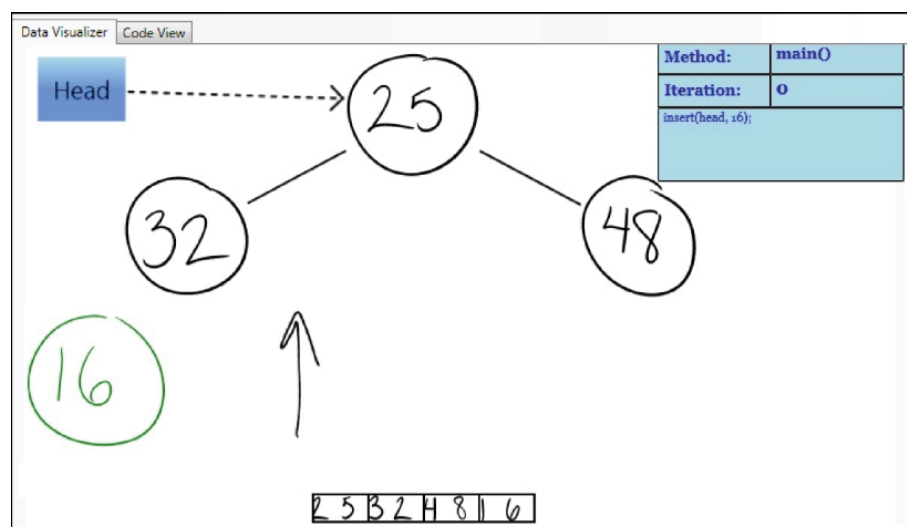


Figura 2.16: Tela do software CSTutor.(Buchanan e Laviola, 2014)

Os autores dos Santos e Costa (dos Santos e Costa, 2005) apresentaram um sistema com propósito educacional chamado TBC-AED. Com enfoque nos conteúdos fundamentais ao ensino de programação, segundo os autores, ele sintetiza os tópicos básicos de programação, junto de recursos gráficos passo a passo. Apesar de não tomar para si especificamente uma abordagem de

visualização de software, partes do conteúdo demonstram soluções de visualização de programas de forma passiva, como demonstrado na Figura 2.16. Além disto, os autores reiteram como o software possui descrições de todos os elementos em tela, visando alunos iniciantes dos cursos de ciência da computação.



Figura 2.17: Tela do software TBC-AED.(dos Santos e Costa, 2005)

A identificação dos trabalhos nas seções 2.2. e 2.3. aqui demonstrados possui como propósito garantir que a proposta de solução de algoritmos animados embasada na teoria da cognição situada é relevante. Para isto foi levantado o histórico de trabalhos desenvolvidos nesta área de estudo, de forma que seja possível traçar comparativos em relação a facilidade de uso e contextualização do modelo dos algoritmos animados e o modelo de *von Neumann*.

Dado este objetivo, a seção 2.2. demonstrou os trabalhos pioneiros na área de algoritmos animados, onde os trabalhos desenvolvidos (Brown e Sedgewick, 1985) (Stasko, 1990b) (Brown, 1991) possuíam características passivas em sua abordagem, ao contrário da proposta do Core Inc, a solução proposta nesta dissertação. Além disto, estas soluções exigiam um grau de conhecimento elevado no desenvolvimento destas animações, impossibilitando o uso da ferramenta por alunos no início do processo de aprendizado de algoritmos. Mesmo o ANIM (Bentley e Kernighan, 1991) e o trabalho desenvolvido por Carlsen e seus colegas (Carlson et al., 1996) exigem o aprendizado de novos comandos e regras para a construção de uma animação, além de desenvolver o algoritmo que se deseja animar. A proposta de Carlsen (Carlson et al., 1996) de animação através de uma interface amigável também difere da solução proposta nesta dissertação, o Core Inc., pois afasta o aluno de seu próprio algoritmo desenvolvido em sala de aula para reproduzi-lo.

Fator semelhante ocorre a soluções mais recentes propostas na seção 2.3. onde interfaces com melhor usabilidade são fornecidas aos alunos além de soluções com conceitos de programação em blocos (Helminen e Malmi, 2010) (Farrugia e Quigley, 2011), em um destes casos eliminando a necessidade de aprender novos comandos (Goldman, 2004). Ainda assim, a maior parte dos trabalhos de visualização de software exige o aprendizado de novos conceitos e afasta o aluno da utilização do código fonte de seu algoritmo original desenvolvido em sala de aula, adicionando um passo de tradução do material original.

Portanto, os trabalhos encontrados na literatura costumam ter dificuldades em sanar o problema de usabilidade do software por parte dos alunos enquanto o permite utilizar seu algoritmo original desenvolvido em sala de aula. Já a solução proposta no capítulo 3 busca lidar com ambas dificuldades, ao entregar um software com boa usabilidade ao mesmo tempo em que não exige conhecimentos adicionais ou tradução de seu código fonte original para um novo formato ou linguagem.

Outro ponto que deve ser comparado com a solução proposta nesta dissertação é a representação visual dos trabalhos prévios. Enquanto os trabalhos demonstrados na seção 2.2. se enquadram exatamente na área de algoritmos animados, a representação visual dos mesmos utiliza de animações sem contextualização aos alunos, se atendo a representações clássicas de estruturas de dados para apresentar os algoritmos, eventualmente com o auxílio de gráficos e formas geométricas. Este é um fator que a proposta desta dissertação tem um foco diferente, propondo representações visuais que se aproximem mais do cotidiano dos alunos e com um visual mais atrativo ao se utilizar de conceitos de construção de mundo e visuais mais próximos aos videogames.

Esta distância de representações visuais diminui na seção 2.3. que aborda mais trabalhos de visualização de software que não necessariamente são da área de algoritmos animados, apesar de poderem conter elementos dos mesmos. Como é uma área com mais trabalhos recentes, as representações visuais também possuem visuais mais similares ao Core Inc., proposto no capítulo 3. Contudo, somente a representação visual próxima a videogames não é o suficiente, segundo o proposto nesta dissertação, pois entende-se que a representação visual deve ser desenvolvida de acordo com o contexto e cultura dos alunos, como explica a teoria da cognição situada demonstrada na seção 2.5. Assim a solução proposta, Core Inc., difere-se destas em sua parte visual também conceitualmente na construção de suas representações, mesmo que existam trabalhos listados que visualmente tenham propostas muito semelhantes.

Ao olhar todos os trabalhos aqui listados se identificam propostas similares em certas características com o Core Inc., contudo nenhuma delas reúne todas as características propostas nesta dissertação. Alguns trabalhos possuem um foco em algoritmos animados, porém carecem de facilidade de uso por parte dos alunos e representações visuais contextualizadas. Enquanto outros trabalhos possuem melhor usabilidade e visuais mais próximos aos propostos, mas sem o enfoque em algoritmos animados. Além disso nenhuma das soluções propostas levantadas na revisão realizada se preocuparam com a contextualização de suas animações e representações visuais, consequentemente nenhuma delas também foi embasada na teoria da cognição situada. Assim sendo, os trabalhos anteriormente listados podem atender eventualmente alguns pontos propostos por esta dissertação, mas não em sua totalidade e com a união de conceitos teóricos propostos na solução Core Inc..

2.6 EFETIVIDADE DOS ALGORITMOS ANIMADOS NO ENSINO

É notável que o desenvolvimento de pesquisas sobre algoritmos animados reduziu de forma acentuada ao longo dos últimos anos, sendo vários dos trabalhos mais relevantes na área datados anteriormente aos anos 2000. Ao realizar um meta-estudo sobre a efetividade dos estudos na área, Hundhausen e demais (Hundhausen et al., 2002) levantaram alguns dos motivos mais comuns para a redução no número de trabalhos, sendo estes:

- Os professores não acham que possuem tempo suficiente para aprender o necessário para desenvolver uma solução de algoritmo animado;

- Os professores consideram que a abordagem tomaria tempo que é necessário para outras atividades em sala de aula;
- Os professores consideram que criar uma animação para a aula requer muito tempo e esforço;
- Os educadores não consideram a abordagem efetiva.

As três primeiras razões apontadas são inerentes dos tipos de implementação propostas até a publicação do meta-estudo. Como demonstrado na seção 2.2, os trabalhos desenvolvidos eram custosos e complexos ao se tratar de implementação das animações, na maioria das vezes impedindo o próprio aluno de desenvolvê-las, deixando isto a cargo do educador.

No entanto, a última motivação é de caráter diferente. Dúvidas sobre a efetividade da animação de algoritmos foram levantadas, como Hundhausen e colegas demonstraram no gráfico da figura 2.17, onde 46% dos estudos por eles catalogados até o período de sua pesquisa não apresentaram resultados positivos e significantes que demonstrassem a efetividade das soluções propostas. É necessário ressaltar que o meta-estudo não se atém a qualidade e acurácia dos dados levantados por cada estudo, assumindo-se, portanto, que os 24 estão corretos.

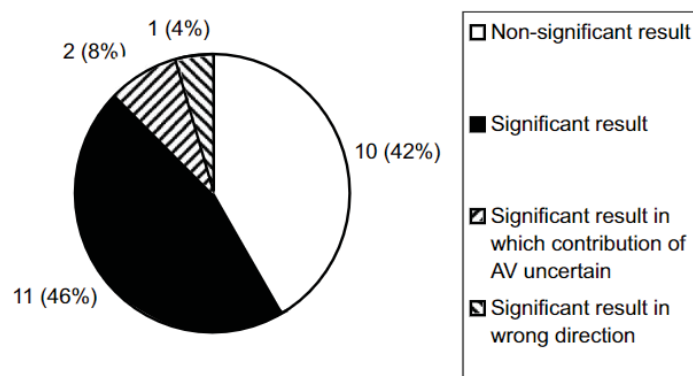


Figura 2.18: Efetividade das Soluções Catalogadas. (Hundhausen et al., 2002)

Ao observar tais números, é aceitável duvidar da efetividade dos algoritmos animados, mas também é importante notar como dentre estes projetos as abordagens e métodos diferem-se entre si. Um projeto realizado com algoritmos animados pode ser aplicado de maneiras diferentes, como já visto com as abordagens passiva e ativa na seção 2.1. Assim como pode ser desenvolvido em momentos diferentes do processo educacional, como laboratórios, estudos individuais, tarefas, palestras e afins. Cada uma dessas aplicações pode indicar resultados diferentes no ensino.

Verificando a efetividade dos diferentes métodos de aplicação dos algoritmos animados, Lawrance e colegas (Lawrance et al., 1994) realizaram um estudo empírico com alunos de ciência da computação do Instituto de Tecnologia da Georgia. Ao todo 62 estudantes foram divididos em grupos, buscando avaliar as diferenças entre os métodos de aplicação dos algoritmos animados. Ao utilizar-se de alguns software desenvolvidos no período, entre eles o XTango, uma evolução do já citado Tango (Stasko, 1990b), os alunos foram divididos de forma a receber o método de ensino tradicional com slides e palestras, o método de ensino passivo, com a utilização dos software de algoritmos animados através da visualização de animações prontas e a abordagem ativa, quando os alunos efetivamente interagiam com o software de animação dos algoritmos.

No fim do experimento, quando testes foram aplicados para todos os grupos, o estudo concluiu que os alunos que mais interagiram com o algoritmo animado se saíram melhor que

os demais. Separando-se aqueles com aprendizado ativo, que interagiram diretamente com as estruturas da animação, e receberam notas melhores, daqueles com ação passiva, que, ao somente assistir as animações, ainda obtiveram uma melhora em relação ao ensino tradicional, mas não tão efetiva. Isto demonstra uma tendência a trabalhos de algoritmos animados com características ativas terem mais sucesso ao impactar o desempenho dos alunos positivamente.

Outro estudo buscando evidências de diferenças entre solução passivas e ativas foi realizado através de ferramentas de animação de grafos (Taylor et al., 2009). Ao aplicar-se um software que permitia interação ativa entre algoritmos desenvolvidos e soluções com animações previamente desenvolvidas, os autores encontraram resultados que demonstraram, através de questões respondidas pelos grupos, que os alunos que fizeram uso da abordagem ativa apresentaram melhor desempenho que os que fizeram uso das animações passivas.

Não obstante, teorias educacionais diferentes foram encontradas por Hundhausen e os demais autores no desenvolvimento de cada solução, sendo elas:

- *Fidelidade Epistêmica*: teoria em que se assume que humanos carregam em suas mentes modelos simbólicos do mundo físico e esses modelos são a base para sua cognição e ações;
- *Codificação Dupla*: uma teoria que entende a cognição como a atividade de dois sistemas simbólicos independentes e parcialmente conectados. Uma parte processa informações verbais e a outra não verbais;
- *Diferenças Individuais*: no contexto de visualização de software esta teoria diz que diferenças mensuráveis em habilidades pessoais e estilos vão levar a diferentes performances no uso de soluções de visualização de software;
- *Construtivismo Cognitivo*: esta teoria afirma que não há conhecimento absoluto, pois os indivíduos constroem seu próprio conhecimento através de experiências subjetivas no mundo.

Dentre estas, os resultados obtidos sobre a efetividade de cada teoria demonstraram que o Construtivismo Cognitivo foi mais efetivo que as demais como é possível observar na Figura 2.18.

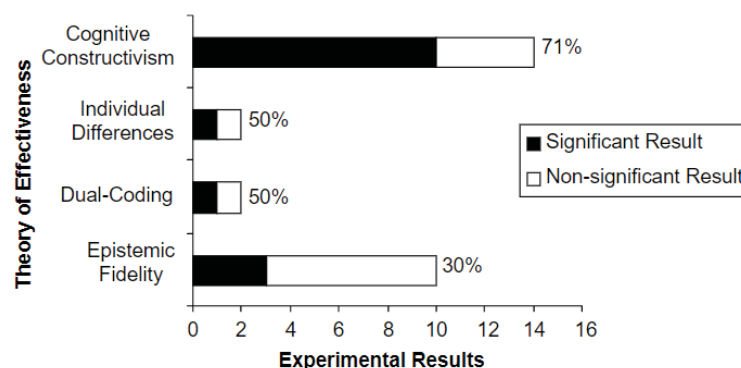


Figura 2.19: Efetividade dos Softwares por Teorias. (Hundhausen et al., 2002)

Quanto aos métodos de avaliação utilizados pelos trabalhos em questão, 22 optaram por aplicar questionários nas turmas que participaram do experimento, utilizando ou não os software. Parte das pesquisas aplicaram antes do experimento e posteriormente o mesmo, enquanto alguns

realizaram o teste somente ao fim do experimento. Dentre estas duas abordagens, a Figura 2.19 demonstra que a diferença no período de aplicação das avaliações apresenta resultados divergentes quanto a significância dos softwares aplicados ao influenciar no desempenho dos alunos.

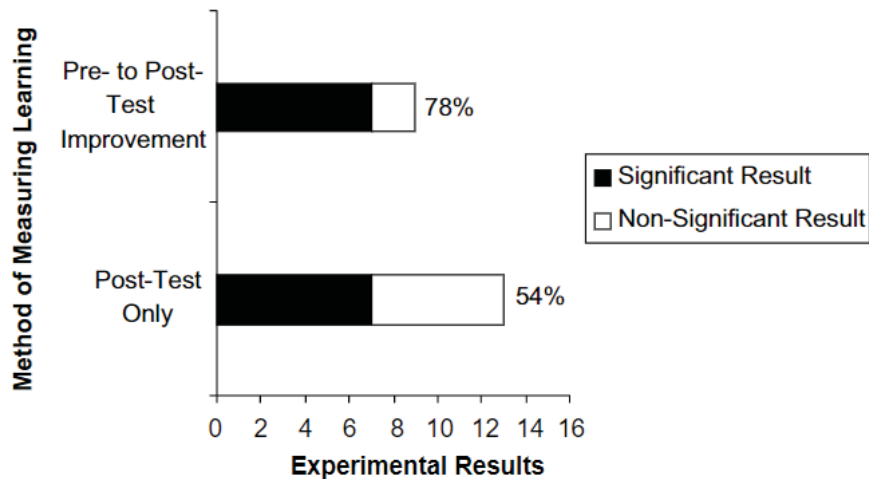


Figura 2.20: Efetividade dos Softwares por Métodos de Avaliação. (Hundhausen et al., 2002)

Com estes dados os autores explicam que é possível que a aplicação de testes antes e depois do experimento seja realmente mais efetiva, supondo-se que os softwares realmente causem impacto nos alunos. Contudo, seria precipitado afirmar isto com uma quantidade de dados tão reduzida, assim como fazer qualquer afirmação, como a coletada pelos educadores e demonstrada no começo da sessão.

O que os estudos demonstram quanto a avaliação da efetividade dos algoritmos animados é, sim, que a teoria do cognitivismo construtivista é a mais aceita na área, porém os demais resultados não parecem ser capazes de afirmar contundentemente a efetividade dos algoritmos animados no processo de ensino-aprendizagem. Entretanto, Hundhausen e seus colegas concluíram que a correlação do nível de engajamento dos alunos com as animações e os melhores resultados em sala de aula existe, e propostas dentro da área de algoritmos animados e visualização de algoritmos precisam engajar os alunos ativamente no processo, capturando sua atenção.

Estudos que abordam o engajamento dos alunos ganharam força, como revisões de literatura ao longo da última década demonstram (Shaffer et al., 2010) (Al-Sakkaf et al., 2019). A revisão mais recente (Al-Sakkaf et al., 2019) faz relações diretas do engajamento com a melhora do aprendizado e motivação dos estudantes segundo a literatura mais recente. Junto disto, os softwares de visualização parecem carecer desta capacidade de engajamento, consequentemente afetando o resultado final do aprendizado, pois quando engajados o aprendizado dos alunos é melhor (Hosseini et al., 2016).

Tendo como base a tese de que as tecnologias de visualização teriam pouco valor educacional se não se atentassem ao engajamento ativo dos alunos no processo, Naps e demais pesquisadores (Naps et al., 2002) chegaram a desenvolver uma nova taxonomia. Denominada Taxonomia de Engajamento, seu objetivo era definir o grau de interatividade do software com o aluno, através de 6 níveis diferentes, sendo estes: sem visualização, visualizando, respondendo, mudando, construindo e apresentando. Os desenvolvedores da teoria acreditam que quanto mais engajamento houvesse, e de formas diferentes ao longo do processo, melhores seriam os resultados do aprendizado dos alunos. Fora esta taxonomia, outras surgiram ao longo dos

anos, como a Taxonomia de Engajamento Estendida (Myller et al., 2009) que propõe uma versão mais abrangente da Taxonomia de Engajamento original, e a Taxonomia De Engajamento Bidimensional (Sorva et al., 2013), inserindo uma nova perspectiva as taxonomias anteriores, considerando a sensação e pertencimento do aluno em relação ao conteúdo do software, como quando ele mesmo constrói a animação durante o processo.

Segundo os 58 artigos reunidos por Al-Sakkaf e seus colegas (Al-Sakkaf et al., 2019), a taxonomia mais utilizada nos últimos anos ainda é a original (Naps et al., 2002). Dividindo espaço com trabalhos que se fundamentam nas teorias educacionais construtivistas, que como o meta-estudo (Hundhausen et al., 2002) afirmou uma década mais cedo, ainda são predominantes nesta área de estudo. Junto disto, teorias da área da psicologia relacionadas a motivação também ganharam espaço, ao abordar o engajamento como o objetivo principal destes software.

Além disto, a última revisão da literatura (Al-Sakkaf et al., 2019) estabelece atributos que se relacionam diretamente com o engajamento do software, dando uma perspectiva de quais pontos um software de visualização deve se atentar ao buscar o engajamento dos alunos, sendo eles:

- *Estética*: referindo-se ao visual do software e seus componentes;
- *Apelo Afetivo*: a reação emocional em relação ao software, que encoraja o aprendizado. Diz respeito também ao quão imerso o aluno está no software;
- *Desafio*: O desafio das atividades deve ser adaptado as capacidades do estudantes;
- *Feedback Imediato*: O feedback é um fator essencial para manter a atenção do aluno;
- *Motivação Intrínseca*: Os estudantes se tornam intrinsecamente motivados quando o software proporciona através de seu ambiente competência, autonomia, empatia e propósito;
- *Motivação Extrínseca*: Definida pela performance externa ao software, como a pontuação e uma avaliação posterior;
- *Controle Percebido*: Quando o estudante tem a possibilidade de decidir como interagir com o sistema;
- *Baixa Carga Cognitiva*: Garantir que o estudante não seja sobrecarregado de informações, o que pode causar ele a abandonar o software ou sistema;
- *Interação Social*: Permitir que os estudantes interajam entre si durante o aprendizado.

Por fim, Al Sakkaf e seus colegas afirmam a importância de que os projetos na área de visualização de software estejam bem fundamentados pedagogicamente, já que há uma correlação direta entre esta fundamentação e o engajamento final dos alunos.

2.7 PIXEL ART

A parte visual de um jogo, seja ele educacional ou não, pode ser abordada de diversas formas, seja através de 3 ou 2 dimensões, estilos de desenhos e animações diferentes ou tecnologias que são empregadas. Dentre estes meios, uma técnica de ilustração e animação existente é a Pixel Art: um tipo de ilustração em duas dimensões em que cada pixel inserido é visível,

diferindo-se até mesmo de imagens digitais em baixa resolução, pois nesta técnica cada pixel na tela é colocado intencionalmente em sua posição (Silber, 2016).

Utilizado em jogos das décadas de 1980 e 1990 a princípio, a técnica de ilustração ganhou espaço pelas facilidades técnicas de produção e tamanho reduzido dos arquivos, que se encaixava nas limitações técnicas do período. Atualmente, apesar do poder de processamento abundante e muitas outras técnicas de ilustração no mercado, a Pixel Art ainda se mantém relevante. Ilustrações e animações construídas com a técnica são atemporais, pois não perdem qualidade ao longo do tempo (Silber, 2016).

Alguns jogos recentes que utilizam tais técnicas são: Ikenfell na Figura 2.21, Hyper Light Drifter, Undertale e Blaphemous.



Figura 2.21: Ikenfell.(Games, 2020)

2.8 MODELO DE ACEITAÇÃO DE TECNOLOGIA - TAM

Nas últimas décadas o crescimento da área de tecnologia da informação e informática passou a influenciar significativamente o dia a dia de organizações com a inserção de software nestes ambientes, seja no trabalho, estudos, pesquisas e afins. Com isto, a partir da década de 1990 pesquisas na área de avaliação da receptividade de software se intensificaram (da Silva et al., 2009).

Apesar de os avanços tecnológicos dessas novas tecnologias poderem apresentar benefícios aos processos que ocorrem nestes ambientes, é um desafio definir a qualidade do software desenvolvido. Mesmo quando boas práticas de desenvolvimento são seguidas, é difícil medir a eficácia dos sistemas entregues e a percepção dos usuários em relação ao software desenvolvido. Portanto, compreender e ser capaz de medir a aceitação destes software é um desafio.

Dentre as propostas que surgiram ao longo dos anos, Davis (Davis, 1989) desenvolveu o *Technology Acceptance Model* (TAM), também conhecido como Modelo de Aceitação de Tecnologia. (Davis, 1989) que se tornou um dos mais aceitos e utilizados no meio (Dillon e Morris, 1996) (da Silva et al., 2009). Proposto em 1989, ele é uma adaptação de um modelo anterior chamado de Teoria da Ação Racionada (TRA), e busca obter medidas que possam prever e explicar a utilização de um novo sistema pelo ponto de vista do usuário (Davis, 1989). Para estabelecer estas medidas o método TAM estabelece construtos que são responsáveis pela

percepção do usuário em relação ao software. O diagrama do modelo demonstrado na figura 2.22 possui estes construtos e suas relações.

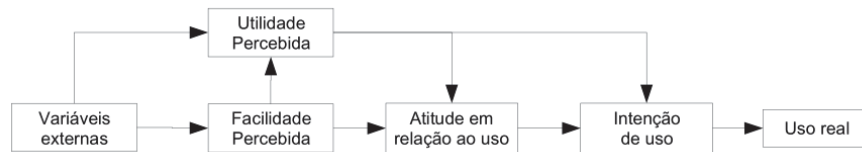


Figura 2.22: Modelo de Aceitação de Tecnologia (Saleh, 2004)

Utilidade Percebida é o grau em que cada pessoa acredita que o sistema proposto irá facilitar seu trabalho e melhorar seu desempenho. A utilidade acaba por ser relevante pois melhor desempenho no trabalho costuma ser uma boa prática que proporciona benefícios futuros as pessoas (Davis, 1989).

Facilidade Percebida se refere ao grau que a pessoa considera que utilizar o sistema é livre de esforço. Esforço, neste caso, seja recursos desempenhados para utilizar o software, sejam estes físicos, mentais ou morais. Neste caso, sistemas com facilidade percebida maior tendem a serem melhor aceitos pelos usuários (Saleh, 2004).

Segundo os experimentos de Davis a utilidade e facilidade percebida influenciam na *atitude em relação ao uso* do software, que irá definir se o sistema será bem aceito ou não, pois a *Intenção de Uso* é diretamente influenciada pela atitude e é a característica fundamental para definir se o sistema será usado e bem aceito.

Uma extensão do modelo de aceitação de tecnologia foi proposta nos anos seguintes. Segundo os autores (Mathieson et al., 2001) existem situações em que uma série de recursos disponíveis, ou não, influenciam na intenção de uso dos sistemas, pois podem restringir as possibilidades de utilização pelo usuário. O novo modelo proposto é demonstrado na figura 2.23.

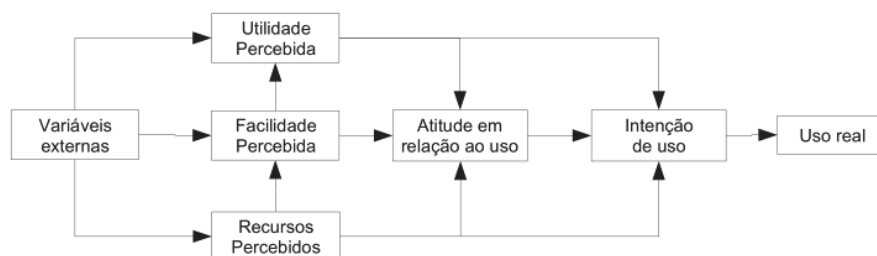


Figura 2.23: Extensão do Modelo de Aceitação de Tecnologia (Saleh, 2004)

Neste modelo é adicionado o construto *Recursos Percebidos*, que representa uma lista de elementos que podem ser considerados pelo usuário recursos tanto pessoais quanto organizacionais que influenciam em sua atitude e intenção e uso. Entre estes estão habilidades pessoais, hardware, software, dados, documentação, assistência humana e tempo (Mathieson et al., 2001).

Por fim, este é um modelo comportamental, podendo abordar somente sobre questões relacionadas a percepção do usuário em relação ao uso do sistema. Os construtos desenvolvidos devem sempre buscar obter opiniões pessoais em relação ao software, mesmo quando estas não sejam diretamente um indivíduo em específico, como o caso de instituições ou terceiros (Saleh, 2004).

2.9 CONSIDERAÇÕES

Na Seção 2.3 foi apresentado o conceito de algoritmo animado e a motivação por trás do começo dos estudos na área. Estes que foram motivados pela alta capacidade de processamento que as máquinas tinham na época. Essa capacidade de processamento gráfico é ainda mais evidente hoje em dia com tantas tecnologias voltadas ao processamento gráfico, como os jogos o fazem.

Neste período, onde várias tecnologias surgiam no âmbito visual, soluções diferentes despontaram tentando fazer uso de elementos visuais para o ensino de computação, contudo só uma pequena área se atribuiu aos algoritmos, principalmente os algoritmos animados. Definindo-se os algoritmos animados e suas possibilidades, uma gama de possibilidades se abriu, com capacidades promissoras de auxiliar alunos e programadores a entender, consertar e desenvolver algoritmos.

Diversas abordagens surgiram ao longo dos anos e preocupações com os métodos de aplicação da tecnologia, seja ela em uso ativo ou passivo. Além disso a qualidade das animações e transições também foi colocada em pauta, com paradigmas de desenvolvimento de linguagens animadas sendo desenvolvidos, em busca de facilitar o desenvolvimento de animações enquanto as deixavam mais eficientes e suaves.

Contudo, quando observamos os software já desenvolvidos na Seção 2.4, é evidente a diferença entre os anseios dos desenvolvedores em relação as características do software e a realidade do período. Apesar de desejar-se animações suaves e claras, que demonstrassem não só as estruturas de dados como a atuação da semântica do algoritmo, as tecnologias e capacidade das máquinas da época não atingiam o potencial pleno desejado. Apesar de paradigmas terem sido criados para o desenvolvimento mais ágil de animações mais suaves (Stasko, 1990b), a sua aplicação ainda era difícil. Os sistemas possuíam gráficos simples que se utilizavam de formas geométricas, vários deles incapazes de atualizar-se em tempo real, e aqueles que o faziam, necessitava de momentos específicos para atualização da animação.

Questões técnicas acabavam por limitar os software a representações clássicas das estruturas de dados e algoritmos, potencialmente dificultando a obtenção de resultados expressivos. Na seção 2.6 foram acompanhados estudos sobre a eficácia dos algoritmos animados no ensino, que demonstraram a dúvida sobre a eficácia da técnica. Contudo, vários destes trabalhos encontraram barreiras em tópicos que os educadores levantaram como problemas da tecnologia. Estes são a dificuldade e tempo elevado necessário para o desenvolvimento de animações, além da dificuldade de encaixar o conteúdo na grade curricular dos cursos.

Sabendo deste histórico e dos avanços até então feitos, e que somente replicar o que foi feito no passado com melhor qualidade de imagem não seria suficiente, mesmo com os conhecimentos obtidos através dos trabalhos que promovem o engajamento dos estudantes, a seção 2.1 apresenta o conceito da Teoria de Cognição Situada. Esta que não foi utilizada como fundamentação do desenvolvimento de um software de algoritmos animados ou visualização de software na literatura demonstrada. A teoria, ao contrário das abordagens físicas e cognitivas utilizadas pelas soluções predecessoras, considera que a construção do conhecimento se dá pela interação constante do indivíduo com o conhecimento e seu contexto, e apresentou um novo paradigma relevante a ser considerado no processo educacional.

Por outro lado, notou-se a capacidade destas soluções em promover o interesse dos alunos pelos conteúdos e, ao longo dos anos, apesar de uma diminuição de trabalhos específicos de algoritmos animados, a área de visualização de software em geral cresceu, apresentando novas soluções com o enfoque na promoção do interesse dos alunos. Trabalhos mais recentes abordados na seção 2.5 demonstram diversas soluções que mesmo que não se enquadrem nas definições de

algoritmos animados especificamente, demonstram técnicas e soluções novas que avançaram os estudos na área e utilizaram-se de novas tecnologia hoje disponíveis, com conhecimentos que também são relevantes para a área de algoritmos animados especificamente.

Portanto, ao utilizar esta teoria cognitiva como um pilar de uma solução de algoritmos animados e uni-la com a capacidade de contextualização que os visuais de um jogo é capaz de proporcionar, é possível abordar os algoritmos animados de um ponto de vista que combina a representação visual e teoria cognitiva ainda inédita para a área, buscando atingir resultados melhores do que as pesquisas de algoritmos animados realizadas previamente.

Esta abordagem a ser demonstrada no capítulo 3 trata-se de interagir com os conhecimentos prévios de mundo do aluno e seu dia a dia ao transformar os conceitos de algoritmos e programação em um mundo fictício, mais familiar e próximo aos seus conhecimentos, utilizando de elementos da cultura e dia a dia destes alunos na representação das animações.

Já visando atingir este objetivo, a conceituação da arquitetura de *von Neumann* é importante, pois embasa os fundamentos da execução dos algoritmos e consequentemente qualquer contextualização que venha a ser proposta durante o processo da animação dos algoritmos. Para desenvolver esse processo de animação, o breve histórico e aplicação da técnica de pixel art é importante, pois não só demonstra como o método de ilustração funciona, como também demonstra o quão atual ele ainda permanece, apesar de seus visuais aparentemente datados. Este fator que é importante quando se trata de manter a atenção do aluno.

Após este levantamento, é reconhecido desde cedo o potencial dos algoritmos animados. A possibilidade de conectá-lo com uma nova abordagem científico cognitiva e representação visual prática e atual é promissora o suficiente para que seja desenvolvida e avaliada, de forma a talvez atingir resultados melhores do que os apresentados até o momento na literatura.

3 DESENVOLVIMENTO DA SOLUÇÃO DE ALGORITMO ANIMADO

Após o levantamento realizado no capítulo 2 dos conceitos relevantes para esta dissertação, é possível encaminhar-se para o que é a proposta de solução do problema, ou seja, como usar de algoritmos animados de forma contextualizada para melhorar o processo de ensino-aprendizagem de algoritmos e programação. Portanto, este capítulo aborda a proposta de solução de algoritmos animados embasada na teoria da cognição situada que tem o nome de Core Inc. e seu processo de desenvolvimento, com diagramas e documentos gerados no desenvolvimento do Core Inc., além das telas e funcionalidades mais relevantes.

3.1 CORE INC.

O Core Inc. é um software de algoritmos animados que usa como base para sua construção a teoria da cognição situada demonstrada na seção 2.5 e voltado para o uso no ambiente de sala de aula, com alunos que estão nas primeiras semanas de aula de algoritmos e programação. Desenvolvido de maneira a seguir os preceitos da teoria cognitiva selecionada e se utilizando de características visuais de jogos digitais o Core Inc. tem como objetivo apresentar aos estudantes nos estágios iniciais do aprendizado de algoritmos e programação o funcionamento dos algoritmos desenvolvidos em sala de aula de acordo com o ciclo de instrução da arquitetura de *von Neumann*, esta apresentada na seção 2.6. Arquitetura esta que foi considerada como base fundamental da solução pela importância nas disciplinas de algoritmos ministradas para alunos que estão ingressando no curso de ciência da computação da Universidade Federal do Paraná (UFPR) nas disciplinas de algoritmos e programação e pela aplicação ainda atual de seu ciclo de instrução em máquinas contemporâneas.

Para buscar alcançar uma proposta que interaja com o conhecimento prévio de mundo do aluno e o seu dia a dia o Core Inc. transforma os conceitos de algoritmos e da máquina de *von Neumann* em um mundo fictício, onde o computador foi transformado em uma empresa de nome Core Inc. como demonstrado na logo da figura 3.1 e um desenho da a mesma na figura 3.2, cada componente da arquitetura foi antropomorfizado em um funcionário desta empresa. Estes personagens tomam inspirações em elementos do dia a dia dos alunos e da cultura a sua volta, com referências diretas a filmes, seriados, desenhos e afins.

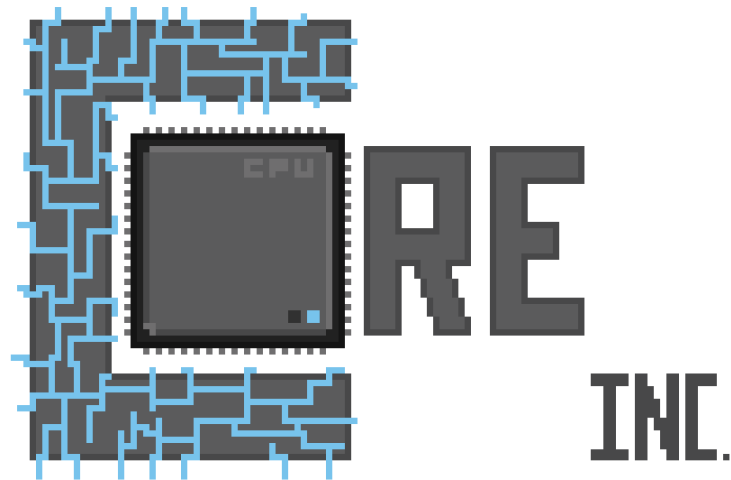


Figura 3.1: Logo Core Inc. O Autor

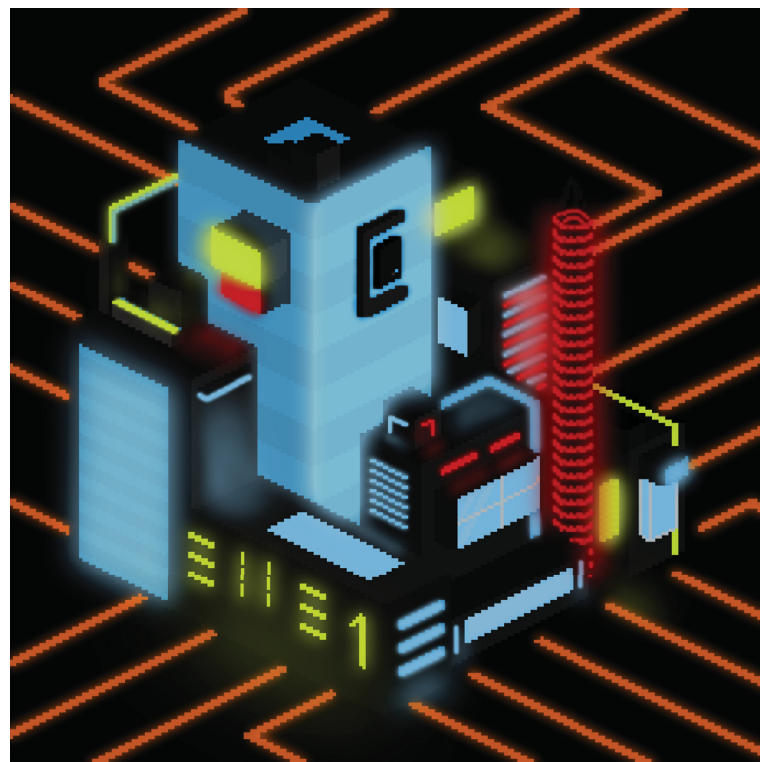


Figura 3.2: Ilustração Empresa Core Inc. O Autor

O elenco de personagens do Core Inc. é composto de quatro personagens que representam componentes da arquitetura de *von Neumann*, sendo eles: Unidade de Controle, Barramento de Memória, Unidade Lógica e Aritmética, Barramento de Entrada/Saída de Dados. A personagem Lucy, que representa a unidade de controle é demonstrada na figura 3.3 com seu nome, história, inspirações e representação visual em forma de retrato e animação, seguida da apresentação da personagem Úrsula na figura 3.4, esta uma antropomorfização da Unidade Lógica e Aritmética.

Na figura 3.5 o personagem M3M0 retrata o barramento de memória e por último na figura 3.6 o personagem Bus representa o barramento de entrada e saída de dados.

LUCY

Unidade de Controle

Core Inc.



História

Lucy foi a primeira na sua turma de cadetes no departamento de Unidades de Controle. Um prodígio na empresa, Lucy é capaz de gerenciar unidades de processamento de grandes capacidades com agilidade sem igual. Suas qualidades junto de sua ambição a tornou a candidata ideal para coordenar a CPU do programa de Auditoria do Usuário.

Características

Lider Nata
Determinada
Responsável

Visual na Animação



Referências



EDI – Mass Effect 2



Beth – The Queen's Gambit



Pertisia – Ikenfell

Figura 3.3: Personagem Lucy. O Autor

URSULA

Unidade Lógica e Aritmética

Core Inc.



História

Ursula desmonstrava aptidão aos números desde que saiu do berço, aprendeu a resolver a equação de Bhaskara antes mesmo de falar. Pouco antes de passar no exame de admissão das Unidades Lógicas e Aritméticas desenvolveu seus braços, apelidados carinhosamente de **Bits**. Tímida, ela não sai muito de sua estação, mas calcula mais rápido do que qualquer outra ULA da **Core Inc.**

Características

Tímida
Inteligente
Sensitiva

Visual na Animação

Observações:
Os braços superiores se esticam, buscando o conteúdo dos registradores.



Referências



Úrsula – A Pequena Sereia



Margaret Hamilton



Olivia Octavius – Aranhaverso

Figura 3.4: Personagem Úrsula. O Autor

M3M0

Barramento de Memória

Core Inc.



História

M3M0 Foi criado em uma família tradicional dentro da Core Inc. Seus ancestrais já faziam o controle de memórias desde os primeiros anos de empresa. Seguindo os passos de seu predecessores, M3M0 assume com orgulho e espantosa eficiência o mesmo cargo. Seu perfil trabalhador e exemplar fez dele a opção ideal para o programa de Auditoria do Usuário, mesmo que ainda tão novo.

Características

Confiável
Metódico
Eficaz

Visual na Animação



Referências



O Gigante de Ferro



Julius – Todo Mundo Odeia o Chris



Robô – Três Robos

Figura 3.5: Personagem M3M0. O Autor

Bus

Barramento Entrada/Saída

Core Inc.



História

Bus é uma nova tecnologia quando se trata de transporte de dados. Oriundo dos estudos em animais de estimação robóticos descobriu-se que o pássaro é muito mais rápido e forte do que os funcionários que trabalhavam no departamento de barramentos. Contudo, sua imprevisibilidade causa temores, sua posição no programa de **Auditoria do Cliente** é o primeiro teste de **Bus** para ocupar uma vaga fixa na **Core Inc.**

Características

Veloz
 Forte
 Imprevisível

Visual na Animação



Referências



Família Blue – Rio 2



Arara Azul

Figura 3.6: Personagem Bus. O Autor

3.2 CONSTRUÇÃO DA CONTEXTUALIZAÇÃO

O primeiro passo no desenvolvimento do Core Inc. foi a decisão de como os elementos de jogos e a contextualização seriam utilizados para representar o ciclo de instrução de *von*

Neumann através das animações do software de algoritmos animados. Portanto, esta seção é responsável por demonstrar o primeiro passo no desenvolvimento da solução de algoritmos animados contextualizada, que é a definição de como sua contextualização foi desenvolvida até chegar ao mundo e personagens apresentados na seção 3.1.

Para tornar o aluno parte do contexto do conteúdo é necessário inseri-lo no mesmo universo que este, estabelecendo uma relação social com o seu dia-a-dia e cultura. É necessário que ele acredite fazer parte de seu contexto social e que as funcionalidades que ele possua à disposição, sejam as ferramentas para que o aluno interaja com este contexto. Neste caso os comandos da linguagem de programação e os componentes que compõe a arquitetura de *von Neumann* são as ferramentas de interação.

Inicialmente, entendeu-se que aplicar um contexto ao ensino de algoritmos que apresente pontos em comum entre todas as instruções que serão abordadas pela proposta não é trivial, ainda mais fazê-lo utilizando somente cenários realistas, até porque tais situações nem mesmo são conhecidas por estudantes de primeiro período destes cursos, que acabaram de entrar em sala de aula e estão familiarizando-se com o desenvolvimento de software.

Para que o ambiente proposto pelo Core Inc. possa se integrar de maneira natural ao contexto do aluno é necessário que haja uma relação com os conceitos do mundo real em que o aluno está familiarizado. Portanto, o objetivo foi estabelecer um contexto fictício que promova o interesse do aluno pelo Core Inc. ao mesmo tempo que possua características suficientes do contexto atual dos alunos em seu cotidiano e cultura.

Optou-se por utilizar a abordagem que os jogos utilizam comumente para contextualizar suas mecânicas, que no caso da solução desta dissertação são as animações dos algoritmos. Este método parte do princípio que um mundo com locações e personagens fictícios é criado, justificando as ações do jogador dentro daquele mundo.

Contudo, somente criar um mundo fictício sem o mínimo de cuidado com sua história pode afastar o aluno, seja pelo mundo ser desinteressante ou até mesmo estranho demais, tornando-o não relacionável e afetando diretamente o objetivo de contextualizar a animação.

Uma técnica para manter o equilíbrio na criação desses mundos é o Triângulo da Esquisitice (Rogers, 2016) da figura 3.7. Rogers desenvolveu este triângulo com o objetivo de controlar o quão estranho a história de um jogo pode se tornar, algumas vezes tão estranha ao ponto de confundir o público e fazer com que ele perca o interesse. Para evitar que a história se torne tão estranha ou esquisita, como diz o nome do triângulo, são definidos 3 pilares que devem ser levados em conta:

- **Os personagens** da história;
- **As atividades** que esses personagens realizam;
- **O mundo** em que estes personagens se encontram.

Segundo Rogers, o importante é que sua história nunca subverta para longe da realidade mais de um destes vértices, pois corre-se o risco de alienar o público. Ele cita exemplos de histórias que seguem esta regra ao longo dos anos, como *O Mágico de Oz*, que subverte seus personagens e visuais, mas suas motivações, emoções e o cenário de fantasia já era comum e próximo a realidade dos leitores mesmo em 1900.

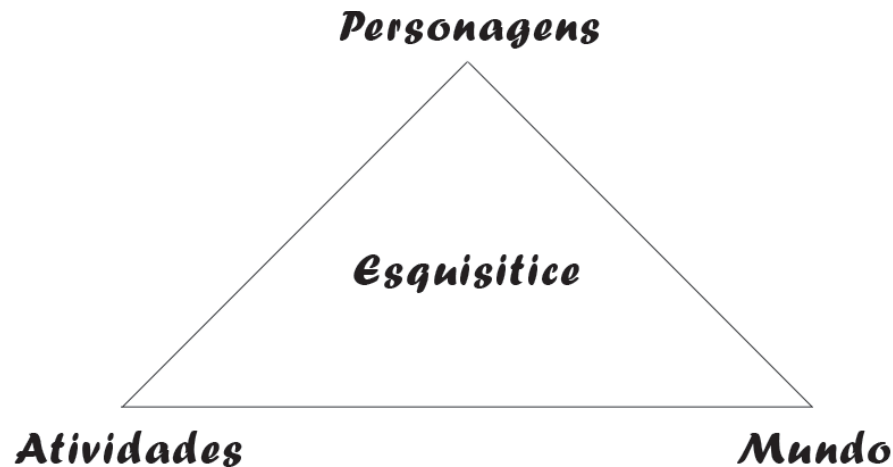


Figura 3.7: Triângulo da Esquisitice (Rogers, 2016)

Obedecendo o triângulo da esquisitice de Rogers, a seguinte narrativa foi criada com o objetivo de contextualizar a animação de algoritmos:

- *Mundo*: A empresa Core Inc. é localizada dentro de um mundo em que os componentes do computador foram antropomorfizados, portanto, seus funcionários executam a tarefa de receber instruções de uma entidade chamada Cliente, este sendo o usuário, e executá-las.
- *Personagens*: A Core Inc. é composta por quatro funcionários. Cada um deles fazendo alusão direta aos seguintes componentes: Unidade de Controle, Unidade Lógica e Aritmética, Barramento de Memória, Barramento de Entrada e Saída de Dados.
- *Atividades*: Estes personagens irão executar as ordens de seu cliente de acordo com o ciclo de execução de instrução de *von Neumann*.

Portanto, o mundo criado para contextualizar as animações é o que foi subvertido, criando uma realidade fictícia dentro da unidade central de processamento da máquina, responsável pela execução do algoritmo e saindo de uma realidade próxima dos alunos quando veem esse conteúdo em sala de aula. Porém, ainda que subvertido, uma preocupação adicional foi tomada na decisão da representação desta fábrica, para que a relação dos alunos com a tela da animação seja o mais familiar possível. Portanto, decidiu-se por manter a animação somente com uma tela, representando concisamente todas as partes da fábrica, como demonstrado na figura 3.8.



Figura 3.8: Cenário Core Inc.

A caracterização da fábrica estabelece elementos que façam sentido tanto no contexto do Core Inc. quanto no mundo real e contexto cultural dos alunos, fazendo alusão a itens do cotidiano ou familiares em nossa cultura, como esteiras, mesas, pacotes de entrega e computadores. A relação entre os desenhos em pixel art e os reais é demonstrado na figura 3.9.

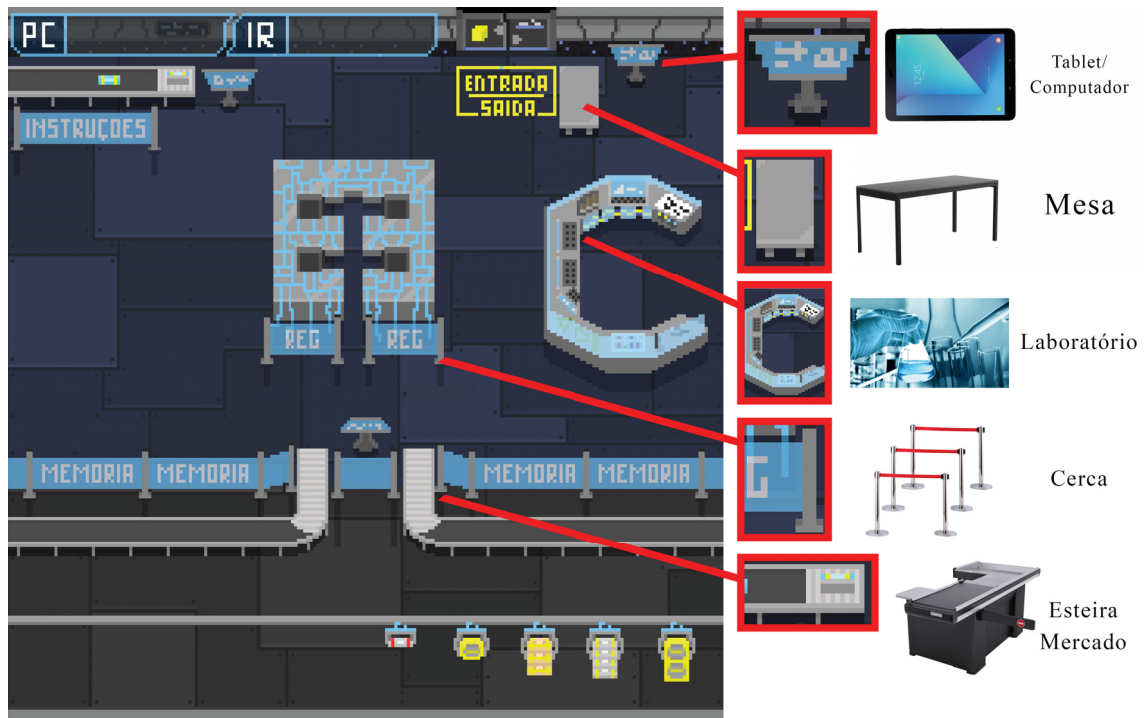


Figura 3.9: Relações de Objetos com Contexto

Além disso, os pacotes de entrega localizados na parte inferior da tela, onde é localizada a memória, possuem tamanhos e formatos diferentes de acordo com o tipo de variável declarada, como é detalhado na figura 3.10. Neste caso levou-se em conta no design o tipo de dado que cada variável armazena e o espaço em bits que cada uma ocupa em memória.

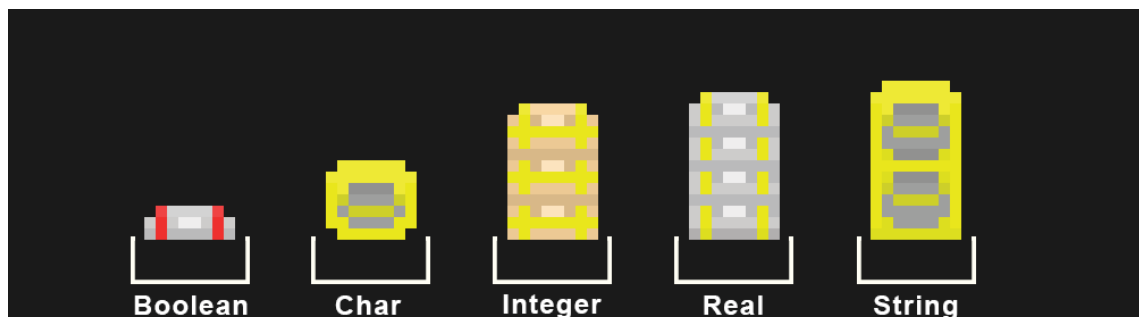


Figura 3.10: Tipos de Pacotes de Entrega (Variáveis)

Já os personagens da fábrica fazem alusão direta a elementos previamente conhecidos conceitualmente pelos alunos quando apresentados à arquitetura de *von Neumann* previamente na disciplina. Ainda levando em conta a contextualização destes personagens, a caracterização de cada um deles faz referências a personagens e conceitos familiares ao dia a dia dos alunos, com aparências e personalidades familiares a obras audiovisuais populares, como robôs, pessoas e personagens famosos, como foi demonstrado nos perfis dos personagens na seção 3.1.

Por último, as atividades executadas neste ambiente e por estes personagens não é estranha ou fora de contextualização ao ambiente da animação e do conteúdo em sala de aula, devido ao fato de que estas serão a execução do próprio algoritmo que o aluno elaborou, e sua demonstração é dada seguindo os sete passos do ciclo de instrução de *von Neumann*.

Conhecendo o contexto para a animação, agora é possível descrever como as funcionalidades ao redor da animação interagem com a mesma, quais são seus requisitos funcionais e não

funcionais, quais são os comandos a serem animados e como as próprias animações funcionam passo a passo para cada tipo de instrução.

3.3 ANÁLISE ESSENCIAL

Os requisitos definidos foram estabelecidos para atender as expectativas de um software de algoritmos animados segundo a literatura pesquisada no capítulo anterior, habilitando o Core Inc. a executar estas funções essenciais e reforçar as características que funcionaram em outros softwares correlatos enquanto minimiza os problemas encontrados na literatura.

Os requisitos da tabela 3.1 são essenciais para caracterizar o Core Inc. como um software de algoritmos animados, pois definem a animação de um programa ou código fonte em específico, ao possibilitar o aluno digitar de forma autônoma seu próprio código para ser animado, sem que seja necessária uma capacitação extensa para a utilização do software, reduzindo a necessidade de assistência constante do professor.

Buscando as características ativas das abordagens de algoritmos animados, capazes de obter melhores resultados em sala de aula, os requisitos da tabela 3.2 foram estabelecidos para permitir interatividade constante do aluno com a animação, permitindo que o aluno leve o tempo que for necessário para observar a fazer anotações com as animações, ao mesmo tempo que obtém informações adicionais dentro da animação caso interaja com a mesma, incentivando a interação.

O último requisito funcional da tabela 3.3 é essencial para a solução devido as características de contextualização que o Core Inc. propõe. Este requisito introduz inicialmente ao contexto do software, dando uma explicação prévia dos elementos fictícios que serão apresentados na animação posteriormente, enquanto busca promover o interesse do usuário exibindo o mundo, personagens e atividades criados na seção anterior.

Abaixo são listados os requisitos funcionais justificados junto de suas características não funcionais:

Tabela 3.1: Codificar

F1 – Codificar	
Descrição: O usuário deverá ser capaz de digitar seu código em um campo texto dedicado para esta função.	
Requisitos Não Funcionais	
Nome	Restrição
NF 1.1 – Suporte para Web	O campo de texto será compatível com as tecnologias mais básicas WEB (html, css, javascript).
NF 1.2 – Linguagem suportada	É esperado que os códigos digitados sejam na linguagem Pascal.
NF 1.3 – Extensão por Tag	Ao chamar a tag no código html da página web o campo de texto aparece.

Tabela 3.2: Escolher Animação

F2 – Escolher Animação	
Descrição: O usuário deverá ser capaz de marcar as linhas que deseja ver animadas ou ver o código completo animado.	
Requisitos Não Funcionais	
Nome	Restrição
NF 2.1 – Ícone de marcação	As linhas marcadas para serem animadas devem receber um círculo azul na lateral esquerda, antes do código da respectiva linha.
NF 2.2 – Animação Completa	No topo da caixa de texto um botão azul com a opção de animar o código deve estar disponível.

Tabela 3.3: Pausar Animação

F3 – Pausar Animação	
Descrição: O usuário poderá pausar a animação em certos momentos através do mesmo botão que executa a animação ou passando o mouse por cima da animação.	
Requisitos Não Funcionais	
Nome	Restrição
NF 3.1 – Cor do botão	O botão deve ser da cor vermelha.
NF 3.2 – Texto do botão	O botão de Executar deve trocar seu texto para “Pausar” durante a execução.

Tabela 3.4: Elementos de descrição

F4 – Elementos de descrição	
Descrição: O usuário poderá visualizar a descrição de alguns objetos da animação ao passar o mouse sobre os mesmos	
Requisitos Não Funcionais	
Nome	Restrição
NF 4.1 – Posição do Texto	Os textos com descrição dos objetos apareceram ao lado direito da animação.
NF 4.2 – Modo Pausado	As descrições devem ser exibidas mesmo que a animação esteja pausada.

Tabela 3.5: Animação de Apresentação

F5 – Animação de Apresentação	
Descrição: Caso seja a primeira vez do usuário utilizando o framework o software deve recomendar ao usuário assistir uma animação introdutória.	
Requisitos Não Funcionais	
Nome	Restrição
NF 5.1 – Verificar Usuário Inédito	Manter salvo no navegador um cookie informando se o usuário é novo ou não.
NF 5.2 – Pular Animação	Caso deseje o usuário pode pular a animação introdutória.

Para a execução da animação do Core Inc. é importante que o framework possua os dados sobre as variáveis em memória, seus conteúdos e alterações ao longo dos programas submetidos, além de alertar sobre possíveis erros na codificação do programa. Desenvolver tais ferramentas não é do interesse do Core Inc., que tem como objetivo animar os algoritmos e não depurar o código submetido, portanto o auxílio de um depurador externo é necessário para agilizar esta etapa do processo, ao receber o código do usuário e devolver as informações relevantes ao framework de animação. O Diagrama de Contexto da figura 3.11 demonstra visualmente o fluxo destes dados de forma sucinta.

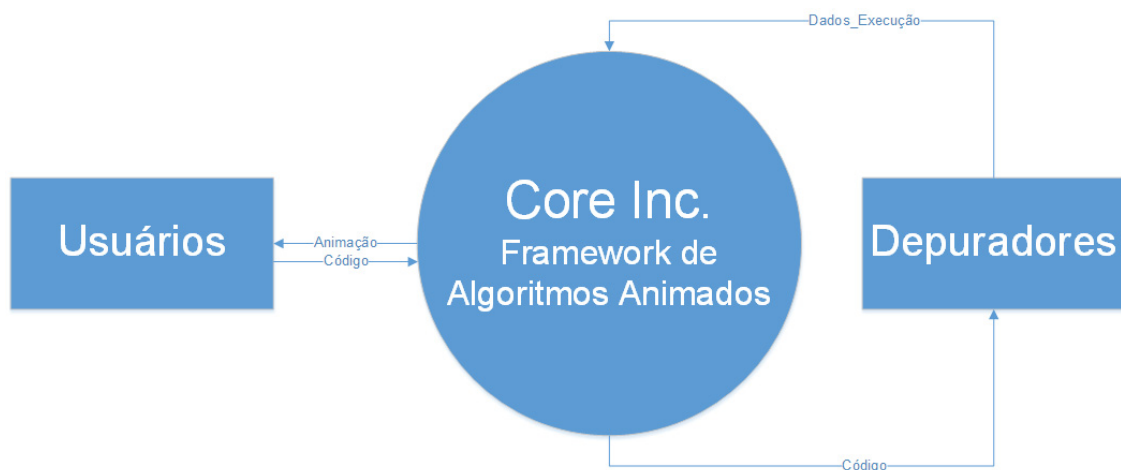


Figura 3.11: Diagrama de Contexto

A passagem do código não ser direta para os depuradores também é proposital, de modo que causaria uma dependência indesejada com um depurador específico caso houver o desejo de troca de linguagem ou depurador utilizado. Portanto o código é recebido e tratado pelo próprio Core Inc., que selecionará seu módulo correto para a linguagem e depurador desejado, de acordo com as configurações indicadas previamente pelo desenvolvedor.

Tal separação também proporciona uma independência do módulo de animação do Core Inc., característica considerada essencial por alguns dos trabalhos relacionados, listados na seção 2.4 O Diagrama de Fluxo de Dados da figura 3.12 ilustra a separação das ações dentro do framework.

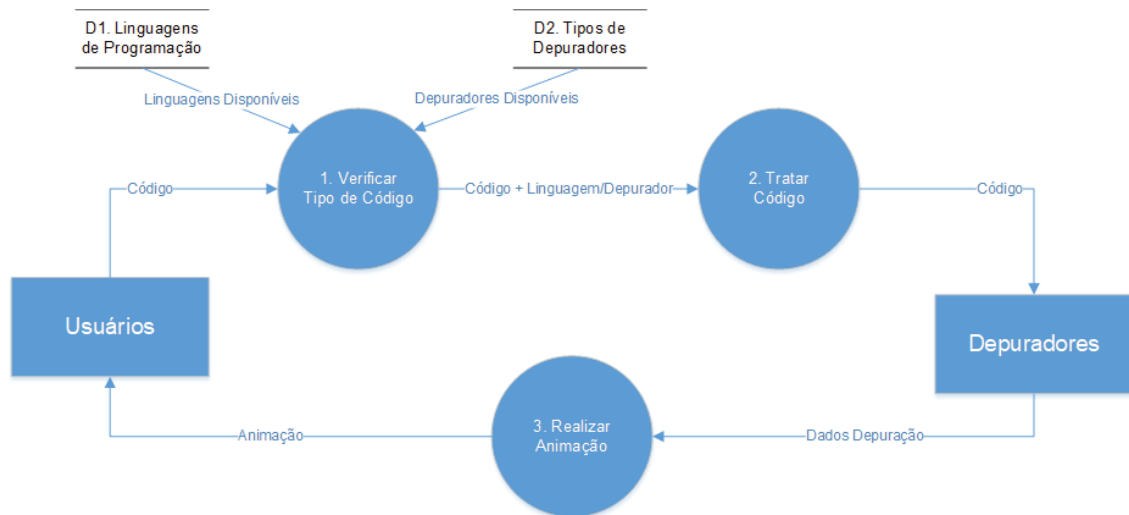


Figura 3.12: Diagrama de Fluxo de Dados

3.4 ESTRUTURA DA ANIMAÇÃO DE ALGORITMOS

Ao animar os algoritmos, não é somente necessário demonstrar a alteração de fluxo e modificação dos dados contidos em cada variável, como considerar o contexto da animação. Esta contextualização, como já explicada anteriormente é uma das bases que fundamentam a proposta e a Teoria da Cognição Situada. Portanto o estilo clássico com formas geométricas apresentado em trabalhos correlatos não é mais o suficiente para atingir os objetivos desejados.

Com isto em mente criou-se um contexto identificável para a animação, com seu mundo e personagens próprios, de forma a dar uma motivação e uma apresentação visual da animação mais próxima à contextualização que os jogos educativos buscam.

Possuindo a contextualização definida e também os requisitos do sistema, um protótipo inicial de baixa fidelidade foi desenvolvido, este que teve como principal objetivo ser o mais claro e direto possível nas funcionalidades e interações do software. O protótipo é demonstrado abaixo na figura 3.13.

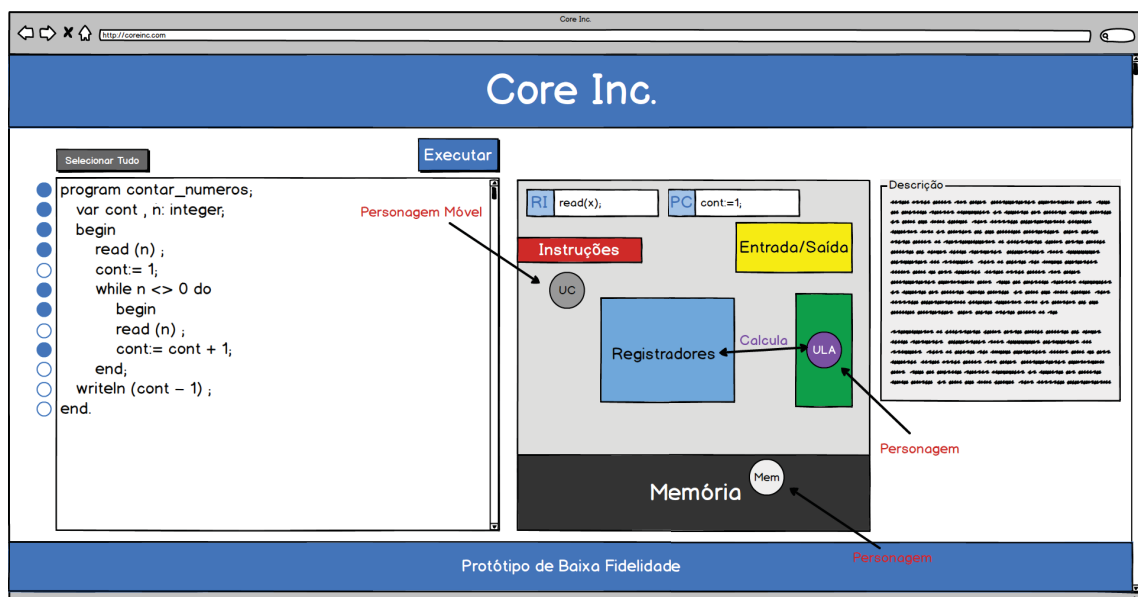


Figura 3.13: Protótipo de Baixa Fidelidade

Enquanto os campos de texto e botões do protótipo são mais claros, possuindo um grau alto de familiaridade com demais software web, a tela responsável pela animação não é autoexplicativa, pelo fato que uma imagem estática não é capaz de demonstrar a interação dos objetos durante a animação.

A ideia por trás dos elementos contidos no retângulo cinza é aplicar um ciclo contínuo de animações baseado na execução de instrução da arquitetura de *von Neumann*, demonstrada na seção 3.1 do texto. Baseado nesta arquitetura, busca-se demonstrar de forma contextualizada, lúdica e simples como o computador executa as instruções essenciais no princípio do aprendizado de algoritmos e programação.

A decisão de seleção das instruções relevantes a serem animadas foi baseada no livro (Castilho et al., 2020) utilizado pelos docentes da disciplina de Algoritmos e Estrutura de Dados I do curso de Ciência da Computação da Universidade Federal do Paraná (UFPR). Levando em conta que a versão inicial do software usa a linguagem Pascal, as instruções selecionadas foram as seguintes:

1. Início

- program "nome do programa"

2. Controle de Fluxo

- if then else

3. Declaração

- CONST
- var

4. Entrada de Dados

- readln;
- read;

5. Saída de Dados

- writeln;
- write;

6. Expressões Aritméticas

- Soma
- Subtração
- Multiplicação
- Divisão
- Raiz quadrada
- Resto

7. Operadores Lógicos

- Igual

- Maior
- Menor
- Diferente
- AND
- OR
- NOT

Com as instruções listadas, abaixo uma série de tabelas detalham como as animações se comportam em execução:

Tabela 3.6: Animação Instrução program

Instrução	program " <i>nome do programa</i> "		
Sequência de Eventos			
N	Personagem	Interage com	Descrição
1	-	-	Tela preta com o sistema inicializando o Core Inc. com o o <i>nome do programa</i> .

Tabela 3.7: Animação Instrução if then else

Instrução	if then else		
Sequência de Eventos			
N	Personagem	Interage com	Descrição
1	Lucy	Painel de Instrução	Caminha até o painel de instrução, coleta a instrução modificando o HUD em PC e IR.
2	Lucy	Painel de Memória	Caminha até o painel de memória e solicita as variáveis necessárias.
3	M3M0	Pacote da Variável	Coleta os pacotes solicitados e coloca na esteira.
4	Lucy	Pacote da Variável	Coleta os pacotes da esteira.
5	Lucy	Registradores	Insere os pacotes nos registradores.
6	Úrsula	Registradores	Pega o conteúdo dos registradores e verifica se é verdadeiro ou falso.
7	Lucy	Painel de Instrução	Caminha até o painel e seleciona verdadeiro ou falso.
8	Lucy	Painel de Instrução	Fim da sequência, inicia-se próxima animação.

Tabela 3.8: Animação Declaração

Instrução	Declaração		
Sequência de Eventos			
N	Personagem	Interage com	Descrição
1	Lucy	Painel de Instrução	Caminha até o painel de instrução, coleta a instrução modificando o HUD em PC e IR.
2	Lucy	Painel de Memória	Caminha até o painel de memória e solicita as variáveis necessárias.
3	Lucy	Painel de Instrução	Fim da Sequência. Inicia-se próxima animação.

Tabela 3.9: Animação Entrada de Dados

Instrução	Entrada de Dados		
Sequência de Eventos			
N	Personagem	Interage com	Descrição
1	Lucy	Painel de Instrução	Caminha até o painel de instrução, coleta a instrução modificando o HUD em PC e IR.
2	Lucy	Painel de Memória	Caminha até o painel de memória e solicita as variáveis necessárias.
3	M3M0	Pacote da Variável	Coleta os pacotes solicitados e coloca na esteira.
4	Lucy	Pacote da Variável	Coleta os pacotes da esteira.
5	Lucy	Bus	Solta pacotes na Entrada/Saída e chama Bus.
6	Bus	Pacote de Variável	Carrega pacotes para saída.
7	Bus	Pacote de Variável	Retorna pacotes após inserção de dados.
8	Lucy	Pacote de Variável	Coleta o pacote da mesa de entrada/saída.
9	Lucy	Painel de Memória	Entrega pacote na esteira da memória..
10	Lucy	Painel de Instrução	Fim da Sequência. Inicia-se próxima animação.

Tabela 3.10: Animação Saída de Dados

Instrução	Saída de Dados		
Sequência de Eventos			
N	Personagem	Interage com	Descrição
1	Lucy	Painel de Instrução	Caminha até o painel de instrução, coleta a instrução modificando o HUD em PC e IR.
2	Lucy	Painel de Memória	Caminha até o painel de memória e solicita as variáveis necessárias.
3	M3M0	Pacote da Variável	Coleta os pacotes solicitados e coloca na esteira.
4	Lucy	Pacote da Variável	Coleta os pacotes da esteira.
5	Lucy	Bus	Solta pacotes na Entrada/Saída e chama Bus.
6	Bus	Pacote de Variável	Carrega pacotes para saída.
7	Bus	Pacote de Variável	Retorna pacotes após exibição dos dados.
8	Lucy	Pacote de Variável	Coleta o pacote da mesa de entrada/saída.
9	Lucy	Painel de Memória	Entrega pacote na esteira da memória..
10	Lucy	Painel de Instrução	Fim da Sequência. Inicia-se próxima animação.

Tabela 3.11: Animação Operadores Lógicos e Aritméticos

Instrução	Operadores Lógicos e Aritméticos		
Sequência de Eventos			
N	Personagem	Interage com	Descrição
1	Lucy	Painel de Instrução	Caminha até o painel de instrução, coleta a instrução modificando o HUD em PC e IR.
2	Lucy	Painel de Memória	Caminha até o painel de memória e solicita as variáveis necessárias.
3	M3M0	Pacote da Variável	Coleta os pacotes solicitados e coloca na esteira.
4	Lucy	Pacote da Variável	Coleta os pacotes da esteira.
5	Lucy	Registradores	Insere os pacotes nos registradores.
6	Úrsula	Registradores	Pega o conteúdo dos registradores e realiza a operação aritmética ou lógica, devolvendo o resultado no registrador.
7	Bus	Pacote de Variável	Retorna pacotes após exibição dos dados.
8	Lucy	Pacote de Variável	Coleta o pacote da mesa de entrada/saída.
9	Lucy	Painel de Memória	Entrega pacote na esteira da memória..
10	Lucy	Painel de Instrução	Fim da Sequência. Inicia-se próxima animação.

A execução de tais instruções pode sofrer variações na animação conforme o resultado de algumas operações, assim como as próprias animações listadas se combinam de diferentes formas na demonstração de operações mais complexas. Contudo, o posicionamento claro de

cada personagem, junto de suas silhuetas e cores características deve ajudar o aluno a identificar a função de cada parte de unidade central de processamento. Não só isto, mas como as ações que a máquina utiliza para entregar o resultado desejado de cada instrução são físicas no contexto do mundo estabelecido, expondo com clareza a execução de cada etapa seja, por exemplo, ao utilizar um computador ou levantar uma caixa.

3.5 DESENVOLVIMENTO DA ANIMAÇÃO

As tecnologias selecionadas para o desenvolvimento das animações são simples quando se olha para o número de ferramentas e linguagens para desenvolvimento web disponíveis no mercado.

Contudo, isto é proposital. Devido a um dos objetivos do software ser buscar a compatibilidade com diversos tipos de software e navegadores, a utilização de ferramentas de terceiros extremamente específicas criaria barreiras para uma possível expansão do projeto por demais desenvolvedores, e também a integração do Core Inc. com ambientes terceiros. Portanto, com pouquíssimas exceções, as tecnologias utilizadas visam a completa compatibilidade do software e sua manutenibilidade. A construção das telas estáticas se dá com puro HTML e CSS3, que ao tratar as imagens feitas em pixel art, atinge o resultado desejado, como já demonstrado na Figura 2.3.

Para fazer com que as animações ocorram, as imagens são sequenciadas lado a lado, como demonstrado na figura 3.14, no que são chamados de *sprite sheet*, ou seja, fotos de momentos diferentes durante a ação do personagem. Enfim, o *sprite sheet* é cortado dinamicamente através do código e animado através da tecnologia de *keyframes* disponibilizada pelo CSS3 e compatível com qualquer sistema ou navegador atual, sem necessidade de nenhuma configuração específica.

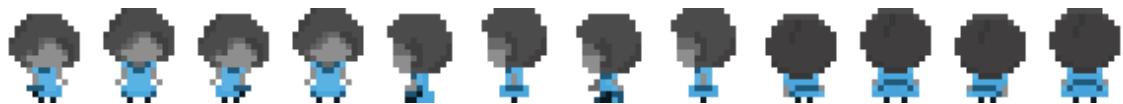


Figura 3.14: Sprite Sheet Lucy

A movimentação dos personagens, essencial para realizar as animações propostas, é compartimentalizada o máximo possível no que foram chamadas de rotas, onde cada personagem possui várias delas. Estas consistem em mover-se do ponto A ao ponto B, de forma a permitir a flexibilização das animações, pois quando separadas podem ser unidas de diversas formas diferentes, permitindo a construção de várias animações mais longas e complexas sem a necessidade de refazer a extensão completa dos movimentos do personagem a cada nova movimentação.

Na figura 3.15 é possível visualizar as rotas na forma que são utilizadas no desenvolvimento, como indicativos visuais para as movimentações possíveis de cada personagem. Neste caso, as linhas azuis correspondem à personagem Lucy, as vermelhas ao M3M0, as rosas a Úrsula, as azuis escuras ao Blue e as amarelas aos pacotes de dados.

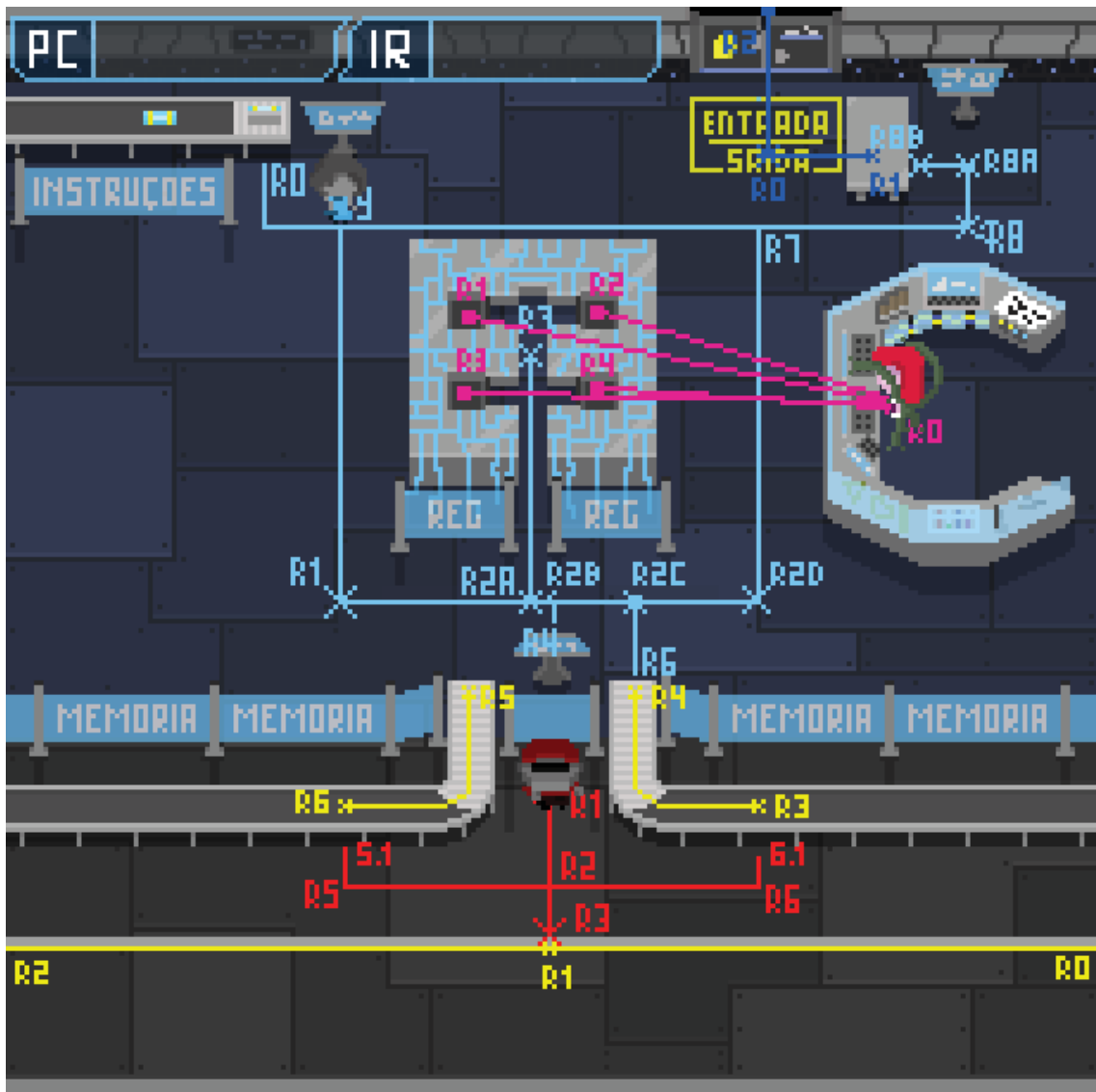


Figura 3.15: Rotas de Animação

Cada um dos cliques de animações, responsáveis por unir várias destas rotas para formar uma movimentação mais longa e complexa, é atrelado a classes CSS que sequenciam e sincronizam os keyframes das rotas, ao mesmo tempo que o nome da classe serve como o ativador da animação por meio de Javascript ou JQuery, como demonstrado nos trechos de código abaixo:

Listing 3.1: Sequenciamento de Cliques em Animação Completa

```

1 #lucy.ask-package {
2   animation-name      : lucy-route1-l, lucy-sprite-front, lucy-route2B-l;
3   animation-duration  : 1s,          0.5s,          1s;
4   animation-timing-function : linear, steps(4),      linear;
5   animation-fill-mode  : forwards,   forwards,      forwards;
6   animation-iteration-count : 1,      2,            1;
7   animation-delay      : 0.5s,       0.5s,          1.5s;
8 }

```

Listing 3.2: Chamada de Animação

```

1 $(".lucy").toggleClass("ask-package");
2 $(".screen").toggleClass("ask-package");

```

Com isto, o ciclo de animação é flexibilizado e ao mesmo tempo facilitado, pois possibilita a criação de diversos cliques de animação diferentes, e as ordens que tais cliques são acionados também pode ser modificada através de controladores em Javascript. Logo abaixo no listing 3.3 é demonstrado parte do código do controlador geral criado para suportar os comandos listados anteriormente. Onde cada função chamada aciona diversos cliques de animação de forma sincronizada.

Listing 3.3: Trecho do Controlador de Animação

```

1  async function Main(instructions) {
2      ResetRegisters();
3      RemoveAnimationClasses();
4      animationStatus = true;
5      for (var index = 0; index < instructions.length; index++) {
6          var string = instructions[index][0];
7          var args = instructions[index][1];
8          var end;
9
10         switch (string.toLowerCase()) {
11             case "program":
12                 end = await Begin(args[0], args[1], args[2]);
13                 console.log("terminou program");
14                 break;
15             case "if":
16                 FadeLoading();
17                 end = await IfThenElse(args[0], args[1], args[2], args[3],
18                     args[4], args[5]);
19                 console.log("terminou if");
20                 break;
21             case "var" || "CONST":
22                 FadeLoading();
23                 end = await Declare(args[0], args[1], args[2]);
24                 console.log("terminou declare");
25                 break; (...)

```

Desta forma, qualquer sistema que deseje absorver o sistema de animações do Core Inc. só necessita realizar chamadas ao controlador geral, desde que tenha como parâmetro os dados corretos para que a animação seja executada. Os parâmetros necessários variam dependendo do comando a ser animado, as operações de soma e atribuição, por exemplo, exigem mais dados do que uma simples leitura de dados. No listing 3.4 é exibido alguns dos parâmetros necessários para animar a leitura de uma variável. Os parâmetros necessários para cada uma das animações disponíveis no Core Inc. estão disponíveis no Apêndice A.

Listing 3.4: Exemplo de Entrada de Dados ao Módulo de Animação

```

1  // Leitura
2  [
3      "read",
4      "linha de instrução",
5      "próxima linha de instrução/comando",
6      "nome da variável",
7      "tipo da variável",
8      "valor da variável"
9  ]

```

Uma descrição detalhada do processo de codificação da ferramenta, com os desafios desde a captura do código do aluno até a execução da animação estão disponíveis no Apêndice G.

3.6 DEMONSTRAÇÃO DE FUNCIONAMENTO

Sabendo os meios com que as animações foram construídas e como são estruturadas, a seguir é demonstrado dado o código abaixo um exemplo de como o Core Inc. funciona ao animar um código em Pascal tipicamente construído por alunos de introdução à algoritmos e programação. O código selecionado foi escolhido dentre os códigos exemplos já inseridos dentro do Core Inc. e utilizados em sala de aula segundo o material utilizado pelo curso de Ciência da Computação (Castilho et al., 2020) da UPFR.

Listing 3.5: Programa para Exemplo

```
1 program imprime_se_positivo;  
2 var a: integer;  
3 begin  
4   read(a);  
5   if a > 0 then  
6     writeln(a);  
7 end.
```

Linha 1: program imprime_se_positivo

De acordo com a tabela 3.6 é exibido o carregamento do programa dentro da animação como demonstrado na figura 3.16 abaixo:



Figura 3.16: Animação Linha 1

Linha 2: var a: integer;

Seguindo a sequência de ações estabelecida na tabela 3.8 a personagem Lucy coleta a nova instrução correspondente a linha em execução e se move pelo cenário solicitando os pacotes necessários para armazenamento na memória e o personagem M3M0 carrega os pacotes até o armazenamento definitivo, como demonstrado na sequência de imagens da figura 3.17.

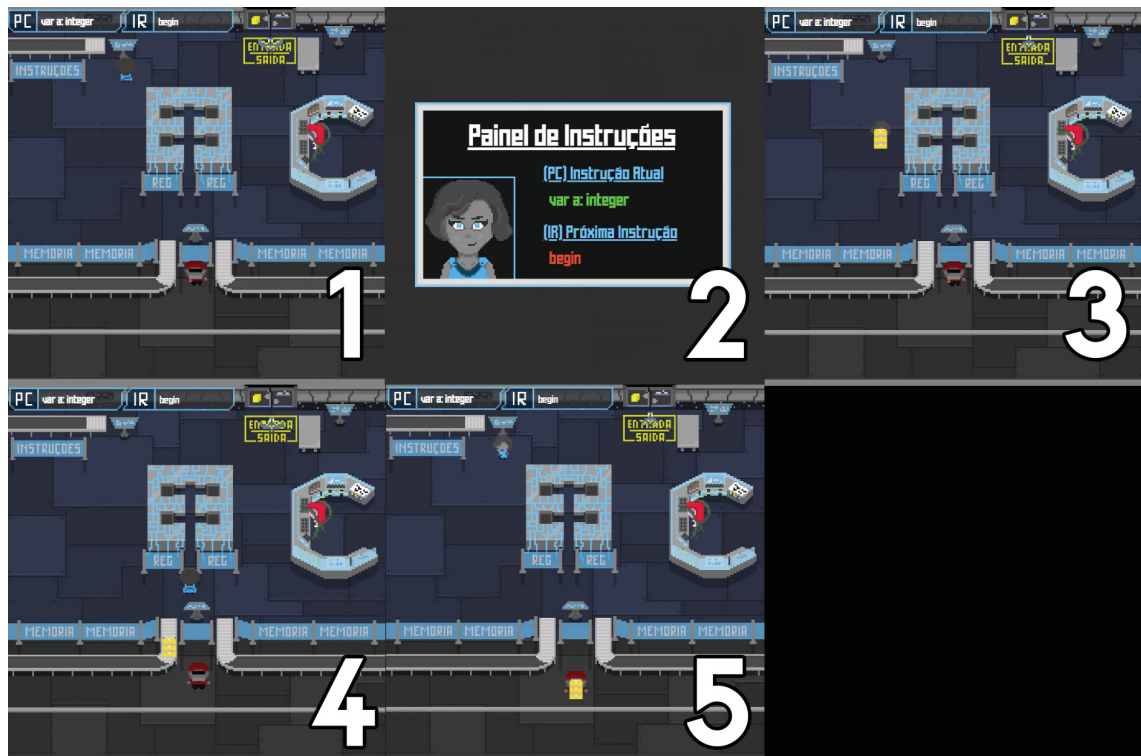


Figura 3.17: Animação Linha 2

Linha 4: read(a);

Para executar a animação de entrada de dados é seguida as instruções da tabela 3.9 em que a personagem Lucy coleta a nova instrução correspondente a linha em execução e se move pelo cenário solicitando o pacote necessário para leitura do dado, ela leva este pacote até o personagem Bus, que sai da fábrica com o pacote vazio e retorna com o pacote preenchido com a entrada de dados fornecida pelo usuário através do campo de Input. Por conseguinte, a personagem Lucy retorna com o pacote para a região da memória e o entrega para o personagem M3M0 armazená-lo. Este processo é demonstrado pela sequência de imagens das figuras 3.18 e 3.19.

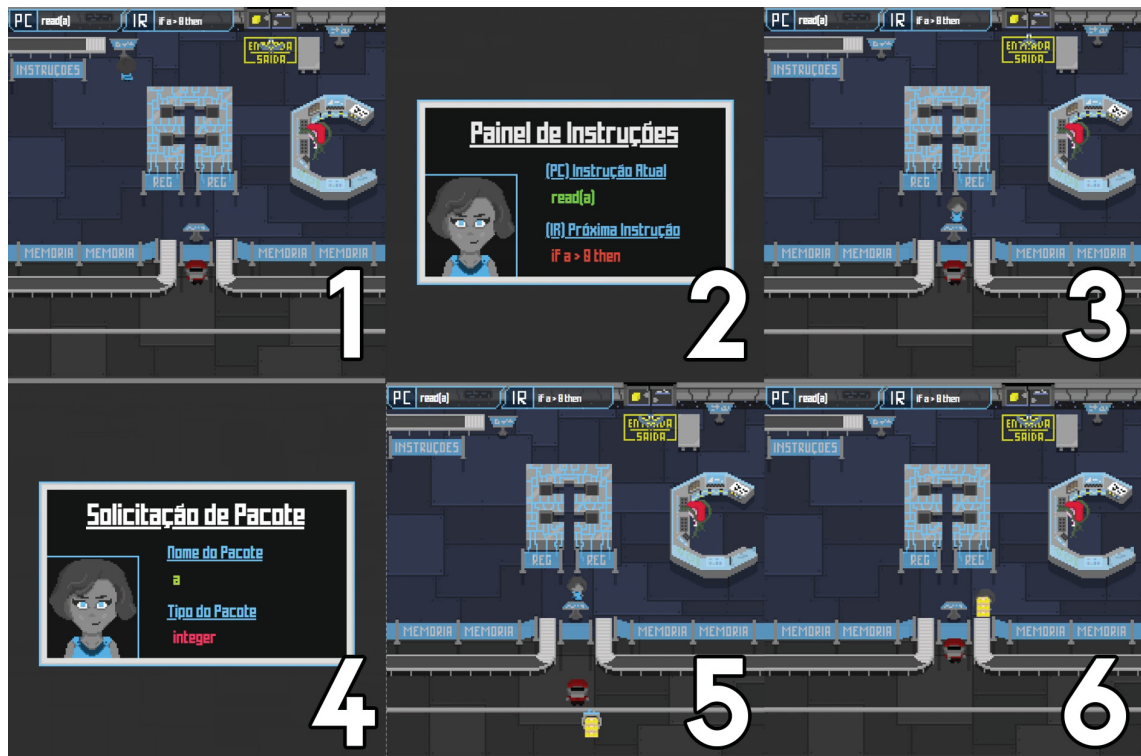


Figura 3.18: Animação Linha 4 - Parte 1

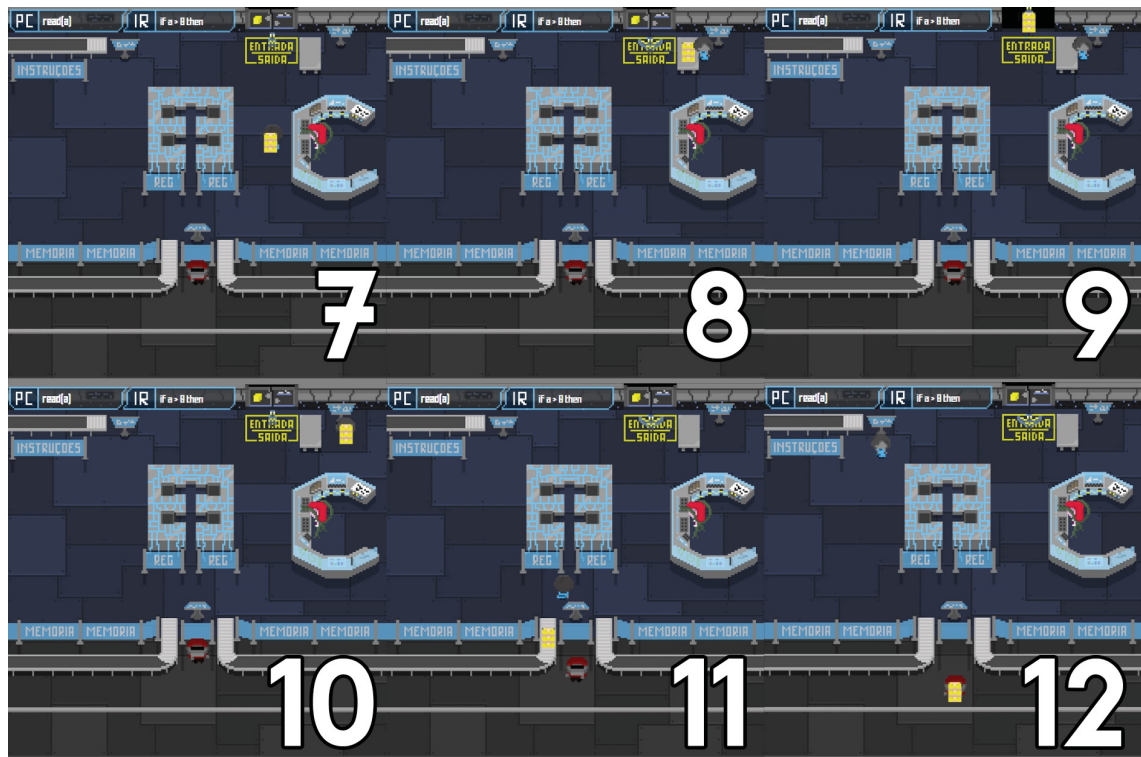


Figura 3.19: Animação Linha 4 - Parte 2

Linha 5: if a > 0 then

Utilizando a ordem de eventos definida na tabela 3.7 a personagem Lucy coleta a nova instrução correspondente a linha em execução e se move pelo cenário solicitando as variáveis

necessárias com o personagem M3M0 para a operação da instrução if e as carrega até os espaços dos registradores como na figura 3.20.

Em um segundo momento a personagem Úrsula, representando a Unidade Lógica e Aritmética estica seus longos braços mecânicos e pega os pacotes para o uso na resolução da expressão, uma tela é exibida com o resultado da expressão e a sequência de animação é finalizada, como demonstrado na figura 3.21.



Figura 3.20: Animação Linha 5 - Parte 1



Figura 3.21: Animação Linha 5 - Parte 2

Linha 6: writeln(a);

A última linha deste código segue a sequência da tabela 3.10 onde a personagem Lucy coleta a nova instrução correspondente a linha em execução e se move pelo cenário solicitando o pacote necessário para escrita do dado, ela leva este pacote até o personagem Bus, que sai da fábrica com o pacote e a animação é encerrada. A sequência de imagens das figuras 3.15 e 3.16 até o quadro 9 representam esta animação.

Fim da Animação

Ao final da animação o Core Inc. retorna para o estado de espera com uma mensagem escolhida aleatoriamente entre as listadas no Apêndice E como é possível visualizar na figura 3.22.



Figura 3.22: Animação Linha 7

3.7 CONSIDERAÇÕES

Esta sessão da dissertação passou por todos os pontos importantes do desenvolvimento da solução em software proposta, demonstrando sua concepção, prototipação e desenvolvimento dos módulos mais relevantes do Core Inc..

Na primeira subseção acompanhou-se passo a passo a formação do software de animação de algoritmos utilizando os conceitos da cognição situada e jogos digitais. Neste processo, nota-se a preocupação com a ambientação das animações, com cenários e personagens que respeitem técnicas utilizadas em jogos digitais enquanto buscar se relacionar com o contexto do cotidiano dos usuários em potencial da solução. A concepção de personagens e cenários levou em conta características de ambientes reais e obras contemporâneas para compor o contexto

familiar necessário segundo a teoria de aprendizagem, enquanto respeitava a técnica do triângulo da esquisitice dos jogos digitais.

Quando definido os conceitos e linhas gerais de como o Core Inc. deveria ser apresentado visualmente e narrativamente, os requisitos do software em si foram estabelecidos, levando em conta as características essenciais de um software de algoritmos animados e as características da cognição situada, conceitos previamente abordados no referencial teórico da dissertação. Atendo-se ao essencial para atingir os objetivos propostos inicialmente, os requisitos foram estabelecidos internamente e todas as funcionalidades foram consideradas essenciais para a criação da solução proposta.

Com o software delineado de forma narrativa e visual, além de suas funcionalidades foram apresentados os diagramas e protótipos criados para o desenvolvimento do software, onde foi demonstrado como foi solucionado desafios técnicos de suporte a várias plataformas e compartimentalização dos módulos de animação para utilização futura. Adicionalmente, o enfoque no módulo de animações, o maior desafio técnico do Core Inc., foi detalhado através da documentação de cada animação e como foi desenvolvido o código fonte deste módulo.

Por fim, esta seção acaba por detalhar os desafios de concepção e desenvolvimento do Core Inc. junto de suas principais características. Contudo, o desenvolvimento por si só da solução não responde aos problemas e nem cumpre todos os objetivos propostos pela dissertação, portanto é necessário avaliar as capacidades do software na visão dos potenciais usuários desta solução e sua viabilidade de aplicação. Somente com estas informações é possível garantir que as decisões tomadas no levantamento de requisitos, definição de telas, narrativa e visuais são capazes de impactar positivamente os usuários. Sendo assim, a seção seguinte é responsável por realizar avaliações qualitativas e quantitativas com a comunidade acadêmica, em busca de esclarecimento sobre as capacidades do Core Inc..

4 EXPERIMENTOS E RESULTADOS

Tendo compreendido quais foram os passos tomados para o desenvolvimento do Core Inc. e como a solução funciona este capítulo avalia se os objetivos pretendidos com o design da solução são atingidos quando colocados a teste. Portanto, aqui são abordados os experimentos realizados como o software e os resultados obtidos. A seção 4.1 inicialmente justifica as motivações que levaram a decisão dos tipos de experimentos realizados. Os experimentos, divididos no experimento principal e complementar nas subseções 4.2 e 4.3 respectivamente. Por fim, os resultados de ambos os experimentos são demonstrados nas seções 4.4 para os dados do experimento principal e 4.5 ao complementar. A seção 4.6 é responsável pelas discussões e considerações finais desenvolvidas após ambos experimentos.

4.1 JUSTIFICATIVA DOS EXPERIMENTOS

Ao longo do desenvolvimento do projeto e da revisão bibliográfica realizada na seção 2.4 notou-se dificuldades em estabelecer relação direta entre o uso de soluções de algoritmos animados com alunos e benefício direto no desempenho dos alunos. Além disto, problemáticas com a percepção dos educadores para com estas soluções também foram identificadas e a ausência de resultados consistentes diminuiu o desejo dos educadores de utilizar tais abordagens. Na última década revisões de literatura demonstraram que a área de visualização e software em geral passou a focar no caráter de engajamento e motivação dos alunos, obtendo resultados mais consistentes com soluções que se afastam mais dos conceitos iniciais de algoritmos animados propostos pelo Core Inc.

Durante o planejamento para realização dos experimentos, o difícil acesso aos alunos em sala de aula devido a pandemia de COVID-19 durante o período de experimentos do projeto nos anos de 2021 e 2022, (momento em que aulas presenciais ainda não haviam retornado normalmente e as turmas existentes no método de ensino a distância) tornaram difícil a realização de experimentos com alunos. Além do que o número de aulas reduzido devido a períodos especiais durante a pandemia retirou do currículo conteúdos que abordavam mais profundamente os temas tratados pelo Core Inc., o que tornou tais temas de caráter opcional, dificultando a introdução do Core Inc. nas aulas. Portanto, apesar do procedimento de aplicação do Core Inc. com alunos passar anteriormente por apreciação do Comitê de Ética do Setor de Ciências Humanas e Sociais para sua liberação, não foi possível realizá-lo.

Por conseguinte, para garantir uma avaliação mais robusta ao Core Inc. decidiu-se abordar os docentes responsáveis por aplicar o software em sala de aula, afinal já foi identificado em anos anteriores que a opinião negativa de professores impacta na decisão de utilização dos software (Hundhausen et al., 2002). As opiniões destes profissionais são tão relevantes quanto o impacto final da solução neste momento da avaliação, pois estes indivíduos podem identificar falhas e potenciais com base em seu conhecimento em sala de aula e ajudar a prever a ação do Core Inc. nos alunos.

Ao realizar avaliações quantitativas através do mesmo questionário utilizado com os alunos e aprovado pelo Comitê de Ética de forma a complementar a avaliação do experimento principal através de entrevistas qualitativas com professores da área de computação é possível identificar falhas na construção do Core Inc. e corrigi-las previamente, permitindo então que a possível aplicação futura da solução em experimentos com uma grande quantidade de alunos seja segura para os mesmos, reduzindo o potencial de impactar negativamente no processo de

ensino-aprendizagem, ao mesmo tempo em que o software estará mais adequado às expectativas e necessidades dos docentes destes alunos.

4.2 DEFINIÇÃO DO EXPERIMENTO PRINCIPAL

O processo iniciou-se com a entrada de um pedido de liberação para a realização do experimento através da Plataforma Brasil, onde o Comitê de Ética do Setor de Ciências Humanas e Sociais aprovou o projeto e liberou a aplicação do Core Inc. junto de um questionário aos alunos e professores no Certificado de Apresentação de Apreciação Ética 55444622.9.0000.0102.

O experimento foi realizado com 4 professores do Departamento de Informática da UFPR, das áreas de sistemas, algoritmos e informática na educação onde eles foram informados sobre o propósito desta dissertação e do software Core Inc. e posteriormente utilizaram o software Core Inc. pelo tempo desejado e testaram suas funcionalidades. Todo este processo foi realizado de maneira digital por videoconferência e o professor compartilhando sua tela com o entrevistador durante o uso do Core Inc.

Um adendo relevante é que o Core Inc. recebeu uma função adicional específica para o processo avaliativo onde o professor poderia utilizar um conjunto de códigos exemplo (Apêndice C) na linguagem Pascal já embutidos no editor de texto através de um botão na interface que trocava entre vários códigos, como demonstrado na figura 4.1. Tal funcionalidade foi adicionada pois nem todos os professores participantes possuíam conhecimento na linguagem Pascal, portanto dificuldades com a plataforma poderiam ser gerados pela dificuldade em codificar um programa na linguagem.

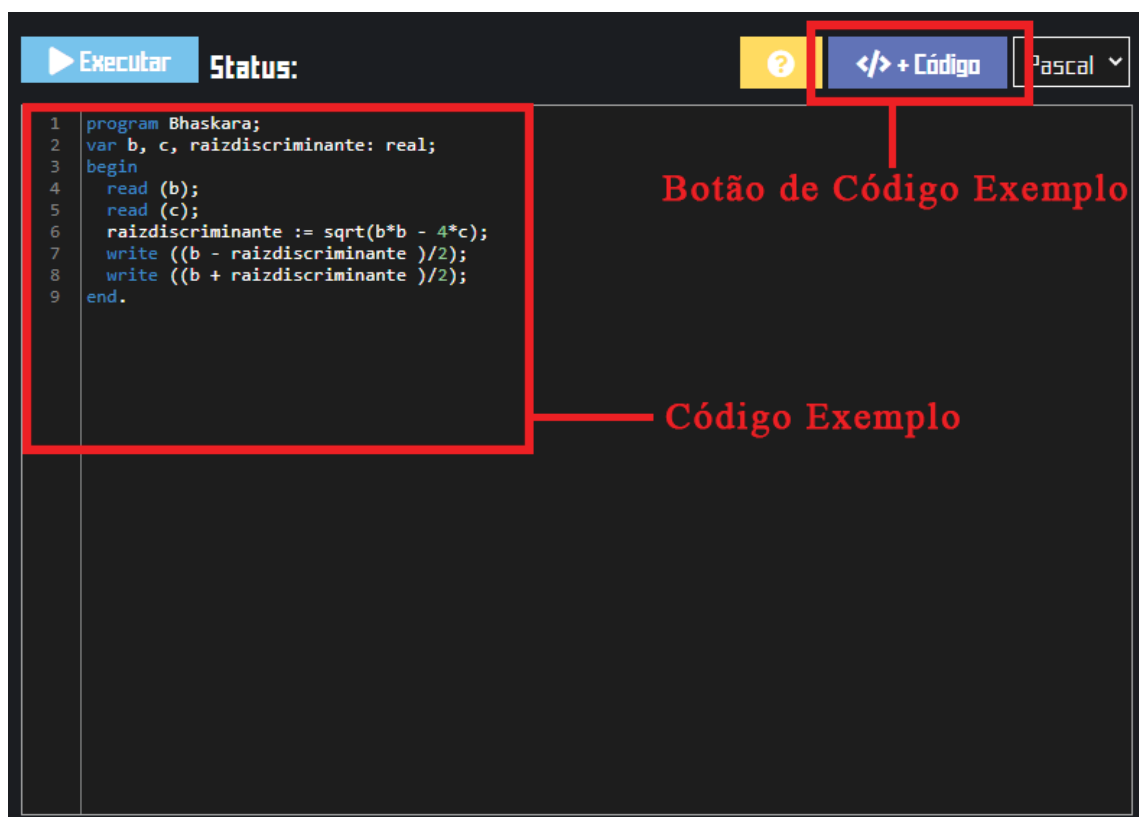


Figura 4.1: Funcionalidade de Códigos Exemplo

Após a utilização do software, foi realizado um experimento qualitativo buscando uma avaliação mais detalhada das impressões dos participantes. Os quatro professores de diferentes

áreas de docência e pesquisa se voluntariaram para participar do experimento. Este consistiu em entrevistas individuais com cada participante.

Esta etapa foi realizada imediatamente após o experimento complementar, que consiste na aplicação de um questionário e é melhor descrito na seção 4.3. A decisão de aplicar o experimento complementar antes do principal foi tomada com o objetivo de permitir aos participantes refletirem sobre o software por alguns minutos antes do experimento principal, com o objetivo de obter opiniões melhor formadas durante a entrevista.

A escolha da metodologia se dá pelo fato das limitações de avaliações quantitativas ao condensar dados e informações em poucas variáveis e categorias, levando a uma generalização dos eventos. Como (Leydens et al., 2004) explicam, dados obtidos através de avaliações qualitativas permitem compreender as motivações a partir de dados coletados através de típicos questionários quantitativos, assim como o método TAM aplicado segundo a seção 4.2, fornecendo uma perspectiva mais humana e detalhada de eventos e fenômenos específicos, neste caso a solução Core Inc.

Outra característica das pesquisas qualitativas é o número reduzido de amostras a serem trabalhadas, focando-se mais nos indivíduos participantes da pesquisa, o contexto e os eventos que ocorreram do que nos números gerados pelo todo (Gerring, 2017), o que se adequa as condições de avaliação do Core Inc com um reduzido número de docentes.

Entretanto, a abordagem qualitativa de pesquisa é uma área ampla com um grande número de técnicas, métodos e perspectivas de trabalho e que a mera definição de utilizar pesquisa qualitativa não seria suficiente para esta avaliação (González, 2020). Sendo assim, dentre os diversos métodos selecionou-se a entrevista, que é um método de coleta de dados amplamente utilizada na área de ciências humanas (Qu e Dumay, 2011) (Manzini, 2012).

A realização de entrevistas porém não pode ser confundida com uma simples conversa entre duas pessoas, o caráter entre entrevistador e entrevistado deve ser levado em conta, assim como o envolvimento do pesquisador com o tema, a base teórica que embasa a entrevista, o modo com que a entrevista será conduzida, a preparação ou não de roteiro por parte do entrevistador. (Qu e Dumay, 2011) Esta preparação é necessária pois entrevistas realizadas sem a preparação adequada podem gerar resultados desapontantes (Hannabuss, 1996).

O primeiro passo ao se preparar uma entrevista de caráter qualitativo é decidir qual é a base teórica utilizada para fundamentar o processo. Dentre as teorias mais importantes segundo Flick e seus colegas (Flick et al., 2004) estão a fenomenologia, etnometodologia, interacionismo simbólico, construtivismo e perspectivas hermenêuticas. A teoria que embasa o experimento realizado nesta dissertação é a fenomenologia.

A fenomenologia, inicialmente um movimento filosófico (Andrade e Holanda, 2010), é considerada uma ciência que se aplica ao estudo de fenômenos de eventos, fatos e objetos da realidade (Petrelli, 2008). Um dos fundamentos da pesquisa de caráter fenomenológico é o retorno às origens, analisando o evento dado como é a própria realidade, abrindo mão de pressupostos propostos pela própria consciência. A consciência em si possui intencionalidade, que é indesejada quando se trata de legitimar cientificamente uma pesquisa (Andrade e Holanda, 2010). Para isto, a redução fenomenológica é o recurso utilizado para chegar à essência deste fenômeno, dando suporte científico a entrevista. Para que isto aconteça o pesquisador/entrevistador precisa se colocar aberto à descoberta, abrindo mão de hipóteses e afirmações prévias sobre o evento. Este processo é que se chama de adotar uma “postura fenomenológica”. Porém, é importante notar que é impossível reduzir o conhecimento prévio do pesquisador em sua totalidade, como (Merleau-Ponty, 1999) explica.

Portanto ao buscar esta redução o roteiro das entrevistas, disponível no Apêndice E, assim como o entrevistador buscaram se distanciar o máximo possível de afirmações prévias que pudessem ser passadas aos entrevistados, modificando a percepção dos mesmos sob o Core Inc.

O roteiro das entrevistas elaborado seguiu um paradigma semiestruturado, onde são elaborados tópicos de discussão e perguntas normalmente abertas para várias respostas, permitindo um diálogo mais natural entre entrevistador e entrevistado (DiCicco-Bloom e Crabtree, 2006).

Dado isto, a relação entre entrevistador e entrevistado gera uma história capaz de compreender um fenômeno (Andrade e Holanda, 2010), neste caso as percepções sobre o Core Inc. com cada professor participante. Após isto, o retorno ao que foi dito e vivido é necessário para avaliar e questionar quais foram os pontos essenciais de cada entrevista. Esta é demonstrada na seção 4.5. Por fim, para garantir que as avaliações e questionamentos gerados em cada entrevista estão de acordo com as ideias originais do entrevistado também é necessária a avaliação do próprio entrevistado sobre o material de análise gerado.

Um adendo importante sobre a condução das entrevistas foi a gravação e transcrição do procedimento com a liberação dos participantes, para uso posterior na análise das entrevistas. Os nomes dos participantes e menções a pessoas e instituições foram omitidos o máximo possível sem afetar o entendimento da entrevista nas transcrições e análises com o objetivo de manter o anonimato dos participantes.

Para um entendimento mais claro do processo de preparação das entrevistas, os seguintes passos foram seguidos:

1. Adoção do embasamento teórico fenomenológico.
2. Desenvolvimento de um roteiro semiestruturado de acordo com a postura fenomenológica.
3. Condução da entrevista com os participantes.
4. Retorno aos dados gerados pela entrevista para análise, disponível na seção 4.5.

4.3 DEFINIÇÃO DO EXPERIMENTO COMPLEMENTAR

O experimento complementar foi realizado com os quatro professores voluntários imediatamente após o uso do software Core Inc. e antes do experimento principal descrito na seção 4.2. Este consistiu a responder um questionário na plataforma Google Formulários, disponibilizado para os professores pelo entrevistador logo após o professor se considerar satisfeito com seu tempo de teste com o Core Inc.

O questionário aplicado consistiu em duas etapas. A primeira delas tinha como objetivo coletar informações sobre a experiência prévia do participante com soluções de algoritmos animados ou visualização de software. Tais dados foram considerados relevantes para análise pois as impressões do Core Inc. poderiam ser afetadas por comparações com experiências prévias dos participantes com este tipo de tecnologia.

A segunda etapa do questionário consistiu no método de avaliação quantitativo selecionado para este experimento. As perguntas foram formuladas de acordo com o Modelo de Aceitação de Tecnologia (TAM), um questionário utilizado na literatura para investigar o grau de aceitação e potencial de uso de um software diante de seu público alvo. Este método também foi demonstrado anteriormente na seção 2.7 desta dissertação. As perguntas de ambas as etapas estão disponíveis no Apêndice D.

4.4 RESULTADOS EXPERIMENTO PRINCIPAL

A entrevistas foram realizadas primariamente de acordo com o roteiro desenvolvido, porém a ordem dos temas abordados não foi a mesma do roteiro, assim como assuntos além deste roteiro surgiram em momentos diversos ao longo das entrevistas, conforme os professores desenvolviam suas linhas de raciocínio. Esta entrevista ocorreu posteriormente ao uso do software Core Inc. e a resposta de um questionário. Todo este processo foi realizado de forma digital e teve uma duração média de uma hora e trinta minutos.

Todos os trechos capturados da transcrição das entrevistas e citados durante a análise da mesma junto da própria análise são separados tematicamente de acordo com os tópicos de relevância abordados pelo roteiro da entrevista. É importante atentar-se que como estas são entrevistas semiestruturadas a ordem dos trechos dispostas e citadas a seguir não correspondem a ordem cronológica destes mesmos trechos durante a realização das entrevistas. Além disto, dado o caráter ambíguo que uma entrevista oral pode possuir, a análise de todas as entrevistas foi revisada pelos respectivos professores participantes de forma a garantir que as opiniões fornecidas e análises realizadas estão de acordo com as opiniões e intenções originais de cada professor.

Abaixo são apresentadas as análises realizadas divididas por professor entrevistado, onde inicialmente é disposto em uma tabela os trechos de cada respectiva entrevista segundo seus temas e logo após a análise realizada de forma textual.

4.4.1 Análise Temática Entrevista Professor A

Tabela 4.1: Frases Entrevista Professor A

Modelo von Neumann	Bom, eu considero muito importante. Vamos dizer assim, a ideia de utilizar isso em cursos introdutórios sempre foi defendida por mim aqui.
Modelo von Neumann	Bom, eu considero muito importante. Vamos dizer assim, a ideia de utilizar isso em cursos introdutórios sempre foi defendida por mim aqui.
Modelo von Neumann	Eu que inicialmente incentivei isso e quando eu dou disciplinas introdutórias, eu sempre uso essa ideia. Então considero ela muito importante.
Modelo von Neumann	Eu acho que a correta compreensão aí desse mecanismo, né? De como é um mecanismo básico. Do ciclo de instrução de como é que as coisas acontecem, por que elas acontecem. Eu acho que ela ajuda bastante o aluno.
Modelo von Neumann	Inclusive pela minha experiência positiva como aluno.
Modelo von Neumann	Uma ideia esquemática, eu acho que sim, faz falta, faz diferença, deixa um buraco aí na formação, se não for corretamente apresentado.
Algoritmos Animados e Visualização de software	Eu não estava pensando tanto em visualização de softwares, mas eu estava pensando em visualização de algoritmos, certo, né?
Algoritmos Animados e Visualização de software	Eu acho que o apoio visual por diagramas, por desenhos, por é, enfim, que seja lá o tipo de instrumento que se use, eu acho que ele faz muita diferença. Essa também é minha experiência, tanto como aluno como professor.

Algoritmos Animados e Visualização de software	Mesmo em nível mais elementar, aí em disciplinas introdutórias e tudo (...) então, acho que mesmo nesse tipo de situação eu acho que também faz diferença, eu acho que o apoio visual faz diferença.
Conteúdos adaptados à cultura do aluno	Eu acho que faz e eu acho que em se tratando de ensino superior é preferível a representação mais abstrata.
Conteúdos adaptados à cultura do aluno	Eu acho que assim representações antropomórficas, representações lúdicas, etc. eu acho que atrapalham em vez de ajudar no processo de aprendizado quando a gente tá falando de ensino superior.
Conteúdos adaptados à cultura do aluno	Eu não tenho experiência, não tenho nenhum conhecimento assim, não me atrevo a falar no de ensino médio de ensino fundamental, porque assim, escapa completamente da minha competência.
Conteúdos adaptados à cultura do aluno	Eu acho que o objetivo deveria ser desses apoios visuais, desse tipo de ferramenta o objetivo deveria ser deixar o mais limpo possível o conceito que você está querendo apresentar. O mais limpo de qualquer distração.
Conteúdos adaptados à cultura do aluno	O objetivo do ensino superior é fazer o aluno se acostumar a pensar de maneira abstrata. Você puxar isso pro outro lado, na minha opinião, é trabalhar contra o objetivo, o que deveria ser o objetivo do ensino.
Conteúdos adaptados à cultura do aluno	A gente tem que facilitar o raciocínio abstrato do aluno, não é? Não tentar substituir ele por outra coisa, como eu vou chamar, mais palatável ou mais próxima do aluno.
Impressões gerais Core Inc.	Então eu fiquei com a minha primeira impressão, foi que isso seria dirigido a uma faixa etária menor. (...) Acho que mais assim de ensino fundamental.
Impressões gerais Core Inc.	A experiência ser muito passiva da parte de quem está usando.
Tutorial e Usabilidade Core Inc.	Sim, eu acho que sim. Eu acho que sim.
Tutorial e Usabilidade Core Inc.	Para mim é imediata associação entre uma coisa e outra, não é? A tal ponto que eu não tenho sensibilidade para saber se para o aluno aquilo está tão claro quanto está para mim.
Tutorial e Usabilidade Core Inc.	É preciso dizer que faz tempo que eu não dou aula introdutória de computação. É a última turma que eu peguei assim de uma disciplina dessas pode fazer mais de 12 anos.
Tutorial e Usabilidade Core Inc.	Pelo que eu entendi não haveria a possibilidade de eu experimentar com valores diferentes lá na entrada, na entrada do algoritmo que está executando.
Temática e representação do Core Inc.	Eu realmente acho que representar as coisas com personagens, com bonequinhos, com animais, ou seja lá o que for, não faz muita diferença. Quer dizer não faz muita diferença no seguinte sentido, você não vai capturar a atenção dele, o interesse dele mais ou menos fortemente.

Temática e representação do Core Inc.	Essa outra representação simbólica na minha apreciação dessa pequena experiência que eu fiz ela talvez esteja um pouquinho mais perto da realidade concreta de como é o funcionamento do computador, mas ainda está longe. Ela ainda é mais uma alegoria do que uma explicação ou do que uma ilustração.
Temática e representação do Core Inc.	Ela está mais para uma alegoria da arquitetura interna do computador do que para uma ilustração, para uma representação. E eu acho que para o aluno do ensino superior, especialmente quem está tendo o seu primeiro contato aí com o assunto e tudo mais eu acho que é muito mais útil, mais produtivo ele ter uma ilustração do que simplesmente uma outra representação simbólica.
Temática e representação do Core Inc.	É, eu, é como eu disse para você, pensando em alunos de 17, 18, 19 anos não me parece que estava particularmente próxima, nem particularmente longe.
Temática e representação do Core Inc.	A maneira como que as coisas estão representadas ali, o uso dos personagens e tal, para mim fazem parte do daquilo que hoje em dia é a realidade de todo o mundo que chega ao ensino superior. Então está próximo no sentido que o alfabeto está próximo, a eletricidade está próxima.
Meio de abstração do Core Inc.	Eu tenho a impressão que, do ponto de vista de um aluno a coisa fica desinteressante assim, não tem um desafio, não tem um engajamento ali.
Meio de abstração do Core Inc.	É enfim, a coisa rapidamente se torna previsível demais. Que eu temo, não é que no uso ela se torne desinteressante, quer dizer, e quando eu falo que ela se torna desinteressante assim ela vai fazer pouca diferença naquilo que é o objetivo que é a compreensão mais fácil, mais rápida, mais eficiente aí dos conceitos fundamentais.
Meio de abstração do Core Inc.	É que a animação para mim trabalha contra o objetivo de uma coisa dessas. A animação em vez de expor, ela esconde conceitos na minha opinião.
Meio de abstração do Core Inc.	É, e esses conceitos que no meu entender são fundamentais eles não são apresentados com clareza, com fidelidade, não é? Então, isso pra mim mata o propósito, o objetivo que você tem quando você quer falar dessas coisas, não é?
Meio de abstração do Core Inc.	Eu acho que ter opções é bom, nem todo mundo aprende do mesmo jeito. (...) eu não acho que do jeito que está, do jeito que eu experimentei aí o Core Inc., eu não acho que ele faça diferença significativa no processo de aprendizado.
Aplicação do Core Inc. em sala de aula.	É o momento em que o aluno está sendo apresentado esses conceitos, não é? E isso tem um momento determinado ali no andamento da disciplina, não é? Então acho que o momento está determinado aí, pela maneira como a disciplina está montada, construída.

Aplicação do Core Inc. em sala de aula.	Eu não sei, é aí, eu não sei, não sei porque não pensei no assunto.
Aplicação do Core Inc. em sala de aula.	Cenários de uso que eu imagino, não é? É um cenário completamente offline, no sentido de você dar o link lá para o aluno e fala, faça os exercícios lá, faça o tutorial lá, enfim dar uma lição de casa para ele, para usar. Ou uma versão mais online onde o professor está presente quando eles usam.
Aplicação do Core Inc. em sala de aula.	É por conta desse elemento que falei de que me pareceu passivo demais a interação, talvez eu me atrevesse a dizer que se você der a coisa offline, como uma lição de casa, ele vai ser menos efetivo porque o aluno vai se desinteressar mais rapidamente.
Modificações e correções Core Inc.	Quer dizer, a variável é uma caixinha onde eu não vejo o que está escrito nela? E só se eu parar. A animação para ver, não é? A variável ela não tem valor num determinado momento não é assim.
Modificações e correções Core Inc.	A coisa da memória apresentada de um jeito que a memória está fora do cenário. Não é assim. Eu interpreto assim porque aquele bonequinho vermelho lá que recebe as memórias, eu interpretei aquilo como sendo uma espécie de caixa, uma espécie de barramento, enfim, ele joga o valor lá num negócio e aquele negócio vai para a memória.
Modificações e correções Core Inc.	E a entrada e a saída estão no mesmo lugar na animação, não é? Isso não corresponde ao mecanismo elementar, não é? A entrada e saída são 2 buraquinhos separados, não é? São independentes.
Modificações e correções Core Inc.	A memória, por sua vez, é um barramento só, mas na animação elas são 2 coisas separadas, tem um por onde chega e um outro por onde a memória sai.
Modificações e correções Core Inc.	É para mim, por exemplo seria menos passivo se o aluno precisasse, sei lá, apertar um botão para que a próxima coisa acontecesse, para que a próxima linha seja executada ou para que o próximo pedaço de ação acontecesse.
Modificações e correções Core Inc.	Por exemplo, se você tivesse, botões diferentes para ações diferentes e o aluno tivesse que acertar a próxima ação. Quer dizer o que vai acontecer agora, ela vai levar o bonequinho, vai levar a memória para a mesa da entrada e saída, então ele tem que apertar o X. E se ele apertar uma outra tecla é um erro.
Modificações e correções Core Inc.	Seriam coisas desse por essa linha, vamos dizer assim, que não tivesse uma resposta única, não é? E o aluno, de alguma forma é, está, vamos dizer assim, interagindo com a animação.

Avaliação do processo de entrevista	Não, não de forma nenhuma, Tudo ok. A conversa foi produtiva, foi bastante objetivo, não é? Quer dizer, você tinha um roteiro aí, um objetivo a desempenhar. Fazer eu ter contato com o software etc., responder ao questionário, depois a entrevista, não é? Tanto o questionário quanto a entrevista as perguntas eram bem objetivas e bem pertinentes eu acho.
Avaliação do processo de entrevista	Tudo ok, não tem absolutamente assim nenhum incômodo não, é pelo contrário.

Como o professor A foi um dos professores selecionados da área de teoria da computação, suas afirmações sobre a importância do modelo de *von Neumann* em seu dia a dia contiveram afirmações como “eu considero muito importante” e “Eu que inicialmente incentivei isso e quando eu dou disciplinas introdutórias, eu sempre uso essa ideia” ao se referir ao modelo e seu ciclo de instrução.”. O professor ainda foi além ao comunicar de maneira clara a importância deste ciclo de instrução para o processo de formação de um aluno da área de computação, ao dizer que “Uma ideia esquemática, eu acho que sim, faz falta, faz diferença, deixa um buraco aí na formação, se não for corretamente apresentado”, corroborando as ideias defendidas nesta dissertação, da importância do ciclo de *von Neumann* na educação dos alunos de ensino superior dos cursos de computação e como a ausência deste tipo de conhecimento pode causar como o professor A mesmo diz um “buraco” na formação do aluno, potencialmente afetando o resto de sua formação.

O professor A também comentou suas experiências com softwares de algoritmos animados e visualização de software. Ele comenta que viu muito o resultado de vários softwares que desenvolviam animações, mas nunca utilizou um software que desenvolve a animação em si. Ao corrigir o entrevistador ao dizer “Eu não estava pensando tanto em visualização de softwares, mas eu estava pensando em visualização de algoritmos” ele explica que está mais acostumados com soluções de animações que trabalham com diagramas e representações mais tradicionais de algoritmos, como alguns dos trabalhos relacionados demonstrados no capítulo 2.

Dado sua experiência ele explica que tanto como aluno quanto como professor é positiva com esse tipo de software ao afirmar que “(...) apoio visual por diagramas, por desenhos, por seja lá o tipo de instrumento que se use, eu acho que ele faz muita diferença.”, ele ainda adiciona a este comentário sobre o tema ao explicar que “Mesmo em nível mais elementar, aí em disciplinas introdutórias e tudo (...) então, acho que mesmo nesse tipo de situação eu acho que também faz diferença, eu acho que o apoio visual faz diferença.” apesar de sua experiência atual ser de algoritmos mais complexos.

Portanto, o professor A não demonstra somente uma positividade em relação ao ensino do modelo de *von Neumann* como também ao uso de softwares de algoritmos animados durante esse período inicial do ensino, onde estas disciplinas se encontram e onde o Core Inc. procura agir.

Apesar de considerar os temas abordados pelo Core Inc. e o método escolhido pela solução de animação de algoritmos adequados e promissores o professor A discorda da abordagem cognitiva que embasa a solução Core Inc. Ao se referir as soluções de algoritmos animados ele afirma que “se tratando de ensino superior é preferível a representação mais abstrata.” Ao contrário da proposta desta dissertação, onde o Core Inc. é completamente fundamentado em ideias que buscam trazer representações lúdicas da cultura e realidade dos alunos para as animações. Apesar de discordar da abordagem do Core Inc o professor A deixa muito claro que sua perspectiva vem somente do ensino superior ao dizer que “Eu não tenho experiência,

não tenho nenhum conhecimento assim, não me atrevo a falar no de ensino médio de ensino fundamental, porque assim, escapa completamente da minha competência.”.

A literatura até aqui abordada sobre os algoritmos animados demonstrou incerteza na efetividade destas soluções com visualizações mais abstratas de algoritmos, estas que já foram desenvolvidas das mais diversas formas ao longo dos anos e várias demonstradas no capítulo 2. Justamente uma das principais justificativas para o desenvolvimento da solução do Core Inc. é sua diferenciação através de seu embasamento na teoria da cognição situada. Mesmo sabendo disto o entrevistador não interferiu nas opiniões do professor A que trouxe mais apontamentos importantes sobre o porquê ele acredita que soluções abstratas são a abordagem correta.

Ele acredita que os conceitos apresentados devem ser limpos, como ele diz “O mais limpo de qualquer distração.”, justificando que soluções que modificam e adicionam elementos na tentativa de facilitar o que o aluno precisa abstrair e a própria complexidade desta abstração do conteúdo vão contra o propósito do ensino superior, como ele diz “O objetivo do ensino superior é fazer o aluno se acostumar a pensar de maneira abstrata. Você puxar isso pro outro lado, na minha opinião, é trabalhar contra o objetivo, o que deveria ser o objetivo do ensino.”. Ao dizer isto o professor A acredita que o Core Inc. não está facilitando a abstração do aluno e sim substituindo: “Não tentar substituir ele por outra coisa, como eu vou chamar, mais palatável ou mais próxima do aluno.”. Com isto se nota diferenças conceituais fundamentais entre a abordagem do Core Inc. e sua metodologia e o as opiniões do professor A sobre este tipo de software, o que aliado com outros problemas que ele aponta mais à frente faz com que sua opinião geral sobre o Core Inc. não seja positiva.

Quando indagado das suas impressões iniciais sobre o Core Inc. o professor A foi extremamente detalhista em cada um de seus pontos. Ele trouxe à tona várias perguntas que seriam feitas posteriormente, contudo apontamentos válidos de nota neste momento inicial foram sua confusão sobre o propósito do software ao dizer que “Então eu fiquei com a minha primeira impressão, foi que isso seria dirigido a uma faixa etária menor. (...) Acho que mais assim de ensino fundamental.”. Esta dúvida mesmo foi sanada durante o uso do software antes da entrevista, em que o professor ficou confuso sobre o propósito do mesmo. Isto foi um ponto importante para que essa dúvida não ocorresse em demais entrevistas.

Além disto, o professor A considerou a experiência muito passiva, ao explicar que mesmo que o aluno possa mexer com a execução da animação ela corre por si só sem a necessidade de input do aluno. Esta percepção é um fator interessante, pois como visto nesta dissertação o Core Inc. busca se enquadrar em características de softwares de algoritmos animados ativos. Contudo o problema aqui para o professor A parece ter mais relação com a passividade da própria área de algoritmos animados do que com o Core Inc.. Ele faz apontamentos sobre a falta de interação durante a execução da animação além dos controles sob sua execução, o que é uma característica desse tipo de solução mesmo em abordagens ativas no tema. Logo adiante ele mesmo irá propor interações diferentes que vão além da área de solução de algoritmos animados.

Sobre a usabilidade e tutoriais do Core Inc. o professor A deu respostas sucintas afirmando que não achou problemas no uso do software. Quando o entrevistador explicou que o próprio conhecimento do mesmo dando aula para os alunos é valioso ao avaliar a tela o professor A adicionou que “É preciso dizer que faz tempo que eu não dou aula introdutória de computação. É a última turma que eu peguei assim de uma disciplina dessas pode fazer mais de 12 anos.”, o que, segundo ele, gera dúvidas consigo mesmo se suas opiniões sobre este tema podem não ser acuradas com a realidade dos alunos que utilizariam o Core Inc. no futuro.

Apesar de sua análise positiva da usabilidade do software o professor se confundiu um pouco com o funcionamento do campo “Input” da tela do Core Inc. durante o uso do software, este campo permite definir quais as entradas de dados do código, caso elas existam.

Esta confusão o levou a fazer afirmações confusas durante a entrevista, que foram rapidamente corrigidas pelo entrevistador, contudo isto orienta trabalhos futuros na melhoria da comunicação do funcionamento desta funcionalidade do Core Inc. e também permitiu uma correção na comunicação desta funcionalidade nas entrevistas que são apresentadas em seguida. Especula-se que a funcionalidade de códigos exemplos disponibilizada possibilitou essa confusão que as entradas de dados são fixas já que o professor não necessita interagir diretamente com o campo de “Input”.

Quanto a temática e representação visual e narrativa do Core Inc. as respostas do professor A foram similares a sua opinião geral sobre o tipo de representação contextualizada que o Core Inc. já busca fazer. Ele deixa claro que não acha que faz diferença esse tipo de abordagem ao dizer “Eu realmente acho que representar as coisas com personagens, com bonequinhos, com animais, ou seja lá o que for, não faz muita diferença.” e “E eu acho que para o aluno do ensino superior, especialmente quem está tendo o seu primeiro contato aí com o assunto e tudo mais eu acho que é muito mais útil, mais produtivo ele ter uma ilustração do que simplesmente uma outra representação simbólica”. Portanto, ele reforça que o tipo de abordagem que o Core Inc. propõe com seus personagens é menos eficaz que abordagens mais abstratas e diretas ao ponto.

Quanto a própria representação, o professor A disse que os personagens e história estava relacionado ao contexto dos alunos da mesma forma que qualquer coisa no mundo real estaria relacionada ao dizer que “Então está próximo no sentido que o alfabeto está próximo, a eletricidade está próxima.”, apesar de ser vago em suas palavras analisa-se que parte do objetivo parece ser cumprido quando o professor consegue visualizar que a representação contida na animação se relaciona ao mundo real dos alunos, ao mesmo tempo ele não enxerga um diferencial que é o buscado através das inspirações que cada personagem possui. É difícil dizer se o professor A não identifica as contextualizações dos tipos de desenhos utilizados e as características que cada personagem tem de elementos da cultura popular, estes listados na ficha de cada personagem na seção 3.1 por uma falha na representação dos personagens ou pelo próprio fator que ele disse do afastamento de sua visão da visão dos alunos mais novos, portanto somente uma avaliação diretamente com os alunos poderia sanar essa dúvida.

O meio de abstração utilizado através das animações para representar o funcionamento do computador com o ciclo de instrução de *von Neumann* também foi abordado durante a entrevista e o professor A voltou a demonstrar preocupação quanto ao engajamento do aluno com a atividade e a passividade do Core Inc. ao dizer que “do ponto de vista de um aluno a coisa fica desinteressante assim, não tem um desafio, não tem um engajamento ali.” Além de acrescentar que isto pode afetar a efetividade do software já que “quando eu falo que ela se torna desinteressante assim ela vai fazer pouca diferença naquilo que é o objetivo que é a compreensão mais fácil, mais rápida, mais eficiente aí dos conceitos fundamentais.”.

Não só isto, mas o professor A levanta problemas na própria representação proposta pelo Core Inc., indo além do fator da passividade das animações ou da própria abordagem contextualizada que ele propõe, já que segundo ele vários conceitos ali não são bem apresentados ao dizer que “esses conceitos que no meu entender são fundamentais eles não são apresentados com clareza, com fidelidade”. Quando indagado ainda sobre se mesmo com os problemas que ele possui com essa abordagem contextualizada, fugindo de opções abstratas que são preferíveis a ele o Core Inc. poderia ser uma boa opção adicional aos alunos o professor A também foi negativo em sua resposta ao dizer que sim, ter opções de métodos diferentes de aprendizado é bom para os diferentes tipos de alunos, contudo ele acrescenta que “eu não acho que do jeito que está, do jeito que eu experimentei aí o Core Inc., eu não acho que ele faça diferença significativa no processo de aprendizado.”.

Parte da avaliação também conteve perguntas sobre a opinião do professor A sobre a aplicação do Core Inc. em sala de aula, independentemente dos problemas até então encontrados por ele. Sobre quais seriam as abordagens que ele utilizaria e os melhores métodos. Ele compreendeu que o momento que o software será aplicado não possui muito espaço para modificações, pois precisa ser quando o modelo de *von Neumann* é apresentado, como ele diz “É o momento em que o aluno está sendo apresentado esses conceitos”.

Quanto ao uso do Core Inc. em sala de aula ele conseguiu comentar sobre o uso como tarefa de casa ou o uso em sala de aula com o acompanhamento por um professor e nisto ele fala sua preferência neste caso por um método em sala de aula, ele comenta que “É por conta desse elemento que falei de que me pareceu passivo demais a interação, talvez eu me atrevesse a dizer que se você der a coisa offline, como uma lição de casa, ele vai ser menos efetivo porque o aluno vai se desinteressar mais rapidamente.”, portanto o elemento da passividade segundo ele pode afetar o interesse do aluno e a supervisão de um tutor ou professor pode ser necessária para uma aplicação mais efetiva do Core Inc. Além destes elementos o professor não apresentou ideias de aplicação muito diferentes das já levantadas durante a construção desta dissertação.

No âmbito de correções e modificações o professor A encontrou diversos problemas quanto a representação que o Core Inc. busca fazer do ciclo de instrução. Deixando de lado suas críticas ao modelo de contextualização em si, segundo ele o próprio desenvolvimento da proposta tem falhas no desenvolvimento das animações.

Inicialmente o professor A fala que a maneira escolhida para representar as variáveis não é interessante, pois o aluno não vê imediatamente os nomes das variáveis e também não possui valor em todos os momentos da animação, já que o aluno só possui essas informações quando passa o mouse sob as caixas e visualiza essas informações adicionais. Segundo o professor essas caixas deveriam ter seu conteúdo visível durante toda a animação.

O professor aponta para a abstração de entrada e saída de dados do Core Inc. ao dizer que “a entrada e a saída estão no mesmo lugar na animação, isso não corresponde ao mecanismo elementar, não é? A entrada e saída são 2 buraquinhos separados”. Ou seja, no momento de abstração do Core Inc. a decisão tomada de simplificar a entrada e saída de dados na animação é considerada uma falha conceitual pelo professor A que necessita de correções.

Em um segundo momento o professor A também aponta falhas de representação com a memória do modelo de *von Neumann* dentro das animações, segundo ele “A memória, por sua vez é um barramento só, mas na animação elas são 2 coisas separadas, tem um por onde chega e um outro por onde a memória sai.”. Ou seja, quando o Core Inc. cria duas esteiras para o transporte de dados para a memória ele também comete falhas segundo o professor A, pois passa a sensação da existência de dois barramentos de memória, um conceito incorreto segundo o ciclo de *von Neumann*. Não só isto, mas quando a animação separa a memória do que está em tela na animação em si, não demonstrando ela em sua totalidade ele também passa um conceito errado segundo o professor A já que como ele diz “A coisa da memória apresentada de um jeito que a memória está fora do cenário. Não é assim.”, portanto a memória deveria ser um conjunto de objetos visível dentro da animação do Core Inc. e não algo que vem e vai para fora do espaço de animação como as caixas de variáveis funcionam no sistema atual.

Além destas críticas a representação das animações o professor reitera a passividade que encontrou no uso do Core Inc. e reforça que modificações seriam benéficas ao dar mais interações da animação com o aluno. Ele comenta em alguns momentos coisas como “apertar um botão para que a próxima coisa acontecesse” e “, se você tivesse, botões diferentes para ações diferentes e o aluno tivesse que acertar a próxima ação” e chega a detalhar exemplos deste tipo de situação, onde a animação é pausada e espera uma entrada de dados correta do aluno para que a animação avance, tais funcionalidade vão além do âmbito de algoritmos animados que o Core

Inc. propõe mas são ideias interessantes para corrigir a passividade apontada pelo professor A e são válidas para análise futura.

Finalizando a entrevista o professor A não teve problemas ou críticas com o procedimento de teste do software e entrevista realizados e como o entrevistador conduziu a conversa.

4.4.2 Análise Temática Entrevista Professor B

Tabela 4.2: Frases Entrevista Professor B

Modelo von Neumann	Totalmente, totalmente porque em arquitetura de computadores é essencial saber o ciclo básico.
Algoritmos Animados e Visualização de software	Eu adoro isso.
Algoritmos Animados e Visualização de software	Então eu sou totalmente favorável, assim eu acho que é uma ferramenta interessante.
Algoritmos Animados e Visualização de software	Depende se os conceitos representados na animação estão bem representados, né? E se essa visualização vai trazer mais entendimento ou dúvidas.
Algoritmos Animados e Visualização de software	Sabe, então assim se é uma simplificação da realidade, eu acho que funciona super não é?
Algoritmos Animados e Visualização de software	Se o professor adotar e sugerir para os alunos, porque hoje em dia existem várias dessas ferramentas.
Algoritmos Animados e Visualização de software	Então, depende de um pouco de instrução assim por parte de quem está lecionando.
Algoritmos Animados e Visualização de software	Eu acho que ela é benéfica, com certeza ela por si só ela não traz o conhecimento. Ela precisa de uma base por trás, isso sim.
Algoritmos Animados e Visualização de software	Eu acho que ela pode ser benéfica se ela tiver uma representação de fácil entendimento e correta do ponto de vista do que se quer passar.
Conteúdos adaptados à cultura do aluno	Isso é sensacional, né? Porque eu acho que é um grande chamariz para os alunos, né? Quase que uma gamificação do conhecimento, né? Até hoje ferramentas desse tipo são muito boas.
Conteúdos adaptados à cultura do aluno	Com certeza, com certeza, sem dúvida.
Impressões gerais Core Inc.	Eu achei interessante poder ter um ar retrô. De games mais retos assim, eu acho que isso é bacana.
Tutorial e Usabilidade Core Inc.	Sim, isso ajudou.
Tutorial e Usabilidade Core Inc.	Eu tenho um apontamento que seria assim, o tutorial não apresentou o cenário. É ele, precisaria apresentar o cenário né, aqui é a entrada e saída, aqui é onde acontece tal, coisa, etc.
Tutorial e Usabilidade Core Inc.	Mas ele foi fácil.

Temática e representação do Core Inc.	Então é. É pra mim não ficou muito claro, é a temática da empresa, sabe? Me parecia mais uma indústria igual uma indústria nuclear assim que os bichinhos têm que ficar fazendo lá, não pode falhar nunca, né?
Temática e representação do Core Inc.	Então me pareceu que se você tivesse puxado mais assim para temática sobre o que é uma indústria de execução de instruções né ao invés de uma empresa genérica, sim, eu acho que ia ser mais claro a relação.
Temática e representação do Core Inc.	Eu acho que foi importante, mas eu acho que faltou um pouquinho de foco, sabe, com mais foco você saberia o que escrever dos personagens, não ia ficar tão, tão genérico.
Temática e representação do Core Inc.	Até se poderia ter uma história no início, no vídeo ter a história. (...) Para que tenha uma narrativa em torno disso sabe, uma narrativa mais clara.
Temática e representação do Core Inc.	Você tem que dar uma pirada, os alunos vão falar assim, ah tá meio bobinho, mas aluno adora coisa bobinha assim, né? Nós adoramos coisas bobinhas.
Meio de abstração do Core Inc.	Eu acho que sim, acho que está claro.
Meio de abstração do Core Inc.	Tipo assim, nossa, vamos tentar ver se tem um refinamento que a gente consegue fazer, aí eu pegaria cada ação específica daria um zoom nela e tentaria identificar o que está acontecendo ali.
Meio de abstração do Core Inc.	Eu acho que ele poderia ser mais bem adaptado no cenário das animações.
Meio de abstração do Core Inc.	E eu fiquei um pouco perdido ali na organização dos componentes da animação. Então eu entenderia que a animação ela precisaria de uma melhor organização dos componentes.
Meio de abstração do Core Inc.	A posição visual e a organização dos componentes, eu colocaria de outra forma que seria mais benéfico.
Meio de abstração do Core Inc.	Então eu acho que eu, eu não sei assim, do ponto de vista de algoritmos e do modelo de Von Neumann, quando ele é ensinado, mas do ponto de vista de arquitetura de computadores alguns ajustes poderiam trazer ele muito, muito próximo para realidade de uma arquitetura simplificada, sabe? Um processador genérico qualquer e aí eu acho que teria um potencial maior ainda, sabe, de impacto da ferramenta.
Meio de abstração do Core Inc.	Nesse sentido é buscar essa harmonia entre o modelo de Von Neumann e arquitetura simplificada.
Meio de abstração do Core Inc.	E tudo bem ter abstrações, é igual você usar analogias, né? Então, eu acho que faz muito sentido.
Aplicação do Core Inc. em sala de aula.	E inclusive, o visualizador eu acho que ele é muito mais útil para a arquitetura do que para algoritmos, porque o pensamento algorítmico eu não vejo assim ele tão dependente daqueles detalhes arquiteturais do barramento de memória e tudo, mas para arquitetura sim, totalmente.

Aplicação do Core Inc. em sala de aula.	Você pode usar o como um objeto de transição, né? Entre várias disciplinas, porque o cara de algoritmos ele pode ensinar sobre algoritmos, o cara de arquitetura pode explicar sobre a arquitetura usando a mesma coisa. Então esse objeto poderia servir como essa ponte, né? De troca de conhecimento entre arquitetura e algoritmos e talvez compiladores mais pra frente, quem sabe?
Aplicação do Core Inc. em sala de aula.	A simulação do mips, por exemplo, a simulação do mips eu acho que seria totalmente passível assim de ser mostrado o que está rolando.
Aplicação do Core Inc. em sala de aula.	Então talvez uma ferramenta dessa seria muito útil assim não é porque isso daqui é o que vai inspirar a nossa execução.
Modificações e correções Core Inc.	A entrada e saída ali está totalmente desconectado de memória, e não é o que acontece. A entrada e saída acontece pela memória.
Modificações e correções Core Inc.	Você ter o controle feito por um único personagem, eu acho que isso talvez não seja tão benéfico porque, afinal de contas, os controles são especializados.
Modificações e correções Core Inc.	Então se poderia ter mais do que um personagem ou metade da tela sendo feito por um personagem, metade sendo feito por outro, para dar essa impressão de especificidade não é, de o controle está acontecendo.
Modificações e correções Core Inc.	Você tem uma parte que gerencia, só busca de instrução, outra parte só faz instrução e vai para outro lugar, outra parte que isso outra parte que aquilo.
Modificações e correções Core Inc.	O rolê todo do teu vídeo eu achei que podia ter uma musiquinha. (...) Alguma coisa que desse a entender que você tinha algo que passasse alguma emoção.
Modificações e correções Core Inc.	Isto, mais a direita é o textinho dos personagens, você poderia talvez quando a pessoa não quer ajuda e tal, ali você poderia utilizar com carga útil, sabe? Poderia, por exemplo, a mostra o que você quer mostrar e depois aquilo fica fixado ali na direita. Então, a animação está acontecendo, mas eu sei a que se refere aquela animação.
Modificações e correções Core Inc.	Se você for ver, tem uma boa parte da tela que está inutilizada ali que você fala em qualquer momento você pode pedir ajuda tal.
Modificações e correções Core Inc.	Ai eu não sei quais são as limitações do seu do seu modelo (...) é importante você deixar claro quais são as limitações em algum lugar.
Modificações e correções Core Inc.	E deixa todo em português tchê, metade em português, metade inglês. Lá em cima código, lá em baixo input. Por que não é código de entrada? Escolhe uma língua e vai até o fim com aquela língua.
Modificações e correções Core Inc.	Poderia ser uma extensão interessante, sabe, talvez você escrever código em assembly.

Modificações e correções Core Inc.	Outra coisa que eu fiquei me perguntando é se não faria sentido você traduzir o código de Pascal em um código assemblylike.
Avaliação do processo de entrevista	Tem algumas partes que eu achei assim, parece que será que está tentando me induzir alguma coisa? Ou ele não estava sendo preciso suficiente, não é?
Avaliação do processo de entrevista	Eu tenho alguns pensamentos confusos sobre se precisaria ser síncrono ou não.
Avaliação do processo de entrevista	Mas eu achei que foi isso, foi muito bom o processo.

O professor B respondeu ao início da entrevista de forma sucinta que em sua área de arquitetura de computadores e a disciplina que ministra saber o ciclo básico ou ciclo de instrução junto da arquitetura de von Neumann está totalmente relacionado ao seu trabalho e o conhecimento que os alunos precisam adquirir.

Inicialmente ele respondeu sobre as ferramentas de visualização de software e algoritmos animados com ênfase ao dizer que “Eu adoro isso.” e “Então eu sou totalmente favorável, assim eu acho que é uma ferramenta interessante.”, manifestando uma opinião positiva em relação a este tipo de abordagem, também ressaltando que simplificações de como funcionam os conceitos podem ser benéficas ao complementar “Sabe, então assim se é uma simplificação da realidade, eu acho que funciona super”. Contudo, ressalvas foram feitas ao longo de sua explicação, onde ele diz que “Depende se os conceitos representados na animação estão bem representados”, sendo que problemas nessa representação podem causar mais dúvidas do que entendimento.

Além disto ele ressalta a importância do suporte do professor ou educador no uso da ferramenta ao dizer que “Se o professor adotar e sugerir para os alunos, porque hoje em dia existem várias dessas ferramentas.” e “Então, depende de um pouco de instrução assim por parte de quem está lecionando.” pois o número de ferramentas existentes hoje com a internet é muito grande e pode sobrecarregar o aluno, além de que ele diz que a ferramenta por si só não é suficiente quando ele ressalta que “(...) com certeza ela por si só ela não traz o conhecimento. Ela precisa de uma base por trás, isso sim.”.

Portanto o professor B considera que estas ferramentas podem ser benéficas se possuírem representação simplificadas da realidade de fácil entendimento e conceitualmente acuradas, contudo, ele também aponta que a ferramenta por si só não é suficiente e o suporte do professor é necessário durante o processo, sendo assim a ferramenta por si só não gera o conhecimento, mas pode dar suporte quando o professor ministra os conteúdos que a ferramenta aborda.

Quanto ao tema de conteúdos adaptados a cultura do aluno, como o teoria da cognição situada que embasa esse trabalho indica, o professor B também é enfático ao dizer que “Com certeza, com certeza, sem dúvida.” Elas são relevantes e interessantes no ambiente de ensino. Além disso ele acrescenta comparações com videogames ao longo de sua fala como “Quase que uma gamificação do conhecimento, né? Até hoje ferramentas desse tipo são muito boas.”, demonstrando já inicialmente na entrevista que ele identifica os elementos visuais que o Core Inc. utiliza de videogames e pixel art também como benéficos para a construção de ferramentas de visualização de software ao dizer que “Isso é sensacional, né? Porque eu acho que é um grande chamariz para os alunos, né? (...) Até hoje ferramentas desse tipo são muito boas.”.

Quando indagado sobre o Core Inc. especificamente o professor B fez vários apontamentos profundos já em suas impressões gerais, assim como o professor A dando informações valiosas que seriam perguntadas posteriormente pelo entrevistador, tais informações são descritas nos demais temas discutidos adiante nesta análise. Entretanto, um dos apontamentos gerais feitos

pelo professor B é interessante de mencionar, quando ele diz “Eu achei interessante poder ter um ar retrô. De games mais retros assim, eu acho que isso é bacana.”. Mais uma vez identificando os elementos de videogames e o estilo de arte adotado pelo Core Inc., demonstrando que nesta entrevista um dos objetivos do software que é capturar esses elementos visuais de jogos parece ter sido cumprido, ao mesmo tempo que o professor B acha boa essa característica visual adotada.

Quanto aos tutoriais e a usabilidade do Core Inc. o professor B considerou o tutorial útil e o software fácil e usar ao dizer coisas como “Sim, isso ajudou.” e “Mas ele foi fácil.”. Contudo, o professor B criticou a falta de apresentação dos elementos da animação durante o tutorial, segundo que ele “Eu tenho um apontamento que seria assim, o tutorial não apresentou o cenário.” Pois a apresentação desses elementos pode não ser essencial para o professor, porém são importantes para os alunos.

A temática do Core Inc. e como ele representa seu cenário e personagens também foi comentado durante a entrevista pelo professor B. Apesar do professor ter demonstrado interesse pela representação visual do Core Inc. e em suas palavras a temática “retro” ele acha que a história que embasa a criação dos personagens e mundo pode ser melhor desenvolvida, em suas palavras: “Eu acho que foi importante, mas eu acho que faltou um pouquinho de foco, sabe, com mais foco você saberia o que escrever dos personagens, não ia ficar tão, tão genérico.”. Ele reforça durante a entrevista de diversas formas como a história deveria ser mais desenvolvida ao dizer coisas como “É pra mim não ficou muito claro, é a temática da empresa” e “Até se poderia ter uma história no início, no vídeo ter a história. (...) Para que tenha uma narrativa em torno disso sabe, uma narrativa mais clara.”.

No fim as impressões do professor B são que os visuais e a temática são interessantes, mas falta desenvolvimento para que não pareça tão genérico. Em suas palavras ele diz “Você tem que dar uma pirada, os alunos vão falar assim, ah tá meio bobinho, mas aluno adora coisa bobinha assim, né? Nós adoramos coisas bobinhas.”, ele chegou a desenvolver no meio da entrevista exemplos adicionais que não são demonstrados aqui na íntegra, em que ele comenta ideias de como a história poderia ser e exemplos de videogames antigos e como eles contam suas histórias através de imagens e texto estáticos, de maneira similar ao que o Core Inc. já faz em sua introdução, mas segundo ele de maneira mais interessante e detalhada.

Sobre a forma com que os conceitos do modelo de *von Neumann* e o ciclo de instrução foram abstraídos no Core Inc. o professor B falou o seguinte sobre a compreensão das animações: “Eu acho que sim, acho que está claro.” e “tudo bem ter abstrações, é igual você usar analogias, né? Então, eu acho que faz muito sentido.”, demonstrando que não há em sua opinião um problema com a identificação do que cada personagem faz em suas animações. Segundo o professor B caso fosse possível um refinamento ele disse “eu pegaria cada ação específica daria um zoom nela e tentaria identificar o que está acontecendo ali.” De forma a detalhar melhor cada ação do personagem, porém o professor B deixou claro que isso não parece uma prioridade para ele, pois existem outros problemas com a abstração.

O principal problema para o professor B são falhas na abstração do modelo de *von Neumann*, ao Core Inc. errar em abstrair conceitos do modelo e também de uma arquitetura simplificada, segundo o professor fazendo um pouco de ambos, mas não sendo nenhum dos dois atualmente, o que causa problemas no aprendizado do aluno. Ele diz que o modelo “Eu acho que ele poderia ser mais bem adaptado no cenário das animações.” e que “fiquei um pouco perdido ali na organização dos componentes da animação. Então eu entenderia que a animação ela precisaria de uma melhor organização dos componentes.”.

Ainda neste tema o professor B reforça que é preciso fazer ajustes na posição dos componentes e encontrar um balanço sobre o modelo que o Core Inc. vai representar. Em suas

palavras “Nesse sentido é buscar essa harmonia entre o modelo de von Neumann e arquitetura simplificada.”.

No tópico da possível aplicação do Core Inc. em sala de aula o professor B considera que o Core Inc. pode ser muito útil para as disciplinas de arquitetura de computadores, ele diz que “o visualizador eu acho que ele é muito mais útil para a arquitetura do que para algoritmos, porque o pensamento algorítmico eu não vejo assim ele tão dependente daqueles detalhes arquiteturais do barramento de memória e tudo, mas para arquitetura sim, totalmente.”. Ele acrescenta que ele pode ser usado em múltiplos momentos em sala de aula para unir conhecimentos de disciplinas diferentes, ele exemplifica: “Você pode usar o como um objeto de transição, né? Entre várias disciplinas, porque o cara de algoritmos ele pode ensinar sobre algoritmos, o cara de arquitetura pode explicar sobre a arquitetura usando a mesma coisa.”.

Adicionando a isto o professor fala sobre modelos que poderiam ser adaptados com o Core Inc. em arquitetura e reforça sua utilidade ao comentar que “A simulação do mips, por exemplo, a simulação do mips eu acho que seria totalmente passível assim de ser mostrado o que está rolando.” e “Então talvez uma ferramenta dessa seria muito útil assim não é porque isso daqui é o que vai inspirar a nossa execução.”. Com isto o professor B dá informações valiosas até aqui, onde o Core Inc. tinha como objetivo ser utilizado somente em sala de aula durante o ensino do ciclo de instrução em disciplinas introdutórias de algoritmos, mas em sua opinião também pode ser utilizado em disciplinas de arquitetura de computadores caso passe por correções necessárias, estas que ele comenta logo a seguir.

Ao longo da entrevista o professor B fez comentou uma extensa lista de problemas que devem ser corrigidos no Core Inc. para que em sua opinião ele se torne adequado para o uso com alunos em sala de aula, além disto também fez sugestões de modificações que poderiam ser adicionadas para melhorar o software. Uma das principais críticas do professor B foi na abstração dos conceitos básicos para a animação, ele lista problemas na representação da entrada e saída de dados ao dizer que “A entrada e saída ali está totalmente desconectado de memória, e não é o que acontece. A entrada e saída acontece pela memória.”, junto disto ele também comenta o modo com que cada atividade é dividida na animação e como isso poderia ser melhor representado, ele explica que “Você ter o controle feito por um único personagem, eu acho que isso talvez não seja tão benéfico porque, afinal de contas, os controles são especializados.”, ou seja o protagonismo dado a personagem Lucy durante as animações não parece benéfico ao professor, pois não reflete a realidade com acurácia. O professor B ainda desenvolve mais sobre o tema explicando meios possíveis de corrigir a animação ao dividir as tarefas entre personagens diferentes.

Além dos problemas de representação o professor B também comenta de modificações no modo com que o enredo do Core Inc. é demonstrado ao dizer que “O rolê todo do teu vídeo eu achei que podia ter uma musiquinha.”, justificando falta de emoção ao transmitir a história para o aluno e a própria falta de som confundindo-o durante a execução do vídeo introdutório. Isso alia-se as próprias críticas anteriores do professor B a falta de detalhes e desenvolvimento na história do Core Inc. através do tutorial e vídeo de apresentação inicial.

O entrevistado também comenta sobre como nota que parte da tela responsável por mostrar informações adicionais sobre a animação ao aluno é uma parte praticamente inútil da tela. Segundo ele o espaço mal é utilizado caso o aluno não se interesse em interagir com a animação, ele comenta que “ali você poderia utilizar com carga útil, sabe? Poderia, por exemplo, mostrar o que você quer mostrar e depois aquilo fica fixado ali na direita.”, ou seja, o professor B gostaria que informações adicionais sobre a animação que está em execução fossem adicionadas nesta área ociosa enquanto o aluno não interage com a animação, já que na versão atual do Core Inc. ela dá instruções de como utilizar o software.

Outra sugestão adicional do professor é mostrar um indicador do comando executado no momento: “você ter uma setinha mostrando com a instrução em execução. Poderia ter uma setinha do lado do código mostrando” e demonstrar no portal as limitações do software e quais comandos ele suporta. Isto já é feito de forma sucinta em uma das páginas do portal, porém o professor B não explorou essa região do software durante o teste, ainda que ele diga que isto esta sugestão demonstra que as limitações do software podem ficar disponíveis em um local mais visível do portal.

O professor B também critica a mistura de termos usados no Core Inc. ao dizer que “E deixa todo em português tchê, metade em português, metade inglês. Lá em cima código, lá em baixo input. Por que não é código de entrada? Escolhe uma língua e vai até o fim com aquela língua.”, portanto ele recomenda que todos os termos do software estejam em português, ele inclusive comenta que com a pouca exposição que o enredo tem atualmente nem mesmo o nome em inglês “Core Inc.” se justifica.

Uma sugestão adicional do professor B é a adição da possibilidade de suportar assembly dentro do Core Inc., seja traduzindo um código de alto nível para o mesmo ou permitindo a entrada de código assembly para ser animado, como ele diz “Poderia ser uma extensão interessante, sabe, talvez você escrever código em assembly.”. Esta funcionalidade parece conversar diretamente com sua indicação que seria útil em sala de aula durante disciplinas de arquitetura de computadores. Tal apontamento não chega a ser uma crítica, mas sim uma indicação de possível trabalho futuro.

Quanto à execução da entrevista o professor ficou confuso quanto ao questionário do método TAM aplicado anteriormente, que segundo ele poderia ser revisado alguns termos que pareciam induzi-lo a determinadas respostas e dúvidas se o processo precisava ser realmente síncrono. Ambas as dúvidas foram discutidas com o entrevistador que comentou sobre o questionário do método TAM utilizado e que sim poderia ser revisado certas perguntas e a ideia de realizar o processo de forma assíncrona foi rapidamente descartada pelas desvantagens que a falta de interação entre entrevistado e entrevistador geraria. Com exceção destes dois apontamentos o professor disse que “Mas eu achei que foi isso, foi muito bom o processo.” encerrando a entrevista sem demais críticas ao processo.

4.4.3 Análise Temática Entrevista Professora C

Tabela 4.3: Frases Entrevista Professora C

Modelo von Neumann	Da minha pesquisa em si, ela está fora, mas do meu trabalho em sala de aula ela está dentro.
Modelo von Neumann	Se a gente for pensar em na arquitetura de von Neumann, sempre quando eu começo a trabalhar a introdução a algoritmos, já eu faço essa analogia com essa arquitetura para que a os alunos consigam compreender.
Modelo von Neumann	Então você ter essa noção dessa arquitetura de von Neumann é importante porque facilita a compreensão na programação em si, né? O que você está fazendo.
Algoritmos Animados e Visualização de software	Eu sou 100% adepta e concordo com esse tipo de solução.
Algoritmos Animados e Visualização de software	Na minha opinião, sendo professora da área de programação aí há quase 20 anos, o visual nesse caso é importante porque é um conteúdo muito abstrato, é fora da nossa realidade.

Algoritmos Animados e Visualização de software	Isso facilita a compreensão do aluno, do conceito em si.
Algoritmos Animados e Visualização de software	Então às vezes eu estou desenhando no quadro, porque quando a gente não está trabalhando com esse tipo de software eu estou fazendo analogias.
Conteúdos adaptados à cultura do aluno	Eu acho que sim.
Conteúdos adaptados à cultura do aluno	Teve uma parte de um livro dele que eu tava lendo lá, que ele traz muito o exemplo dos agricultores, então ele traz muito da realidade de termos e palavras que os agricultores estão acostumados para poder trabalhar alfabetização com eles.
Conteúdos adaptados à cultura do aluno	Então, se a gente consegue desenvolver um software que traga a realidade de um grupo de alunos(...)
Conteúdos adaptados à cultura do aluno	(...) fazer uma analogia com a programação, que é o nosso caso, que é muito abstrato. Eu acho que isso vai facilitar ainda mais a compreensão.
Conteúdos adaptados à cultura do aluno	Um dos autores que eu estudo agora que trabalha extensão é Paulo Freire.
Impressões gerais Core Inc.	Para ser sincera, eu lembrei do Iron Ears. Eu já sei, o autor é o mesmo.
Impressões gerais Core Inc.	Mas é eu acho que eu me senti familiarizada mais do que se fosse o primeiro contato. Porque eu conheci o Iron Ears e eu conheço a lógica de pensamento de vocês.
Impressões gerais Core Inc.	A única coisa que eu senti ali, eu não sei qual que é o objetivo de vocês nesse sentido, eu senti falta do som. Sabem de alguma narrativa.
Tutorial e Usabilidade Core Inc.	Eu achei bem clean, bem fácil, do meu ponto de vista, tá tudo ali e não é uma tela cheia de botão, monte de coisara.
Tutorial e Usabilidade Core Inc.	Está tudo muito claro, ela está bem dividida. No meu ponto de vista, onde que estão as informações, né? Em locais bem definidos, se mantém o padrão.
Tutorial e Usabilidade Core Inc.	Você tem o tutorial e ele é claro e explica.
Tutorial e Usabilidade Core Inc.	Eu acho que mesmo sem tutorial a gente conseguiria no primeiro momento ia se bater um pouco mas depois ia pegar.
Tutorial e Usabilidade Core Inc.	O tutorial facilita o entendimento inicial, mas a tela ela é tão pra mim ela está tão simples, a usabilidade, ela tá tão simples que vai ter gente que vai nem vai ver o tutorial e vai usar direto.
Temática e representação do Core Inc.	Eu acho interessante, porque eu penso assim, a gente usar o software dentro do contexto da universidade para o ensino de graduação, eu acho bem interessante.
Temática e representação do Core Inc.	Porque a gente trabalha atuando em sistemas que vão estar dentro de fábricas ou vem alunos que a gente sabe dos que trabalham em fábricas, têm a noção, tem a ideia, o conhecimento, a prática.

Meio de abstração do Core Inc.	Que nem quando eu vi a primeira vez eu me bati um pouco para entender o que estava acontecendo.
Meio de abstração do Core Inc.	Eu não sei se para um aluno iniciante se ele pegaria com rapidez o que está acontecendo, por isso que eu achava interessante ter a narrativa ali.
Meio de abstração do Core Inc.	E eu gosto desse visual bem, já digo vintage né? E é da minha época, da época dos Ataris. Que é uma tendência que está voltando até para os adolescentes aí que estão retomando esse tipo de tela e tal.
Meio de abstração do Core Inc.	Eu acho que atinge porque você consegue acompanhar certo, porque ele vai mostrando e aí, você vai visualizando a dinâmica.
Meio de abstração do Core Inc.	Então isso acho que fica bem claro que você consegue identificar cada instrução que ele executa na animação.
Aplicação do Core Inc. em sala de aula.	Eu acho que em sala de aula eu o utilizaria nas minhas aulas, primeiro nas minhas aulas introdutórias.
Aplicação do Core Inc. em sala de aula.	É na hora que eu fosse explicar a questão da arquitetura de von Neumann, que eu abordo nessa disciplina.
Aplicação do Core Inc. em sala de aula.	Invés de eu ficar escrevendo o quadro, eu pegaria os exemplos do Core Inc. e já ia mostrando pela execução ali pela animação.
Aplicação do Core Inc. em sala de aula.	Iria trabalhar os algoritmos mais complexos com eles e passaria e pediria para que eles desenvolvessem os algoritmos deles usando o Core Inc. e testando a animação.
Aplicação do Core Inc. em sala de aula.	Testem para verificar qual é a animação que o teu código vai gerar no Core Inc. para você entender o que você fez e como que está funcionando lá em baixo nível lá na máquina.
Aplicação do Core Inc. em sala de aula.	Discutir em sala, mas eles verem a dinâmica do algoritmo que eles próprios desenvolverem sendo animados e vendo como ocorre, é diferente, eu acho que seria interessante.
Aplicação do Core Inc. em sala de aula.	Fora que a gente pode ir para um mesmo algoritmo para o mesmo problema, gerar algoritmos diferentes, correto? E que jogando no Core Inc. ele vai gerar animações diferentes e a gente pode fazer uma análise dessas animações.
Aplicação do Core Inc. em sala de aula.	Então seria um outro tipo de exercício que eu poderia fazer e que sem o Core Inc. eu não consigo fazer.
Modificações e correções Core Inc.	Aí quando começa a animação, por ser a primeira vez que eu estou vendo a animação, eu fico um pouco perdida. Onde que estão as coisas? Se tivesse uma narrativa bem curta, não precisava ser longa ali.
Modificações e correções Core Inc.	Isso não só para o meu entendimento, pra eu que enxergo me situar, mas para quem não enxerga e vai ouvir e vai conseguir também acompanhar.
Modificações e correções Core Inc.	Eu acho que facilita tanto principalmente para quem é iniciante.

Modificações e correções Core Inc.	E você dá a opção de ligar ou desligar a narrativa, porque quando ele tiver já habituado com o software, tem gente que nem vai querer ouvir.
Modificações e correções Core Inc.	Mas se eu, por exemplo, eu professora quisesse usar o software, num projeto de inclusão digital, com pessoas idosas ou sei lá, até mesmo adolescentes que não têm essa vivência de fábrica de contexto, seria interessante eu poder também ter outro contexto para fazer as animações.
Modificações e correções Core Inc.	Sei lá, o contexto da escola, um contexto do supermercado, não sei, mas um contexto, mais cotidiano sabe que daí o software poderia ser usado, o teu software poderia ser usado em contextos diferentes, não só na graduação, na universidade.
Modificações e correções Core Inc.	Então eu ter temas variados aí que eu pudesse escolher. Poderia manter os mesmos personagens, mudaria só a localidade, o contexto, o ambiente, sabe.
Avaliação do processo de entrevista	Não, para mim foi bem tranquilo. Eu gostei desse processo de interagir com o software primeiro.
Avaliação do processo de entrevista	A gente vai pensando sobre o software. Quando chega na entrevista com você, eu já sei do que nós estamos conversando.
Avaliação do processo de entrevista	Mas é um momento anterior, a priori, um pouquinho antes, mas que me ajuda a me dar tranquilidade em responder tuas perguntas com mais veracidade do que eu estou achando.

A professora C atua como pesquisadora na área de informática na educação e ministra aulas de programação e algoritmos básico. Segundo ela o modelo de *von Neumann* abordado no Core Inc. não está totalmente inserido em seu trabalho, pois ela diz que “Da minha pesquisa em si, ela está fora, mas do meu trabalho em sala de aula ela está dentro.”. Mesmo o modelo estando parcialmente inserido em seu dia a dia, a professora ressalta que no aprendizado este conteúdo é relevante, pois explica que “Então você ter essa noção dessa arquitetura de *von Neumann* é importante porque facilita a compreensão na programação em si, o que você está fazendo.”, ela ainda explica como utiliza tais conteúdos em sala de aula, ao dizer que “(...) sempre que eu começo a trabalhar a introdução de algoritmos, já eu faço essa analogia com essa arquitetura(...)”. Portanto, compreende-se que em sua vida de educadora a professora C está completamente inserida dentro do público alvo do software e considera extremamente relevante a arquitetura de *von Neumann* que é abordada pelo Core Inc..

Quanto a opinião da professora C sobre solução de algoritmos animados e visualização de software ela foi enfática ao dizer que “Eu sou 100% adepta e concordo com esse tipo de solução.”, ela explica sua posição quanto a utilização deste tipo de ferramenta com sua experiência em sala de aula ao justificar que “Na minha opinião, sendo professora da área de programação aí há quase 20 anos, o visual nesse caso é importante porque é um conteúdo muito abstrato, é fora da nossa realidade.”, um dos motivos apontados pela professora C é os benefícios ao aluno em si pois segundo ela “Isso facilita a compreensão do aluno, do conceito em si.”, contudo ela faz ressalvas que mesmo este tipo de ferramenta pode não ser adequada a todos os alunos, pois cada indivíduo possui formas diferentes de aprender o mesmo conteúdo.

Além desta característica, a professora parece apontar praticidade como um dos fatores no uso deste tipo de software visual, ao falar que “Então às vezes eu estou desenhando no quadro, porque quando a gente não está trabalhando com esse tipo de software eu estou fazendo

analogias.”, demonstrando que tais tipo de analogias que o próprio Core Inc. busca fazer já são feitas pela professora C em sala de aula, porém com mais trabalho, pois não é automatizado com as ferramentas de visualização.

Quando indagada sobre conteúdos adaptados à cultura dos alunos, a professora C passou a fazer conexões diretas com seus temas de pesquisa, ao dizer “Um dos autores que eu estudo agora que trabalha extensão é Paulo Freire e o Paulo Freire fala muito do tema gerador.”, ela comenta sobre Paulo Freire e seu trabalho de alfabetização ao citar que “Teve uma parte de um livro dele que eu tava lendo lá, que ele traz muito o exemplo dos agricultores, então ele traz muito da realidade de termos e palavras que os agricultores estão acostumados para poder trabalhar alfabetização com eles.”, comparando diretamente o tipo de ensino contextualizado para adultos de Paulo Freire e os softwares adaptados à cultura dos alunos, sendo assim a professora C afirma que “Eu acho que sim.”, estabelecendo uma posição favorável ao tipo de solução que o Core Inc. busca ser.

Se aprofundando mais neste tema, a professora C comenta como essa contextualização pode fazer diferença ao dizer “Então, se a gente consegue desenvolver um software que traga a realidade de um grupo de alunos(...)” e “(...)fazer uma analogia com a programação, que é o nosso caso, que é muito abstrato. Eu acho que isso vai facilitar ainda mais a compreensão.”. Ao reforçar sua opinião positiva quanto a este tipo de ferramenta também foi possível notar a forte conexão entre suas opiniões e a área de estudo da professora C, pois mais à frente na entrevista a mesma pediu informações sobre o tipo de teoria de ensino-aprendizagem que embasava o trabalho e comentou sobre temas tangentes a proposta do Core Inc., conteúdos não abordados diretamente na análise pois não dizem respeito o propósito direto desta, contudo demonstram aparente domínio da professora C sobre o tema de conteúdos adaptados à cultura no ambiente de computação.

Quanto ao software em si, as impressões gerais da professora C foram inicialmente diferentes dos demais entrevistados, pois ela cita um trabalho anterior realizado pelo autor deste projeto: “Para ser sincero, eu lembrei do Iron Ears. Eu já sei, o autor é o mesmo.”. Entende-se que as similaridades são notadas pelas características visuais semelhantes entre o projeto Iron Ears (da Rosa et al., 2020) e o Core Inc., além de que apesar de propostas de software e temas diferentes, ambas soluções trabalham no âmbito da visualização de software. A professora C adiciona ainda que suas primeiras impressões foram influenciadas por este conhecimento prévio do trabalho do autor ao dizer que “Mas eu acho que eu me senti familiarizada mais do que se fosse o primeiro contato. Porque eu conheci o Iron Ears e eu conheço a lógica de pensamento de vocês.”. Além disto a professora C sentiu falta de elementos narrativos mais presentes no Core Inc. além de efeitos sonoros, ao relatar que “A única coisa que eu senti ali, eu não sei qual que é o objetivo de vocês nesse sentido, eu senti falta do som. Sabem de alguma narrativa.”, neste momento a professora C também traçou comparativos com o software Iron Ears, pois este possui ambos narrativa e efeitos sonoros mais presentes e em maior quantidade que o Core Inc..

A usabilidade do Core Inc. e seu tutorial também foi um dos temas conversados durante a entrevista onde a professora C demonstrou opiniões muito positivas sobre estas características ao dizer que “Está tudo muito claro, ela está bem dividida. No meu ponto de vista, onde que estão as informações, né? Em locais bem definidos, se mantém o padrão.” e “Eu achei bem clean, bem fácil, do meu ponto de vista, tá tudo ali e não é uma tela cheia de botão, monte de coisrada.” ela ressalta o bom posicionamento dos elementos em tela e a clareza com que são apresentados, quanto ao tutorial a professora é sucinta ao dizer que “Você tem o tutorial e ele é claro e explica.” não apontando problemas com os mesmos. Ela ainda adiciona que em sua opinião a tela é muito autoexplicativa e intuitiva ao dizer que “Eu acho que mesmo sem tutorial a gente conseguiria no primeiro momento ia se bater um pouco mas depois ia pegar” e “(...)ela tá

tão simples que vai ter gente que vai nem vai ver o tutorial e vai usar direto.” falando que talvez o tutorial nem fosse necessário em sua opinião para o uso adequado da ferramenta. Enquanto a retirada dos elementos de tutorial não esteja nos planos de desenvolvimento do Core Inc. tal resposta sobre sua usabilidade é benéfica para identificar o quão bem desenvolvido o Core Inc. já se apresenta nestes quesitos segundo a professora C.

Nas perguntas anteriores sobre softwares contextualizados com a cultura dos alunos a professora C já havia demonstrado opiniões favoráveis a este tipo de abordagem e quando se trata da contextualização do Core Inc. em específico com seus elementos de representação e tema ela também considera uma boa escolha, ao dizer que “a gente usar o software dentro do contexto da universidade para o ensino de graduação, eu acho bem interessante.” e “Porque a gente trabalha atuando em sistemas que vão estar dentro de fábricas ou vem alunos que a gente sabe dos que trabalham em fábricas, têm a noção, tem a ideia, o conhecimento, a prática.”, ela relata que a temática do Core Inc. conversa com o dia a dia dos seus alunos, este que já trabalham ou começam a trabalhar em indústrias e fábricas ainda durante o curso. Este apontamento é interessante, pois o Core Inc. embasa sua contextualização em elementos mais gerais e facilmente identificáveis para um público amplo, contudo aqui a professora relata que especificamente a temática de fábricas também interage de forma mais próxima aos alunos de computação, segundo sua perspectiva.

Quanto ao método utilizado para transformar o conteúdo de algoritmos e modelo de *von Neumann* em animações a professora C aponta que não é fácil compreender em um primeiro momento o que as animações representam ao dizer que “Que nem quando eu vi a primeira vez eu me bati um pouco para entender o que estava acontecendo.”, ela ainda reforça que isso é a percepção dela e a recepção de um aluno que não tem conhecimento prévio dos conteúdos pode ser ainda mais difícil quando ela explica que “Eu não sei se para um aluno iniciante se ela pegaria com rapidez o que está acontecendo(...)”. Dado estas problemáticas que a professora C identificou, ela acredita que as animações que foram elaboradas para cada ação dos personagens são de fácil atendimento, pois ela diz que “Eu acho que atinge porque você consegue acompanhar certinho, porque ele vai mostrando e aí, você vai visualizando a dinâmica.” e também “Então isso acho que fica bem claro que você consegue identificar cada instrução que ele executa na animação.”. Por fim, a professora adicionou em sua resposta aos questionamentos deste tópico impressões sobre o estilo de arte escolhido para o Core Inc., se referindo a pixel art ela elabora que “E eu gosto desse visual bem, já digo vintage né? E é da minha época, da época dos Ataris. Que é uma tendência que está voltando até para os adolescentes aí que estão retomando esse tipo de tela e tal.”, isto demonstra que o objetivo da escolha da arte é cumprido segundo a ótica da professora C, que era obter o visual de jogos mais antigos, enquanto que seu apontamento sobre o retorno deste tipo de arte em jogos também é correto, como demonstrado no referencial teórico desta dissertação.

Devido a experiência recente e extensa do professora C ministrando conteúdos relacionados a ferramenta Core Inc. ela trouxe durante a entrevista apontamentos pertinentes de serem citados quanto a aplicação do Core Inc. em sala de aula, segundo ela “Eu acho que em sala de aula eu o utilizaria nas minhas aulas, primeiro nas minhas aulas introdutórias”, especificamente ela fala de utilizar o software especificamente durante a explicação do modelo de *von Neumann*, o que é esperado devido ao conteúdo que o software aborda, contudo a professora C descreve mais detalhadamente qual seria seu processo de utilização durante as aulas, como nas falas a seguir: “Invés de eu ficar escrevendo o quadro, eu pegaria os exemplos do Core Inc. e já ia mostrando pela execução ali pela animação”; “(...)pediria para que eles desenvolvessem os algoritmos deles usando o Core Inc. e testando a animação.”; e “(...)jogando no Core Inc. ele vai gerar animações diferentes e a gente pode fazer uma análise dessas animações.”. Nesta etapa

notou-se que a professora C parecia confortável em encontrar métodos de inserir o Core Inc. em sua metodologia de ensino, segundo ela de forma a expandir as possibilidades de ensino dos alunos e aprimorar a velocidade em que o conteúdo pode ser repassado, ao tirar a necessidade do uso do quadro negro para desenhar ilustrações durante a aula, por exemplo. Além disso a professora C prossegue dizendo que “Discutir em sala, mas eles verem a dinâmica do algoritmo que eles próprios desenvolverem sendo animados e vendo como ocorre, é diferente, eu acho que seria interessante.” Citando pela segunda vez a ideia de análise das animações geradas pelo Core Inc., permitindo analisar os algoritmos desenvolvidos e o funcionamento da arquitetura por uma nova ótica. Em relação a estes exercícios de análise e discussões de animações a professora C explica que este tipo de exercício só seria possível com a utilização do Core Inc. quando ela diz “Então seria um outro tipo de exercício que eu poderia fazer e que sem o Core Inc. eu não consigo fazer.”, esta análise levanta possibilidades interessantes, justamente por apresentar propostas diferentes de utilização do Core Inc. que podem servir para a elaboração de um plano de utilização do software, não só isto, como também permitindo o desenvolvimento de atividades únicas que expandam as possibilidades e profundidade da abordagem dos conteúdos que o Core Inc. busca demonstrar.

Apesar da professora C demonstrar impressões positivas em relação ao Core Inc. ela propõe algumas modificações para deixar o software mais flexível e claro. Segundo a mesma a narrativa precisa ser incrementada, ela comenta que “Aí quando começa a animação, por ser a primeira vez que eu estou vendo a animação, eu fico um pouco perdida. Onde que estão as coisas? Se tivesse uma narrativa bem curta, não precisava ser longa ali.”, onde a narrativa mais presente exerceria um papel ativo na melhor explicação dos elementos em tela, segundo ela “Eu acho que facilita tanto principalmente para quem é iniciante. “. A narrativa está presente hoje no Core Inc. de maneira mais detalhada, mas não de maneira explícita, já que dependem de ações específicas do usuário. Por fim, ainda sobre a narrativa a professora C comenta que a narrativa não é interessante a todo momento e sim nos estágios iniciais do uso do software, portanto ela fala que “E você dá a opção de ligar ou desligar a narrativa, porque quando ele tiver já habituado com o software, tem gente que nem vai querer ouvir.”, trazendo a ideia de dar opção ao aluno de níveis de narrativa dentro do Core Inc. atendendo as diversas fases do uso do software por parte do aluno, é uma solução interessante para evitar que o aluno fique entediado com narrativa em excesso ao mesmo tempo que a fornece para os alunos interessados.

Ela também acha necessário a adição de sons no vídeo introdutório do Core Inc. e nas próprias animações ao comentar “Isso não só para o meu entendimento, pra eu que enxergo me situar, mas para quem não enxerga e vai ouvir e vai conseguir também acompanhar.” Acreditando que a adição de sons pode ser útil até mesmo para alunos deficientes visuais. Enquanto o som para a melhor imersão do aluno com a animação é uma característica de melhoria mais fácil de se estabelecer, o suporte para usuários deficientes visuais exige um estudo mais aprofundado das características que devem ser atendidas para estas pessoas com deficiência, o que é um trabalho interessante no futuro, porém não é viável no desenvolvimento atual do Core Inc.

Por último, a professora C manifesta o desejo da adição de mais cenários e temáticas ao Core Inc. além da fábrica, ela exemplifica ao dizer que “(..) eu quisesse usar o software, num projeto de inclusão digital, com pessoas idosas ou sei lá, até mesmo adolescentes que não têm essa vivência de fábrica de contexto, seria interessante eu poder também ter outro contexto para fazer as animações.”, nisto ela comenta que contextos de colégio, parques de diversão e supermercados podem ser interessantes para atingir públicos alvos diferentes, indo além dos alunos de ensino superior, esta adição que a professora solicita se relaciona diretamente com a própria teoria da cognição situada que embasa este trabalho. Enquanto a construção do software

foi pensada para este tipo de adição de conteúdo por terceiros, ela não havia sido levantada até o momento como uma possível funcionalidade do software.

Enfim, quando indagado sobre o processo de teste do software e entrevista a professora C disse que “Não, para mim foi bem tranquilo. Eu gostei desse processo de interagir com o software primeiro.”, ela também comenta que a forma com que as etapas foram conduzidas o auxiliou a organizar suas opiniões para fornecê-las de maneira mais assertiva durante esta entrevista.

4.4.4 Análise Temática Entrevista Professor D

Tabela 4.4: Frases Entrevista Professor D

Modelo von Neumann	Eu acho que nas disciplinas que eu já lecionei, nesse nível de detalhe que o software está chegando, eu nunca usei, nunca cheguei a basear assim nenhuma explicação.
Modelo von Neumann	Acho que porque nas disciplinas de programação que eu já lecionei até hoje, era disciplinas um pouquinho mais para frente.
Modelo von Neumann	Agora que eu estou lecionando uma disciplina de arquitetura e de algoritmos e programação, que é um pouquinho mais básica, aí talvez viesse mais a calhar, mas ainda estou começando então eu não tenho a experiência para te dizer.
Algoritmos Animados e Visualização de software	Eu acho muito legal.
Algoritmos Animados e Visualização de software	Eu já usei algumas ferramentas de visualização na disciplina de orientação a objeto que elas não necessariamente animavam o código, mas ela tinha um componente visual que facilitava de você compreender os conceitos.
Algoritmos Animados e Visualização de software	A resposta dos alunos costuma ser muito positiva.
Conteúdos adaptados à cultura do aluno	Acho interessante, principalmente para usar nas disciplinas introdutórias quando nesses conceitos que a gente tenta passar pra eles ainda são muito abstratos.
Conteúdos adaptados à cultura do aluno	Acho que tendo uma coisa para eles verem, acontecendo e seguirem um passo a passo visualmente eu tendo a achar que vai funcionar.
Conteúdos adaptados à cultura do aluno	Vai ser uma ferramenta interessante para o aluno que vai ter mais facilidade de entender.
Impressões gerais Core Inc.	Como você me falou que era pra disciplina de algoritmos e programação e bem para o início, eu esperava que fosse mais simples.
Impressões gerais Core Inc.	Talvez eu esteja falando isso, porque eu estou considerando o meu público aqui, que costuma ter muita dificuldade com programação, principalmente no início.

Impressões gerais Core Inc.	Eu não tive a impressão que a forma como estava funcionando seria necessariamente para o aluno que está começando ou para um aluno que está fazendo a primeira disciplina de algoritmos de programação.
Impressões gerais Core Inc.	Eu acho que talvez fosse mais útil para para o aluno que está que está aprendendo arquitetura do que para o aluno que está aprendendo programação.
Impressões gerais Core Inc.	De buscar as instruções, fazer solicitações da memória e sair, não aconteceria aqui no primeiro semestre.
Tutorial e Usabilidade Core Inc.	Eu achei no início quando, quando rodou o primeiro teste eu fiquei um pouquinho confuso.
Tutorial e Usabilidade Core Inc.	Acho que é porque a tela não é tão grande.
Tutorial e Usabilidade Core Inc.	Quando aparece aquelas telinhas lado que está acontecendo agora com a instrução, que a unidade de controle está buscando agora, isso aqui é o que está sendo buscado na memória, aquelas paradinhas que ele dá para mostrar cada estágio, é legal.
Tutorial e Usabilidade Core Inc.	Quando chegou no segundo teste, "ahh tá é isso aqui que está acontecendo".
Tutorial e Usabilidade Core Inc.	Tem uma leve curva para você compreender, mas aqui também acho que é porque eu estava esperando uma coisa mais simples, então eu não estava esperando aquele nível de detalhamento.
Tutorial e Usabilidade Core Inc.	Não aguardei a informação de que você podia colocar o mouse em cima dos personagens do jogo e ver o que eles estão fazendo.
Tutorial e Usabilidade Core Inc.	Não é tão contra intuitivo, dá pra ele entender só vendo acontecer.
Temática e representação do Core Inc.	Acho que é bem claro, acho que é bem interessante.
Temática e representação do Core Inc.	Depois que você vê acontecendo um ou dois testes, você entende bem certinho quem é quem naquele cenário e qual é a função de cada de cada personagem.
Temática e representação do Core Inc.	Quando eu ia explicar mais ou menos esse funcionamento em alto nível de como é que esta as operações ocorrem dentro do computador o exemplo que eu dava era uma coisa parecida de coisas conversando entre si e fazendo solicitações uns aos outros.
Temática e representação do Core Inc.	Bonequinhos é bem mais interessante. Também com os bonequinhos, tem mais uma facilidade assim que você entender que é uma coisa que está fazendo uma ação.
Temática e representação do Core Inc.	Só formas geométricas, como sei lá um fluxograma, você ainda tem mais uma camada de abstração por cima, que o bonequinho deixa claro que é uma coisa dentro do computador.

Meio de abstração do Core Inc.	Eu achei um pouquinho confuso, já que como você coloca aqui que a sua intenção é que seja para alunos dos primeiros dos primeiros semestres.
Meio de abstração do Core Inc.	Porque você está presumindo aí que os elementos de entrada e saída, eles estão mapeados para a memória (...) isso aí eu achei que é um pouquinho confuso para o aluno que está vendo isso pela primeira vez.
Meio de abstração do Core Inc.	Eu estava pensando, tipo, quando você dá aula programando em Portugol, e aí tem lá leia e escreva.
Meio de abstração do Core Inc.	Principalmente para a disciplina de arquitetura está num ótimo nível para um aluno que está vai ver isso pela primeira vez.
Aplicação do Core Inc. em sala de aula.	E primeiro eu acho que assim para usar para apresentar para o aluno para ensinar seja lá o que for, eu acho que seria uma excelente como base pra mostrar para ele "Olha isso aqui vai acontecer e isso isso e isso", e depois eu destrinchar.
Aplicação do Core Inc. em sala de aula.	Se eu fosse usar ele do jeito que ele está agora eu faria desse jeito para mostrar a cara de cada coisa e explicar o que está acontecendo e por que ela faz aquilo, na ordem que ela faz aquilo, considerando as outras coisas que estão acontecendo.
Aplicação do Core Inc. em sala de aula.	Para eu organizar uma aula, por exemplo, para colocar o aluno para escrever um programa ali dentro do próprio software e observar o que está acontecendo eu acho que esperaria até ter algum controle da velocidade.
Modificações e correções Core Inc.	Uma coisa que eu achei que podia ajudar seria se você tivesse alguma indicação no código para dizer em qual linha está.
Modificações e correções Core Inc.	Eu acho que você podia deixar mais claro algumas coisas, tipo onde é que é o quê na memória? Porque é que às vezes a bonequinha da unidade de controle ela deixa o pacotinho do lado esquerdo ou do lado direito.
Modificações e correções Core Inc.	Eu acho que a tela podia ser um pouquinho maior para ver as animações e acho que você podia ter um controle de velocidade.
Modificações e correções Core Inc.	Se eu pudesse ver as animações mais devagar, mostrar para o aluno as animações mais devagar seria bastante benéfico.
Modificações e correções Core Inc.	Um controle de velocidade, ou então um controle de controlar o passo a passo, assim cada coisa e cada ação que está acontecendo, pelo que está acontecendo no sistema.
Modificações e correções Core Inc.	Ele já observa quando acontece com uma exceção? Uma divisão por zero. (...) Seria interessante, não é? Vai que dá várias luzes na fábrica e eles se enlouquecem.
Modificações e correções Core Inc.	Eu acho que no momento do tutorial, lá no início, se você colocar, por exemplo, um exemplozinho do que é o breakpoint?
Avaliação do processo de entrevista	Não, foi tranquilo.

O professor D atua na área de arquitetura de computadores e ministra aulas dentro do mesmo tópico junto de algoritmos e programação básica. Segundo ele o modelo de *von Neumann* se enquadra no seu trabalho e aulas, mas não da mesma forma que o Core Inc. aborda o tema, ele diz que “(...) nas disciplinas que eu já lecionei, nesse nível de detalhe que o software está chegando, eu nunca usei, nunca cheguei a basear assim nenhuma explicação.”, ele cita também o motivo pelo qual nunca utilizou o modelo neste nível de detalhe, segundo ele “(...) porque nas disciplinas de programação que eu já lecionei até hoje, era disciplinas um pouquinho mais para frente.”, ele cita que trabalhou com disciplinas de linguagem orientada a objeto, que não abordam mais esse tipo de conteúdo na ementa das disciplinas, contudo ele diz que agora está ministrando aulas de programação em estágios mais iniciais ao comentar que “Agora que eu estou lecionando uma disciplina de arquitetura e de algoritmos e programação, que é um pouquinho mais básica, aí talvez viesse mais a calhar, mas ainda estou começando então eu não tenho a experiência para te dizer.”, contudo ele prefere não dizer que utiliza muito os conceitos do modelo de *von Neumann* pois acabou e começou o trabalho nestas novas disciplinas.

Quando indagado sobre o tema de soluções algoritmos animados e visualização de software o professor D inicialmente se manifestou positivamente sobre o tema quando diz que “Eu acho muito legal.”, ele estende seu comentário citando sua experiência com este tipo de ferramenta “Eu já usei algumas ferramentas de visualização na disciplina de orientação a objeto que elas não necessariamente animavam o código, mas ela tinha um componente visual que facilitava de você compreender os conceitos”, assim é possível notar que o professor já teve contato com ferramentas de visualização de software, por mais que não tenham sido de algoritmos animados, ao mesmo tempo que parecem ser de caráter passivo pois não permitiam a animação de código em tempo real como o Core Inc., mesmo assim esta experiência anterior dá um embasamento melhor na opinião do professor, onde ele mesmo cita a recepção dos alunos as ferramentas que ele utilizou ao dizer que “A resposta dos alunos costuma ser muito positiva.”, isto demonstra que a opinião positiva do professor D não advém somente de opinião pessoal, mas também de experiência em sala de aula.

Um dos temas da entrevista também é a impressão dos professores sobre materiais e softwares adaptados à cultura dos alunos, quanto a isto o professor D também demonstra impressões positivas sobre a abordagem quando manifesta que “Acho interessante, principalmente para usar nas disciplinas introdutórias quando nesses conceitos que a gente tenta passar pra eles ainda são muito abstratos.”, citando justamente que a aproximação dos conteúdos ao aluno ajuda no processo de ensino de conteúdos abstratos como os abordados em computação, ele reforça essa impressão posteriormente ao afirmar sobre o próprio Core Inc. que “Vai ser uma ferramenta interessante para o aluno que vai ter mais facilidade de entender.”, comentando que esse tipo de ferramenta visual com conteúdo representado mais próximo ao cotidiano do aluno pode ter impactos positivos no aprendizado ao dizer que “Acho que tendo uma coisa para eles verem, acontecendo e seguirem um passo a passo visualmente eu tendo a achar que vai funcionar.”. Entende-se que a impressão que este tipo de software terá um impacto positivo no aprendizado dos alunos é reforçado pelo histórico que estas ferramentas possuem na carreira do professor D, ao ter citado anteriormente o uso de ferramentas visuais em sala de aula, mesmo que não fossem necessariamente adaptadas à cultura dos alunos.

Mesmo tendo citado o Core Inc. em momentos anteriores da entrevista, o professor D foi indagado sobre suas impressões gerais do Core Inc. enquanto entrava em contato com o software. Inicialmente o professor comenta que sentiu uma disparidade entre o que ele esperava do software das informações fornecidas inicialmente e o que ele encontrou no momento do teste, neste sentido ele comenta que “Como você me falou que era pra disciplina de algoritmos e programação e bem para o início, eu esperava que fosse mais simples.”. Neste momento a

entrevista teve um enfoque nessa aparente disparidade de conteúdos ministrados no curso que o professor D atua e o material que embasa o Core Inc.. Segundo ele na instituição em que atua os conteúdos ministrados são diferentes, ele explica que “Talvez eu esteja falando isso, porque eu estou considerando o meu público aqui, que costuma ter muita dificuldade com programação, principalmente no início.”. Por fim ele acrescenta também que em sua impressão o software não necessariamente seria bom para a introdução de algoritmos ao dizer “Eu não tive a impressão que a forma como estava funcionando seria necessariamente para o aluno que está começando ou para um aluno que está fazendo a primeira disciplina de algoritmos de programação.”. Nisto é reforçada a diferença de perspectivas de como o conteúdo é ministrado em dois cursos diferentes, já que os conteúdos abordados pelo Core Inc. são trabalhados nos materiais iniciais de algoritmos (Castilho et al., 2020) utilizados pelo curso de Ciência da Computação da UFPR, mas são vistos mais à frente no curso que o professor D atua, como ele comenta “De buscar as instruções, fazer solicitações da memória e sair, não aconteceria aqui no primeiro semestre.”.

Entretanto o professor explica que esses conteúdos são vistos na grade do curso em que trabalha e o Core Inc. poderia ser adequado a outras disciplinas, especificamente arquitetura de computadores como ele diz “Eu acho que talvez fosse mais útil para o aluno que está que está aprendendo arquitetura do que para o aluno que está aprendendo programação.”. Com isto, foi possível notar que as impressões iniciais do professor D focaram-se muito na adequação do software ao público-alvo pela disparidade entre o embasamento teórico diferente utilizado pelo Core Inc. e o das disciplinas ministradas pelo professor, ainda que nenhuma das duas abordagens esteja incorreta é possível notar que o Core Inc. pode ser utilizado em momentos diferentes do processo de ensino de algoritmos e programação dependendo do local em que será utilizado, pois mesmo que as grades curriculares dos cursos não sejam idênticas, o conteúdo abordado pelo software ainda é relevante.

Quanto a usabilidade do Core Inc. e seus tutoriais o professor D não foi contundente em suas afirmações como em outros temas, mas deu a entender em diversos momentos que há problemas. Em um primeiro momento ele comenta da dificuldade em compreender o software quando diz que “Eu achei no início quando, quando rodou o primeiro teste eu fiquei um pouquinho confuso.”, ele parcialmente tenta justificar essa confusão pelo tamanho da tela de animação do Core Inc. ao comentar “Acho que é porque a tela não é tão grande.”, com isso compreende-se que pode ser benéfico ao Core Inc. buscar dar mais destaque as suas animações com opções de telas maiores.

Ele comenta também que ao ver o software rodando as coisas se tornaram mais intuitivas: “Quando chegou no segundo teste, 'ahh tá é isso aqui que está acontecendo'.” e cita o momento da animação que mostra a coleta de novas instruções e variáveis no painel fictício da animação como uma boa decisão, pois deixou mais claro o que está acontecendo em cada situação “Quando aparece aquelas telinhas lado que está acontecendo agora com a instrução, que a unidade de controle está buscando agora, isso aqui é o que está sendo buscado na memória, aquelas paradinhas que ele dá para mostrar cada estágio, é legal.”. Além disto, o professor D comenta outro problema a respeito do tutorial quando fala “Não aguardei a informação de que você podia colocar o mouse em cima dos personagens do jogo e ver o que eles estão fazendo.”, aqui é relevante salientar que esta informação é transmitida pelo tutorial em um de seus passos, contudo o professor se esqueceu, demonstrando que pode ser necessário um destaque maior do tutorial nesta etapa.

Dado estas críticas o professor D concluí que “Tem uma leve curva para você compreender, mas aqui também acho que é porque eu estava esperando uma coisa mais simples, então eu não estava esperando aquele nível de detalhamento.” e “Não é tão contra intuitivo, dá pra ele entender só vendo acontecer.”, com isto ele explica que em um primeiro momento a tela pode do

Core Inc. é confusa, mas após um curto intervalo de tempo utilizando a ferramenta as coisas ficam mais claras e a compreensão é atingida através dos elementos visuais presentes, dado isto é importante em melhorias futuras aprimorar a usabilidade dos primeiros momentos de uso do Core Inc., com tutoriais mais claros nos pontos citados pelo professor D de forma a diminuir o atrito e estranhamento inicial com o software.

O Core Inc. busca abstrair o conteúdo através de uma temática e representação familiares a cultura e cotidiano dos alunos, quando indagado sobre o quão bem essa representação e temática é feita o professor D disse que “Acho que é bem claro, acho que é bem interessante.”, ele explica novamente que a representação fica mais clara com o passar do tempo “Depois que você vê acontecendo um ou dois testes, você entende bem certinho quem é quem naquele cenário e qual é a função de cada de cada personagem.” Demonstrando que a representação da temática e visual por si só atende bem seu objetivo segundo a perspectiva do professor D.

Dentro de sala de aula o professor D traça paralelos com o Core Inc. também, ao citar que faz uso deste tipo de abstração contextualizada com a realidade para explicar conteúdos como, como ele diz “Quando eu ia explicar mais ou menos esse funcionamento em alto nível de como é que esta as operações ocorrem dentro do computador o exemplo que eu dava era uma coisa parecida de coisas conversando entre si e fazendo solicitações uns aos outros.”. Quando indagado sobre se o professor acha que a temática e a representação do Core Inc. poderiam fazer diferença em relação a outras soluções deste tipo não contextualizada com a vivência e cultura do aluno o professor disse que “Bonequinhos é bem mais interessante. Também com os bonequinhos, tem mais uma facilidade assim que você entender que é uma coisa que está fazendo uma ação.” e “Só formas geométricas, como sei lá um fluxograma, você ainda tem mais uma camada de abstração por cima, que o bonequinho deixa claro que é uma coisa dentro do computador.”, explicando que formas não contextualizadas quando apresentadas para os alunos geram uma dificuldade maior para abstrair os conceitos em relação a proposta do Core Inc. e que por si só os personagens do Core Inc. tornam as coisas mais interessantes para o aprendizado.

Quanto aos métodos utilizados para abstração do modelo de *von Neumann* e seus passos para animações o professor D considerou que alguns conceitos ficaram confusos, ele diz que “Eu achei um pouquinho confuso, já que como você coloca aqui que a sua intenção é que seja para alunos dos primeiros dos primeiros semestres.”, ele explica que o nível de detalhe aplicado pode acabar confundindo os alunos, já que neste momento do aprendizado e algoritmos os alunos estariam vendo uma versão mais simplificada do processo, ele exemplifica isto com as aulas de Portugol que realiza ao dizer que “Eu estava pensando, tipo, quando você dá aula programando em Portugol, e aí tem lá leia e escreva.”, especificamente ele comenta como a demonstração da entrada e saída de dados com leia e escreva é mais complexa na animação do que o conteúdo que ele passa em sala de aula “Porque você está presumindo aí que os elementos de entrada e saída, eles estão mapeados para a memória (...) isso aí eu achei que é um pouquinho confuso para o aluno que está vendo isso pela primeira vez.”. Dado estes problemas de nível da abstração não estar de acordo com o nível de conhecimento dos alunos no público alvo pretendido, o professor D considera que para outro momento do curso o Core Inc. se apresenta como uma opção muito melhor, como ele finaliza “Principalmente para a disciplina de arquitetura está num ótimo nível para um aluno que está vai ver isso pela primeira vez.”.

O professor também comentou sobre como aplicaria o Core Inc. em sala de aula, inicialmente ele comenta que utilizaria o software para expor conteúdo aos alunos, sem a ação direta dos mesmos no software, ele explica “E primeiro eu acho que assim para usar para apresentar para o aluno para ensinar seja lá o que for, eu acho que seria uma excelente como base pra mostrar para ele ‘Olha isso aqui vai acontecer e isso isso e isso’, e depois eu destrinchar.”, ele utilizaria o software desta forma segundo ele para demonstrar como é o funcionamento de cada

parte da arquitetura de *von Neumann*, ele diz que “Se eu fosse usar ele do jeito que ele está agora eu faria desse jeito para mostrar a cara de cada coisa e explicar o que está acontecendo e por que ela faz aquilo, na ordem que ela faz aquilo, considerando as outras coisas que estão acontecendo.”, para a utilização do software pelos alunos o professor D não considera que o software esteja em um estado adequado ainda, segundo ele uma funcionalidade de controle de velocidade da animação seria essencial “Para eu organizar uma aula, por exemplo, para colocar o aluno para escrever um programa ali dentro do próprio software e observar o que está acontecendo eu acho que esperaria até ter algum controle da velocidade.”

Por fim, o professor elencou a pedidos do entrevistador modificações e correções que ele considera necessárias ou benéficas ao software. Ele repete alguns pontos citados anteriormente ao citar o tamanho da tela e o controle de velocidade neste momento “Eu acho que a tela podia ser um pouquinho maior para ver as animações e acho que você podia ter um controle de velocidade.”. Ele também considera mudanças benéficas no campo de digitação do código, com sinalizadores melhores de funcionamento dos pontos de parada (breakpoints) “Eu acho que no momento do tutorial, lá no início, se você colocar, por exemplo, um exemplozinho do que é o breakpoint?” e uma indicação de qual linha está sendo animada no momento “Uma coisa que eu achei que podia ajudar seria se você tivesse alguma indicação no código para dizer em qual linha está.”, tais modificações poderiam facilitar segundo o professor a compreensão da forma que funciona a execução da animação.

Além disto, o professor cita que poderia ser melhor contextualizado algumas ações dos personagens que apesar de possuírem um significado não são explicadas explicitamente, ele comenta alguns exemplos como “Eu acho que você podia deixar mais claro algumas coisas, tipo onde é que é o quê na memória? Porque é que às vezes a bonequinha da unidade de controle ela deixa o pacotinho do lado esquerdo ou do lado direito.”, por último ele fala que os erros serem tratados pela animação também seria uma boa adição de funcionalidade, quando ele diz “Ele já observa quando acontece com uma exceção? Uma divisão por zero. (...) Seria interessante, não é? Vai que dá várias luzes na fábrica e eles se enlouquecem.” animações específicas é o que o professor D solicita, demonstrando como funciona o tratamento de erros na máquina também, onde atualmente a animação nem é executada.

No fim da entrevista o professor foi inquirido sobre o processo de entrevista, se ele esteve confortável com o processo ou tinha propostas de melhoria e ele respondeu “Não, foi tranquilo.”.

4.5 RESULTADOS EXPERIMENTO COMPLEMENTAR

O experimento complementar realizado consistiu em um questionário com uma série de perguntas, este dividido em duas partes. A primeira parte busca analisar a experiência dos professores participantes do experimento principal com softwares de visualização e algoritmos animados e a segunda parte consiste em perguntas que se adequam ao Modelo de Aceitação de Tecnologia ou método TAM (Technology Acceptance Model) demonstrado na seção 2.8 desta dissertação.

Abaixo são apresentados os gráficos gerados com as respostas dos quatro professores participantes e uma análise destes dados.

A primeira pergunta realizada diz respeito a experiência dos professores com o tipo de ferramenta que o Core Inc. é. Os resultados na figura 4.2 demonstram que todos os participantes já haviam tido experiência com o uso de ferramentas de visualização de software enquanto professores ou pesquisadores, isso dá mais credibilidade as entrevistas realizadas no experimento

principal pois demonstra que os participantes do experimento possuem vivência com este tipo de ferramenta.

Você já utilizou ferramentas de visualização de software, ou seja, ferramentas que utilizem de abordagens visuais para auxiliar no ensino em sala de aula?

4 respostas

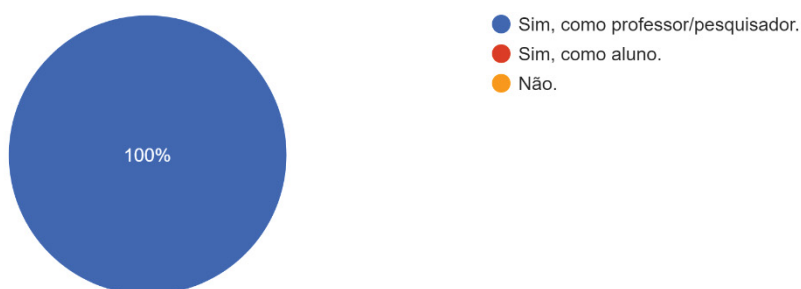


Figura 4.2: Uso de ferramentas de visualização de software

Quando indagados especificamente sobre o histórico de uso de ferramentas de algoritmos animados demonstra-se na figura 4.3 que um dos professores apesar de possuir experiência com ferramentas de visualização, nunca utilizou especificamente softwares de algoritmos animados como o Core Inc. Ainda assim a maior parte dos professores já possuía uma base do que esperar do software.

Você já utilizou especificamente ferramentas de algoritmos animados, ou seja, que abstraíssem o código fonte desenvolvido em animações, gráficos ou formas geométricas?

4 respostas

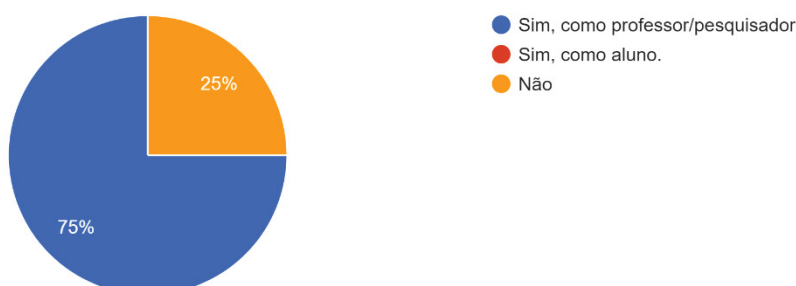


Figura 4.3: Uso de ferramentas de algoritmos animados

Sabendo disto, a pergunta seguinte confirma que as experiências anteriores dos professores não causaram um efeito negativo em sua opinião geral sobre soluções de algoritmos animados, pois segundo a figura 4.4 todos os professores ainda possuem interesse em conhecer e utilizar novas ferramentas deste tipo. Esta informação é importante para validar que nenhuma das entrevistas possui um viés de rejeição imediato mesmo antes do Core Inc. ser apresentado aos professores.

A frequência com que os professores utilizam este tipo de ferramenta, no entanto é relativamente baixa, pois ao solicitar aos mesmos que classificassem a frequência com que

Você possui interesse em utilizar soluções de algoritmos animados?

4 respostas

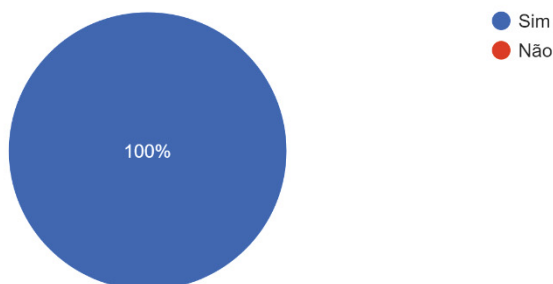


Figura 4.4: Interesse em algoritmos animados

utilizam essa ferramenta de 1 à 5, sendo 1 (um) Pouco Frequente e 5 (cinco) Muito Frequente, a figura 4.5 apresenta que 3 dos 4 professores ficaram entre os valores 1 e 2, demonstrando que apesar do interesse neste tipo de ferramenta o seu uso em sala de aula e pesquisa ainda é pouco frequente entre os mesmos.

Com que frequência você utiliza ou utilizou soluções de algoritmos animados e similares?

4 respostas

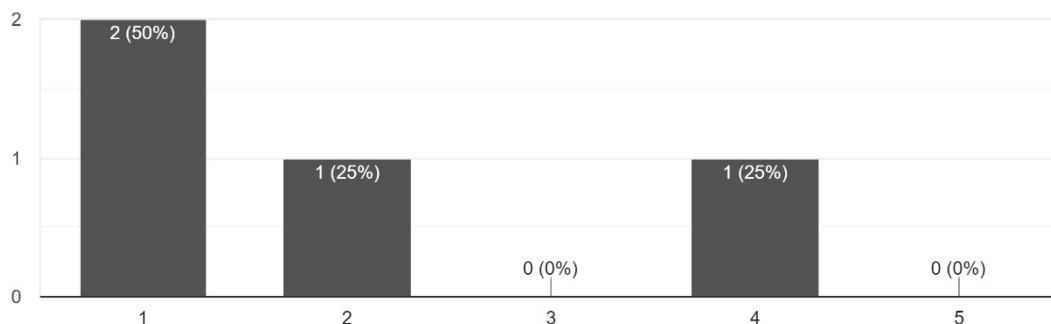


Figura 4.5: Frequência de uso algoritmos animados e visualização de software

Buscando mais informações sobre a intensidade do uso de softwares similares ao Core Inc. dentre os pesquisadores, a pergunta seguinte diz respeito a quantos softwares similares eles já haviam utilizado e segundo a figura 4.6 50% dos participantes utilizou 2 (dois) softwares enquanto um deles utilizou 3 (três) e por fim um dos professores só teve uma experiência similar ao Core Inc. Isso demonstra que apesar da experiência dos professores com este tipo de ferramenta, ela é limitada dado a quantidade pequena de softwares similares ao Core Inc. que eles utilizaram até hoje.

Quanto aos temas abordados pelas soluções utilizadas pelos professores anteriormente, não há uma hegemonia nos dados segundo o gráfico da figura 4.7, pois com exceção de três áreas de estudo os softwares abrangeram conteúdos diferenciados, indo de estudos com orientação a objetos, grafos, arquitetura de computadores até conceitos mais fundamentais de programação e estrutura de dados. Contudo ferramentas que abordam introdução a algoritmos e estrutura de

Quantos softwares de algoritmos animados e similares você já utilizou?

4 respostas

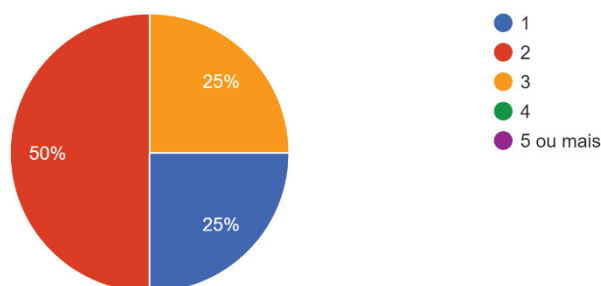


Figura 4.6: Número de softwares utilizados

dados são um pouco mais relevantes na experiência dos professores, o que dá mais experiência aos mesmos ao avaliar o Core Inc. que também pertence a área de introdução a algoritmos.

As soluções que você utilizou abordavam quais dos temas abaixo:

4 respostas

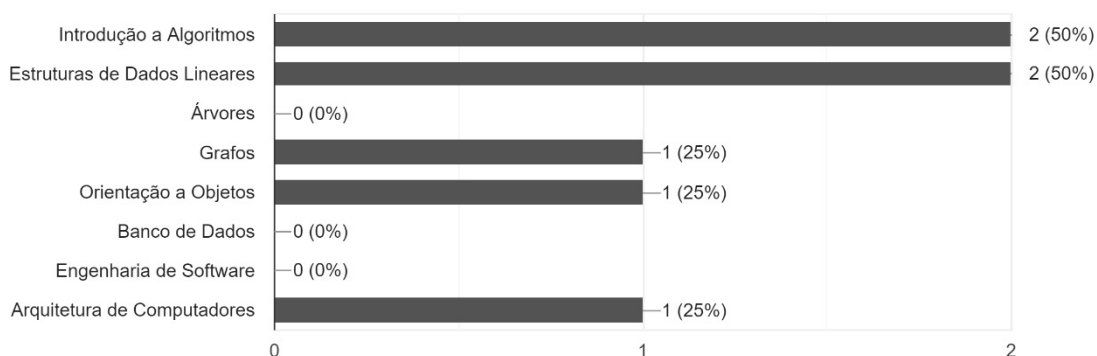


Figura 4.7: Temas das soluções utilizadas

A pergunta sobre os motivos de nunca ter utilizado ferramentas de visualização de software ou algoritmos animados foi desconsiderada, pois não há dados relevantes já que todos os professores entrevistados possuem alguma experiência com estas ferramentas durante sua pesquisa e trabalho.

A última pergunta desta sessão sobre as experiências anteriores dos professores foi sobre sua satisfação geral com este tipo de ferramenta. Onde 1 significa Pouco Satisfatório e 5 Muito Satisfatório. A figura 4.8 demonstra que 75% dos professores têm uma opinião neutra sobre os softwares e um deles graduou a questão com o valor 4 (quatro), indicando uma experiência mais positiva com este tipo de ferramenta. Estes dados demonstram que os professores de forma geral não são previamente contra soluções de algoritmos animados, porém em sua maioria também não tem experiências anteriores que os façam inicialmente possuir opiniões extremamente positivas somente pelo fato de o Core Inc. ser uma ferramenta de algoritmos animados, isto demonstra que as opiniões fornecidas nas entrevistas não aparentam ter um viés de confirmação prévio positivo ou negativo somente pelas características básicas do Core Inc.

De maneira geral, o quão satisfatória você considera que sua experiência com estas soluções foi?

4 respostas

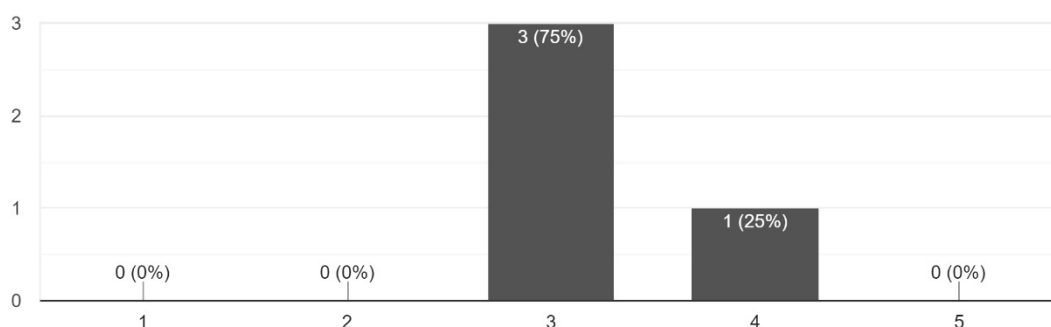


Figura 4.8: Satisfação das experiências anteriores

As perguntas deste experimento complementar apresentadas até então dizem respeito ao histórico de experiências dos professores sobre o tipo de ferramenta relacionada ao Core Inc. e suas opiniões prévias sobre o tema. A partir de agora o questionário aborda especificamente a experiência dos professores ao utilizar o Core Inc. em seu período de testes, onde todas as perguntas são fundamentadas segundo o Modelo de Aceitação de Tecnologia.

O Modelo de Aceitação de Tecnologia propõe a apresentação de afirmações aos participantes e utilizando a Escala Likert os mesmos classificam o quanto eles concordam com cada uma das afirmações com valores que vão de 1 a 5, onde 1 significa Discordo Totalmente e 5 Concordo Totalmente.

A primeira afirmação apresentada para os participantes é “Em geral, usar o Core Inc. é fácil.” E segundo os dados da figura 4.9 os professores concordam que a utilização do Core Inc. é fácil.

Em geral, usar o Core Inc. é fácil.

4 respostas

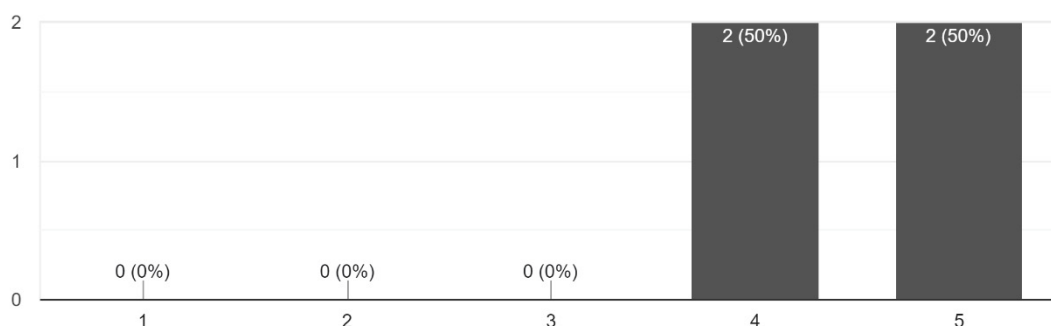


Figura 4.9: Respostas “Em geral, é fácil usar o Core Inc.”

A segunda afirmação foi “É fácil lembrar como realizar ações no Core Inc.” e a maior parte dos professores considerou que sim, é fácil lembrar das ações possíveis de realizar no Core

Inc segundo a figura 4.10. Uma das respostas discordou desta afirmação, o que condiz com o fato de que alguns professores confundiram alguns elementos do Core Inc. durante a entrevista. Os dados apresentados reiteram análises realizadas anteriormente no experimento principal em que é necessária uma atenção maior em alguns elementos do tutorial.

É fácil lembrar como realizar ações no Core Inc.

4 respostas

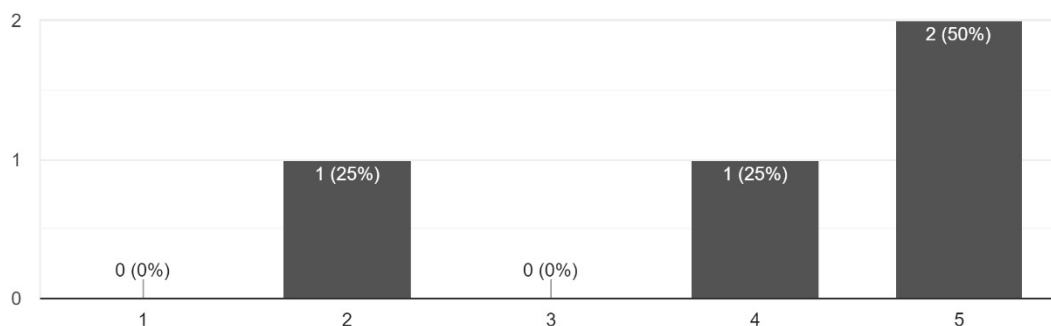


Figura 4.10: Respostas “É fácil lembrar como realizar ações no Core Inc.”

Quanto a afirmação “Não é necessário treinamento para utilizar o Core Inc.” os professores divergiram muito em suas opiniões, na figura 4.11 cada um classificou seu grau de adequação com a frase de maneira diferente. Apesar de nenhum dos mesmos discordarem completamente desta frase, a distribuição dos dados reflete nas diferentes impressões que os professores tiveram da interface do Core Inc., onde alguns viram problemas de usabilidade que precisam ser corrigidos e outros consideraram a tela e suas funções tão simples de entender que o tutorial nem seria necessário. Entende-se aqui que enquanto alguns professores consideraram a tela muito intuitiva, a necessidade do tutorial ainda se faz essencial para guiar o usuário pelas funcionalidades do Core Inc.

Não é necessário treinamento para utilizar o Core Inc.

4 respostas

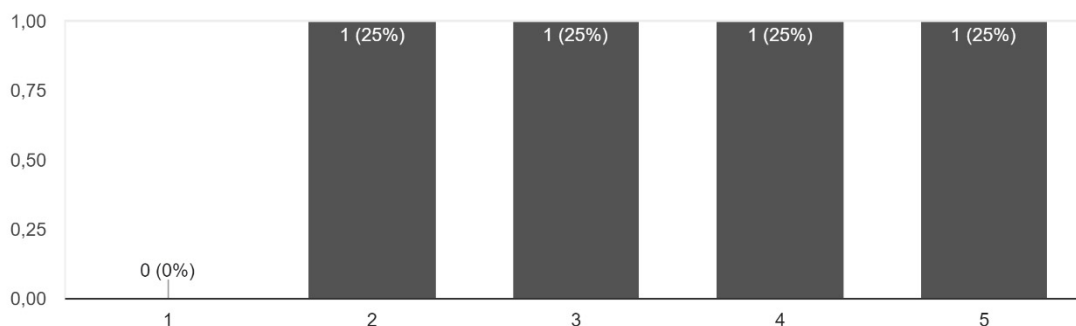


Figura 4.11: Respostas “Não é necessário treinamento para utilizar o Core Inc.”

Nesta afirmação busca-se entender se “Aprender a usar o Core Inc. é fácil.” e todos os professores segundo a figura 4.12 concordam que da forma com que o Core Inc. funciona hoje aprender a utilizá-lo é fácil, isso demonstra-se nas entrevistas, onde apesar da solicitação de correções pontuais nos tutoriais e tela os professores consideraram o funcionamento do Core Inc. bem explicado.

Aprender a usar o Core Inc. é fácil.

4 respostas

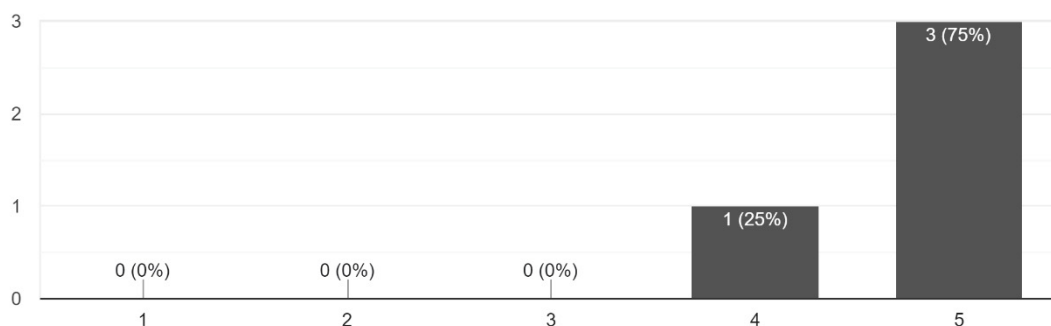


Figura 4.12: Respostas “Aprender a usar o Core Inc. é fácil.”

Apesar dos resultados das perguntas anteriores demonstrarem-se majoritariamente positivos quanto a facilidade de uso do Core Inc, quando encarados com a afirmação “Utilizar o Core Inc. é satisfatório.” na figura 4.13 um dos professores discordou totalmente desta afirmação. Neste caso o professor A é quem discorda desta afirmação completamente e analisa-se que segundo sua entrevista os motivadores desta divergência não são sobre a usabilidade e tutoriais do software e sim sobre como a representação de diversos conceitos está incorreta segundo a visão do professor, assim como a própria abordagem de abstração de conceitos que o Core Inc. propõe.

Utilizar o Core Inc. é satisfatório.

4 respostas

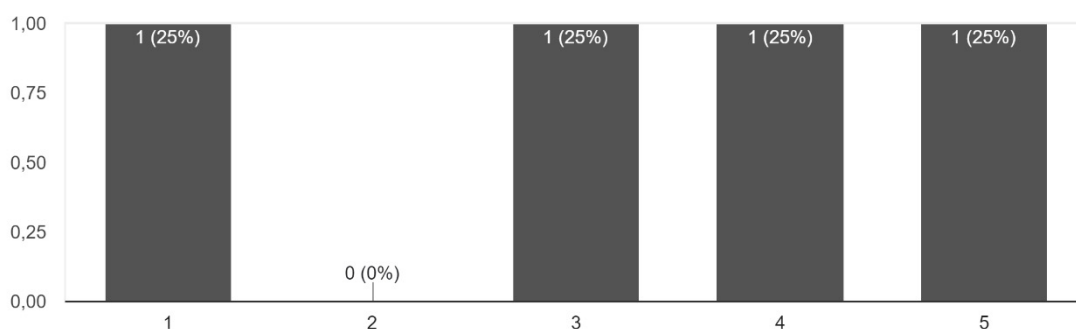


Figura 4.13: Respostas “Utilizar o Core Inc. é satisfatório.”

Com o objetivo de entender se o tutorial está fazendo realmente o suficiente ao explicar o software a afirmação “Consigo utilizar o Core Inc. sem o auxílio de um instrutor.” procura descobrir se um instrutor é necessário para ao uso do Core Inc., e segundo a figura 4.14 os professores parecem em geral concordar que não é necessário um instrutor, ainda que haja espaço para melhora nos tutoriais para que esta necessidade seja completamente extinta.

Consigo utilizar o Core Inc. sem o auxílio de um instrutor.

4 respostas

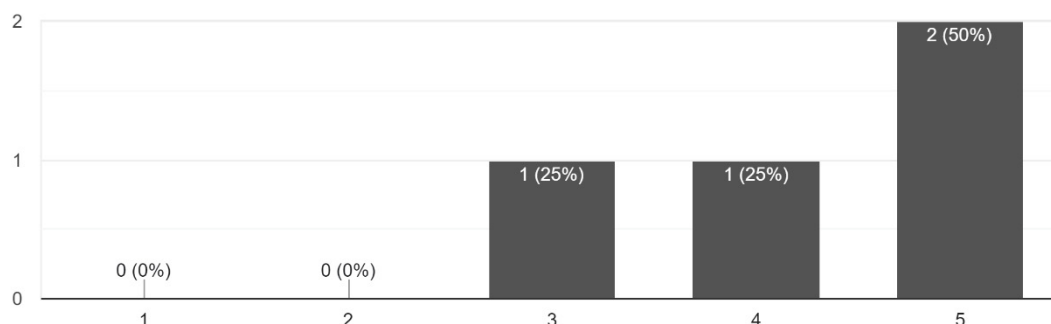


Figura 4.14: Respostas “Consigo utilizar o Core Inc. sem o auxílio de um instrutor.”

Na última das afirmações do modelo de aceitação de tecnologia sobre a usabilidade do software indaga-se se “A interface do Core Inc. é de fácil entendimento.”. Segundo à figura 4.15 metade dos professores não concordam nem discordam desta afirmação e os outros dois tem opiniões divergentes, em que um concorda com a afirmação e o último professor discorda. Entende-se aqui que a interface do Core Inc. por si só não é trivial pois o módulo de animação na tela não é comum de ser observado em outros softwares do dia a dia de alunos e professores, isso relaciona-se diretamente a repetidas impressões iniciais que os professores tiveram de confusão com o funcionamento do software ao declararem suas primeiras impressões na entrevista. Contudo os tutoriais demonstram de muito valor ao processo de explicação da interface do software, pois garantem uma opinião mais positiva sobre a facilidade de uso do software, como observado durante as entrevistas e na figura 4.12.

A sequência de afirmações a seguir diz respeito ao uso do Core Inc. em sala de aula e o quão benéfico ele seria aplicado aos alunos segundo os professores. Assim como a sessão de afirmações anterior estes dados servem para apoiar perguntas já feitas aos professores sobre este tema no experimento principal.

Quanto a utilização do Core Inc. em sala de aula a primeira afirmação apresentada foi “Utilizar o Core Inc. pode melhorar o processo de ensino durante a aula.”, onde na figura 4.16 as opiniões foram muito divergentes, com dois professores concordando com a afirmação dois discordando da mesma. Os dois professores que discordam desta afirmação são os professores A e B que como demonstrado na análise de suas respectivas entrevistas possuem problemas com a forma que o Core Inc. abstraiu conceitos da arquitetura de *von Neumann*, além disso o professor A que discorda totalmente desta afirmação também disse acreditar que o meio de abstração com personagens e características visuais de jogos não é adequado para o ensino superior. Já a professora C e o professor D são os que aqui concordam mais com esta afirmação, como demonstrado também em suas entrevistas os mesmos haviam dado opiniões favoráveis ao uso do software.

A interface do Core Inc. é de fácil entendimento.

4 respostas

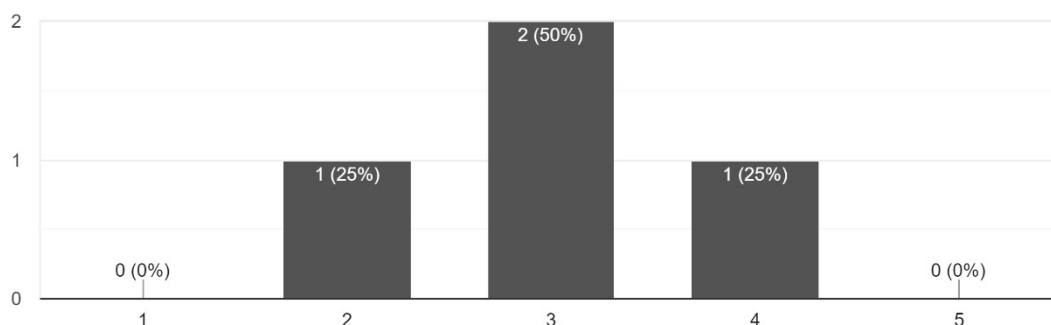


Figura 4.15: Respostas “A interface do Core Inc. é de fácil entendimento.”

Utilizar o Core Inc. pode melhorar o processo de ensino durante a aula.

4 respostas

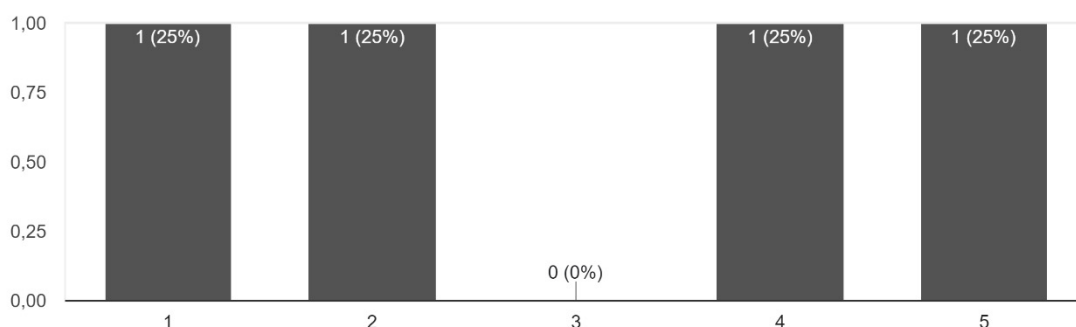


Figura 4.16: Respostas “Utilizar o Core Inc. pode melhorar o processo de ensino durante a aula.”

Prosseguindo sobre se o uso do Core Inc. especificamente em sala de aula melhoraria as aulas duas afirmações foram propostas, sendo elas: “Usar o Core Inc. auxiliaria no ensino de algoritmos e programação em sala de aula.” e “O Core Inc. auxiliaria durante as aulas.”. Em ambas as afirmações das figuras 4.17 e 4.18 os professores tiveram dificuldades em concordar completamente, os professores A e B confirmam mais uma vez aqui suas críticas já feitas anteriormente. O destaque fica com o professor D que acredita que pode ser benéfico em sala de aula, porém não exatamente no ensino de algoritmos, ele compartilha sua visão com o professor B que ao invés de manter a graduação 1 (um) atribuída a primeira afirmação, atribuiu o valor 2 (dois) a segunda afirmação, isto ocorre pois ambos os professores acreditam que o Core Inc. poderia ser melhor utilizado em aulas de arquitetura de software e atribuíram notas melhores a segunda afirmação que não é específica ao ensino de algoritmos.

Sobre a capacidade do Core Inc. de facilitar o trabalho do professor em sala de aula as afirmações “Consigo atingir meus objetivos como professor mais facilmente com o auxílio do Core Inc.” e “Utilizar o Core Inc. pode melhorar a qualidade de meu trabalho.” foram colocadas. As figuras 4.19 e 4.20 demonstram que o professor A mantém sua posição que o Core Inc.

Usar o Core Inc. auxiliaria no ensino de algoritmos e programação em sala de aula.

4 respostas

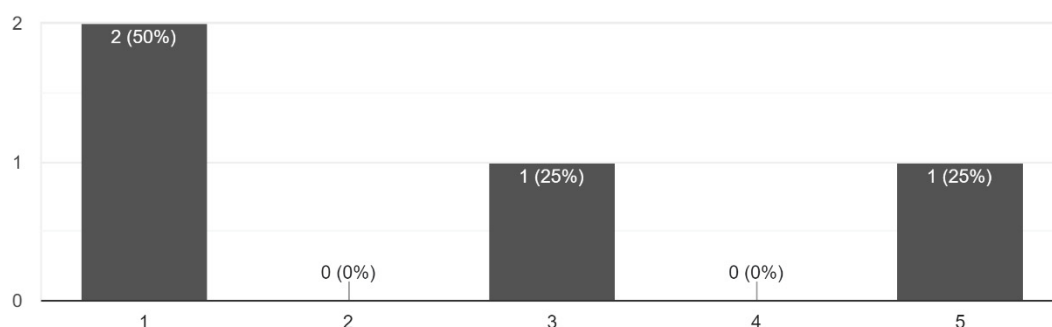


Figura 4.17: Respostas “Usar o Core Inc. auxiliaria no ensino de algoritmos e programação em sala de aula.”

O Core Inc. auxiliaria durante as aulas.

4 respostas

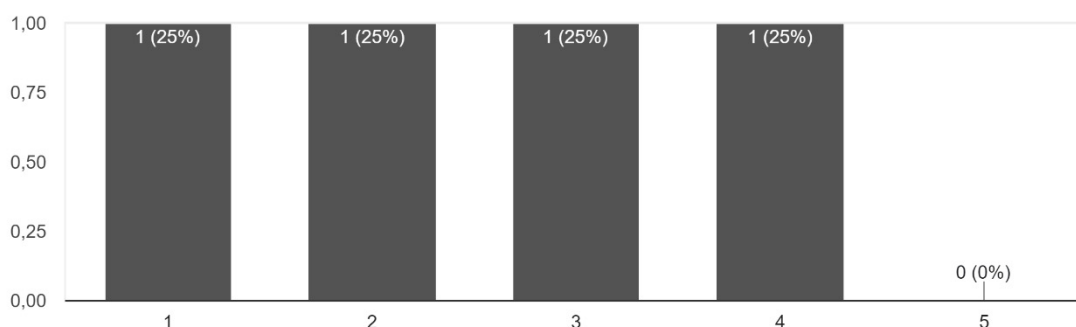


Figura 4.18: Respostas “O Core Inc. auxiliaria durante as aulas.”

não ajudaria em sala de aula no seu trabalho, os demais professores tentem a dúvida ou leve concordância em relação a estas afirmações, condizendo com as opiniões do professor B durante o experimento principal, em que comenta sobre os diversos problemas do Core Inc. ao mesmo tempo que vê potencial de utilização para a ferramenta, ainda mais se realizadas as correções que ele considera necessárias.

Ao relacionar o Core Inc. e produtividade em sala de aula com a afirmação “Utilizar o Core Inc. pode tornar as aulas mais produtivas.” Os dados visualizados na figura 4.21 demonstram que os professores C e D, estes que mais fizeram análises positivas durante sua entrevista do uso do software em sala de aula são os que mais concordam com esta afirmação. Avalia-se que ambos os professores falam em suas entrevistas que gostam de utilizar analogias similares em sala de aula e este trabalho é feito manualmente no quadro, neste sentido eles acreditam que o Core Inc. pode acelerar este processo ao apresentar uma forma de abstração pronta e que se adeque as suas aulas atuais.

A última das afirmações desta sessão sobre o uso em sala de aula também diz respeito a otimização do uso do tempo em sala de aula, a afirmação “Utilizar o Core Inc. pode melhorar a

Consigo atingir meus objetivos como professor mais facilmente com o auxílio do Core Inc.

4 respostas

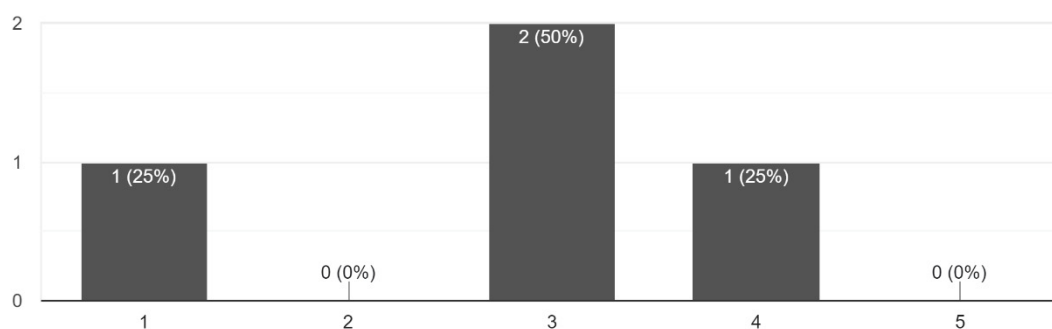


Figura 4.19: Respostas “Consigo atingir meus objetivos como professor mais facilmente com o auxílio do Core Inc.”

Utilizar o Core Inc. pode melhorar a qualidade de meu trabalho.

4 respostas

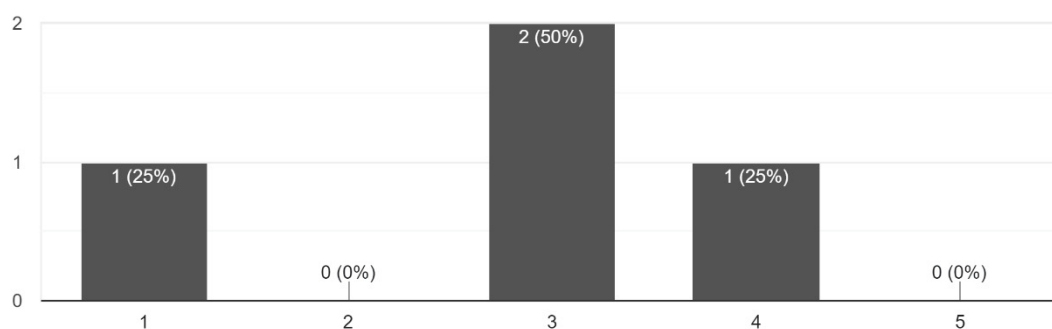


Figura 4.20: Respostas “Utilizar o Core Inc. pode melhorar a qualidade de meu trabalho.”

eficiência do tempo gasto em sala de aula.” trouxe respostas neutras ou negativas em geral, isto se dá principalmente pelas opiniões dos professores A e B que consideram que o Core Inc. no estado atual apresenta conceitos de maneira incorreta e podem gerar um retrabalho por parte do professor ao ter de corrigir os conceitos apresentados pelo Core Inc. ao invés de utilizar o software para demonstrá-los de maneira mais simples.

A seção de afirmações a seguir diz respeito a variáveis adicionais que podem influenciar no uso do software e são específicas das características do Core Inc., como suas animações, história e personagens.

A primeira afirmação desta seção diz respeito aos visuais do Core Inc. em geral, tanto em sua arte de animações quanto na sua tela. Portanto, quando confrontados com “O Core Inc. possui um visual engajante.” Os professores em sua maioria consideram os visuais do Core Inc. chamativos e engajantes para o propósito da ferramenta. O professor A no entanto, discorda desta afirmação pois ele não considera a temática e a apresentação visual do Core Inc. adequada ao ensino superior.

Utilizar o Core Inc. pode tornar as aulas mais produtivas.

4 respostas

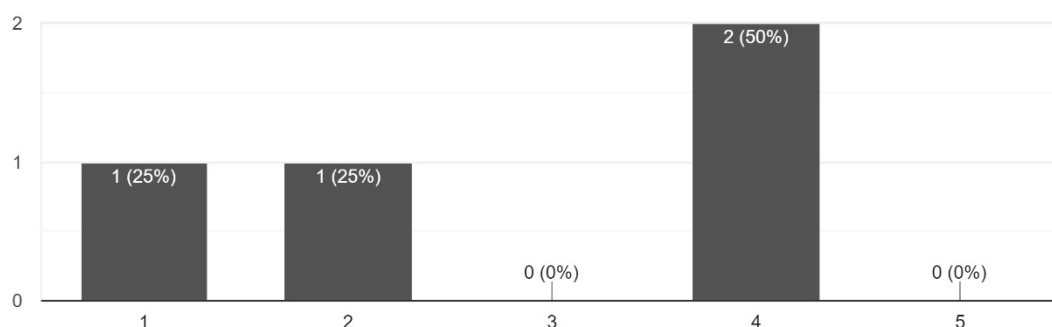


Figura 4.21: Respostas “Utilizar o Core Inc. pode tornar as aulas mais produtivas.”

Utilizar o Core Inc. pode melhorar a eficiência do tempo gasto em sala de aula.

4 respostas

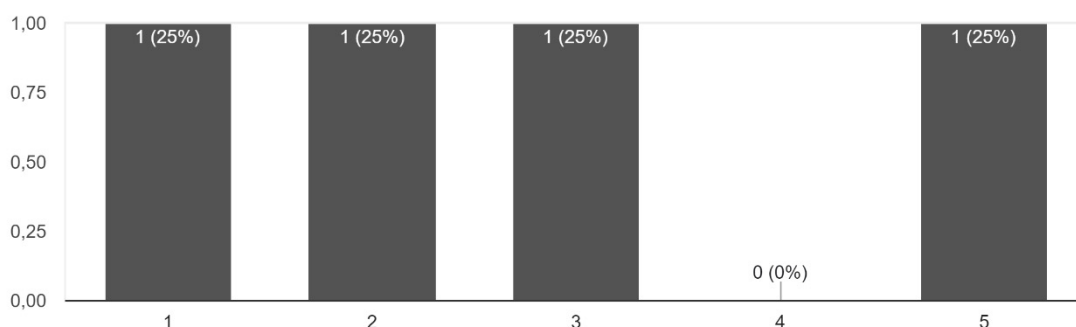


Figura 4.22: Respostas “Utilizar o Core Inc. pode melhorar a eficiência do tempo gasto em sala de aula.”

Em relação aos personagens do Core Inc. foi feita afirmação “Os personagens do Core Inc. são bonitos.” e os professores responderam positivamente a esta frase como exibido na figura 4.24, correspondendo as opiniões emitidas em geral durante as entrevistas.

Uma afirmação similar foi feita também sobre o cenário das animações sendo ela “O cenário das animações de Core Inc. é bonito.”. Com exceção do professor A os demais professores também responderam positivamente aos visuais do cenário do Core Inc. de acordo com a figura 4.25, espelhando a recepção destes professores sobre o assunto durante o experimento principal.

Ainda sobre o cenário das animações a frase “O cenário das animações de Core Inc. é de fácil entendimento.” os professores se demonstraram divididos na figura 4.26. Isto se refletiu nas entrevistas onde alguns professores comentam como é rápido o entendimento de cada personagem e suas ações enquanto outros consideram que uma melhor contextualização narrativa e melhores tutoriais nesta parte são necessários.

Ao abordar características das animações do Core Inc. duas afirmações foram apresentadas no questionário: “As animações do Core Inc. são engajante.” e “As animações do Core Inc. são de fácil entendimento.”. Enquanto na figura 4.27 os professores em sua maioria

O Core Inc. possui um visual engajante.

4 respostas

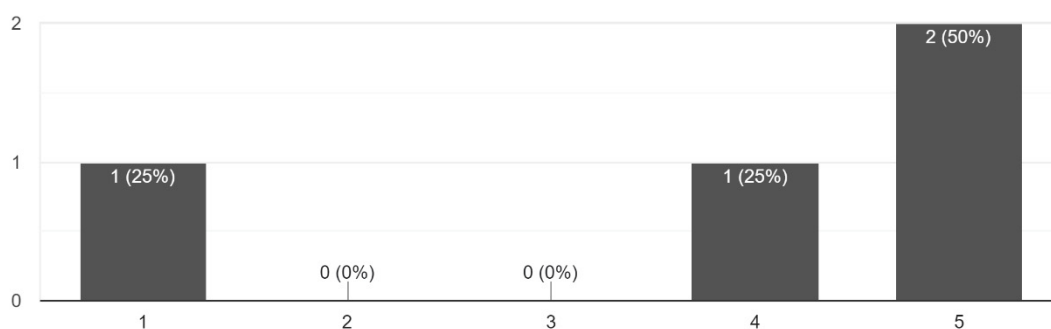


Figura 4.23: Respostas “O Core Inc. possui um visual engajante.”

Os personagens do Core Inc. são bonitos.

4 respostas

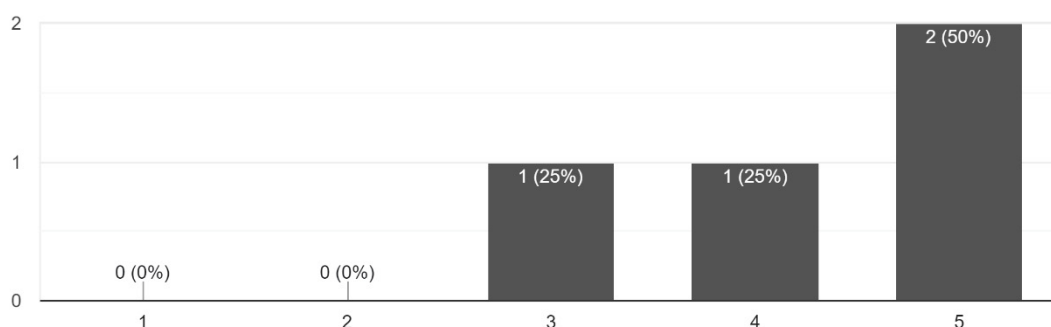


Figura 4.24: Respostas “Os personagens do Core Inc. são bonitos.”

concordam que as animações são engajantes o cenário fica incerto na figura 4.28 quando se trata da compreensão das animações em si com dois professores discordando da afirmação parcialmente, o que reflete nos mesmos pontos tocados pela afirmação anterior da figura 4.26, onde a construção visual das animações é bem feita, mas elas podem ser confusas de se compreender, principalmente em um primeiro momento.

As duas últimas afirmações desta seção de perguntas sobre variáveis adicionais específicas ao Core Inc. dizem respeito aos tutoriais e são: “O tutorial do Core Inc. é de fácil acesso.” e “As instruções do tutorial são de fácil compreensão.”. Na figura 4.29 a maioria dos professores considera de fácil acesso o tutorial, que aparece logo ao entrar no software e possui um botão de tutorial na tela. O professor A apesar de não levantar problemas com o acesso ao tutorial durante sua entrevista discordou parcialmente desta afirmação. A segunda frase diz respeito ao quão fácil é compreender o tutorial e os professores foram majoritariamente positivos na figura 4.30, onde nenhum deles chega a discordar desta afirmação. Entretanto, durante a entrevista os professores levantaram diferentes pontos de melhoria no tutorial que poderiam esclarecer melhor

O cenário das animações de Core Inc. é bonito.

4 respostas

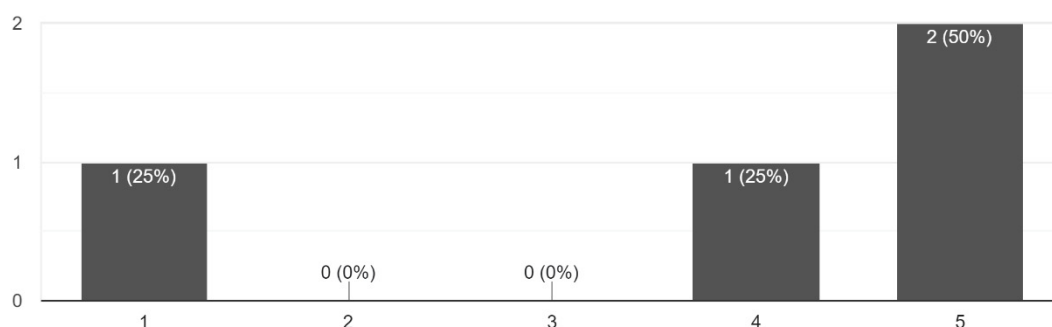


Figura 4.25: Respostas “O cenário das animações de Core Inc. é bonito.”

O cenário das animações de Core Inc. é de fácil entendimento.

4 respostas

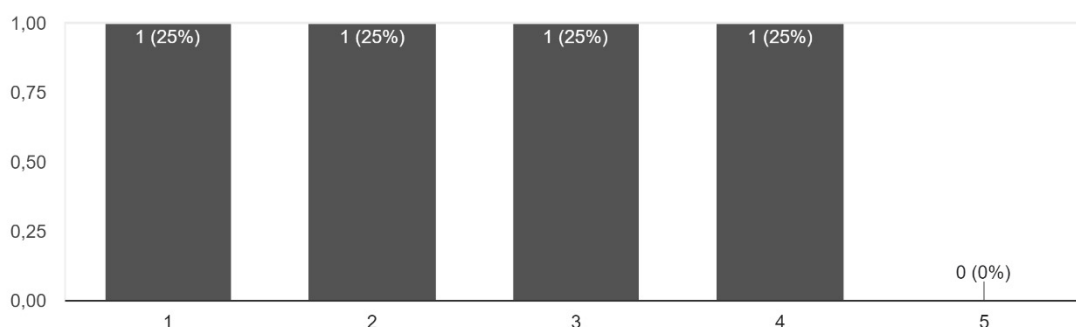


Figura 4.26: Respostas “O cenário das animações de Core Inc. é de fácil entendimento.”

algumas funcionalidades do software, como o professor D fala sobre os pontos de parada em sua entrevista.

A última seção de afirmações do método TAM indaga os professores sobre a intenção de uso do Core Inc. em seu estado atual. Esta etapa finaliza o experimento complementar e é relevante pois as opiniões construídas sobre as diversas partes do software abordadas nas demais afirmações culminam nesta intenção de uso, que é um objetivo de estudo relevante para esta pesquisa ao buscar entender a recepção dos professores ao Core Inc.

Confrontados com a frase “Recomendo a utilização do Core Inc.” as respostas demonstradas na figura 4.31 são antagônicas, pois os professores A e B não recomendariam o uso do Core Inc. e os professores C e D recomendariam. Isto é esperado dado o teor das críticas levantadas pelos dois primeiros professores, ainda que o professor B veja potencial de utilização do Core Inc. ele acredita que isso só ocorrerá com correções essenciais ao software. Já os professores C e D possuem críticas e sugestões de modificações muito mais pontuais que parecem afetar menos sua percepção de uso do Core Inc.

As animações do Core Inc. são engajante.

4 respostas

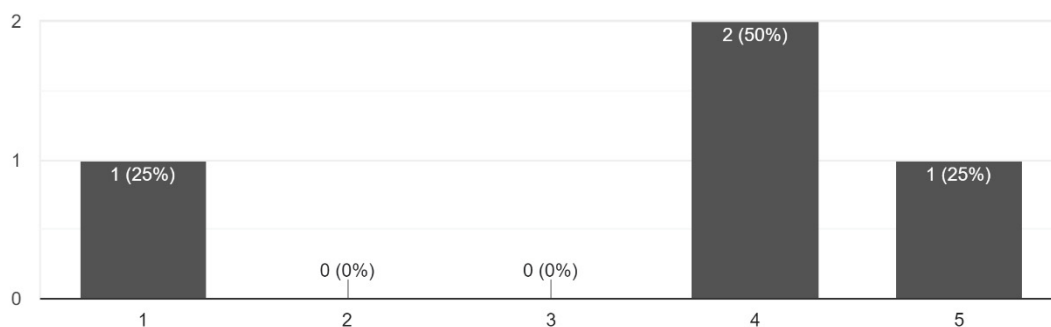


Figura 4.27: Respostas “As animações do Core Inc. são engajante.”

As animações do Core Inc. são de fácil entendimento.

4 respostas

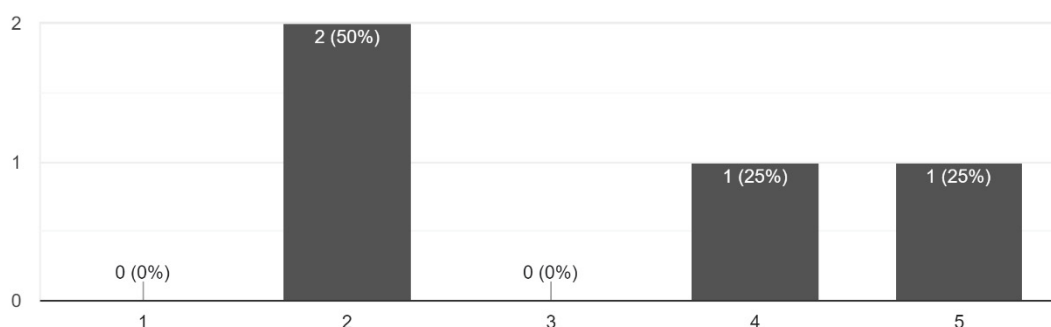


Figura 4.28: Respostas “As animações do Core Inc. são de fácil entendimento.”

Sobre a utilização do Core Inc. especificamente no ensino de algoritmos e programação a seguinte afirmação foi feita “Prefiro utilizar o Core Inc. em sala de aula para auxiliar no ensino de algoritmos e programação.”. Metade dos professores foi neutro em relação a intenção e uso especificamente neste tema segundo a figura 4.32, os professores B e D, ambos de arquitetura de software acabam tendo dúvidas sobre este uso justamente por explicarem em suas entrevistas que a utilidade do software parece ser mais adequada a suas áreas de atuação, ao mesmo tempo que não tem certeza sobre a eficácia na área de ensino de algoritmos e programação.

Na frase “Acho que o Core Inc. deveria ser utilizado nas disciplinas introdutórias de algoritmos e programação.” os professores A e C demonstram na figura 4.33 que mantém suas opiniões da afirmação anterior, entretanto os professores B e C da área de arquitetura discordam parcialmente da afirmação, pois segundo suas entrevistas a forma com que o conteúdo é apresentado não está adequado completamente a estas disciplinas. Enquanto o professor B observa erros conceituais nas animações, o professor D considera o nível de abstração dos conceitos da arquitetura de *von Neumann* em estágios muito avançados para alunos de introdução a algoritmos.

O tutorial do Core Inc. é de fácil acesso.

4 respostas

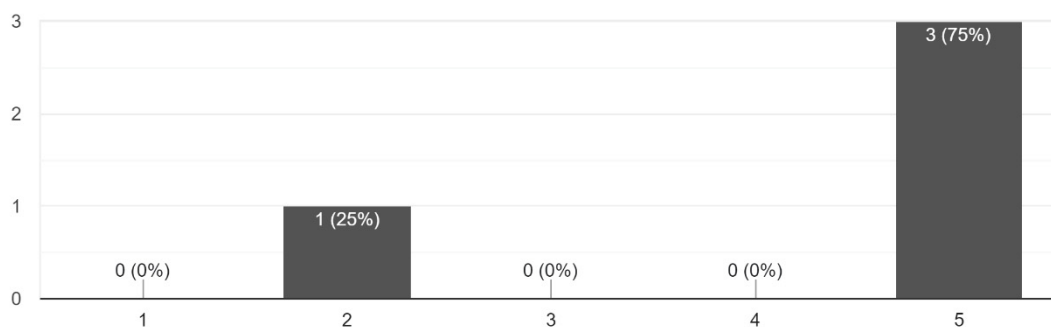


Figura 4.29: Respostas “O tutorial do Core Inc. é de fácil acesso.”

As instruções do tutorial são de fácil compreensão.

4 respostas

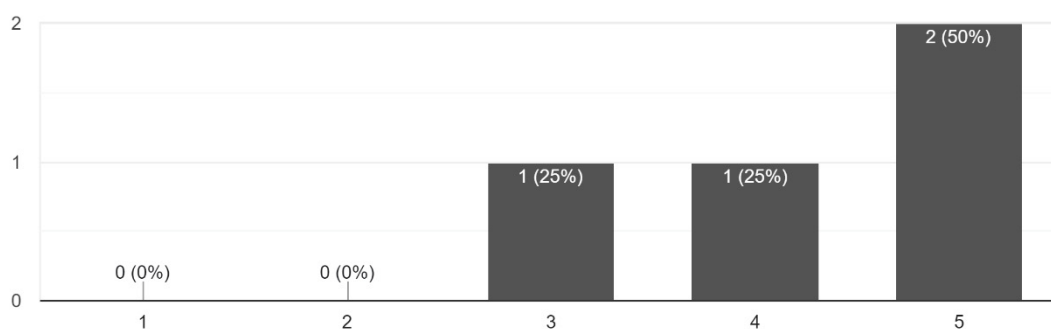


Figura 4.30: Respostas “As instruções do tutorial são de fácil compreensão.”

A última pergunta deste experimento complementar avalia o desejo e motivação dos professores em utilizar o Core Inc. em seu trabalho com a seguinte frase “Estou motivado a utilizar o Core Inc.”. Segundo a figura 4.34 o professor A mantendo sua posição apresentada durante o experimento principal e este experimento complementar não acredita que utilizaria o Core Inc. O professor B e D possuem problemas com algumas funcionalidades faltantes e correções necessárias ao software, portanto não concordam com a afirmação atualmente apesar de não descartar uso futuro caso correções sejam realizadas. A professora C assim como em sua entrevista considera o Core Inc. apto para o uso em sala de aula em seu estado atual.

4.6 CONSIDERAÇÕES

Neste capítulo foi demonstrado a metodologia de análise utilizada para a aplicação e experimentos em busca de avaliar o software de algoritmos animados proposto chamado de Core Inc. Justificando-se na literatura apresentada no capítulo 2 e nas dificuldades de realização de experimentos com alunos, foi realizado um conjunto de entrevistas qualitativas com quatro

Recomendo a utilização do Core Inc.

4 respostas

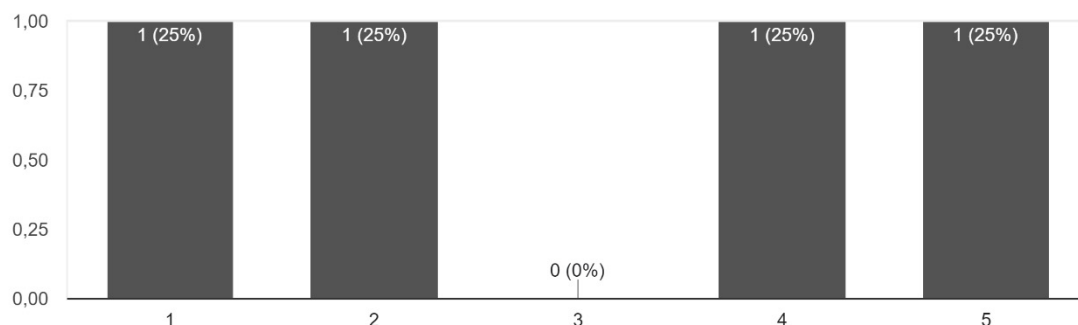


Figura 4.31: Respostas “Recomendo a utilização do Core Inc.”

Prefiro utilizar o Core Inc. em sala de aula para auxiliar no ensino de algoritmos e programação.

4 respostas

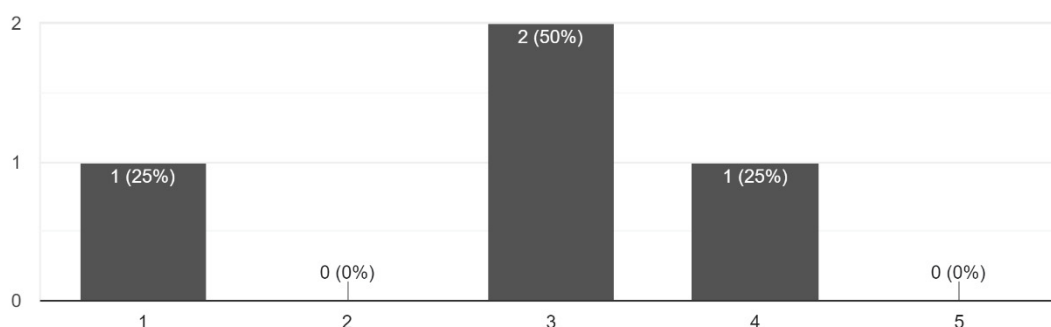


Figura 4.32: Respostas “Prefiro utilizar o Core Inc. em sala de aula para auxiliar no ensino de algoritmos e programação.”

professores de diferentes áreas junto da aplicação de um questionário para identificar o histórico de uso de ferramentas de visualização de software em conjunto com o modelo de aceitação de tecnologia.

Identificou-se por meio da análise das entrevistas e o apoio dos questionários aplicados que os professores e pesquisadores participantes dos experimentos já possuíam conhecimento e experiência prévia com ferramentas de visualização de software, mesmo que de forma limitada. Além disto, demonstrou-se que a percepção dos participantes sobre este tipo de ferramenta visual é de interesse dos mesmos e eles acreditam em seu potencial de auxílio no processo de ensino aprendizagem, o que demonstrou que as opiniões aqui coletadas não possuíam um viés negativo a princípio em relação ao Core Inc., o que era uma possibilidade dada a literatura abordada no capítulo 2.

Além disto, a maioria dos professores demonstraram-se interessados e adeptos a ferramentas que adaptem os conteúdos a cultura e dia a dia dos alunos, acreditando no seu potencial de auxiliar no ensino, eles manifestaram opiniões contundentes ao dizerem que "Eu acho

Acho que o Core Inc. deveria ser utilizado nas disciplinas introdutórias de algoritmos e programação.

4 respostas

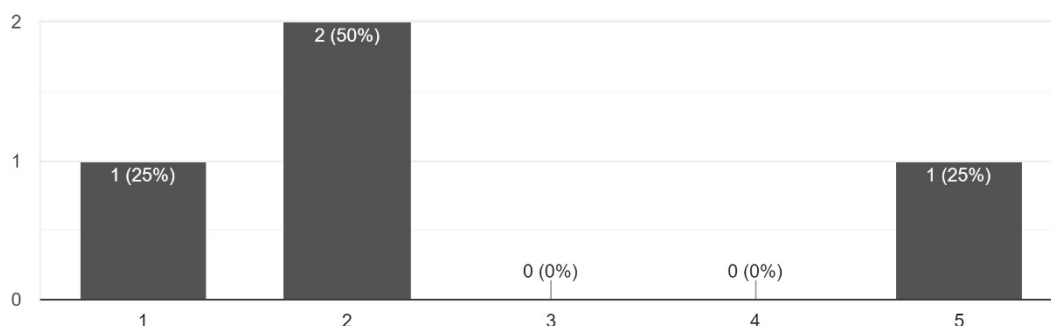


Figura 4.33: Respostas “Prefiro utilizar o Core Inc. em sala de aula para auxiliar no ensino de algoritmos e programação.”

Estou motivado a utilizar o Core Inc.

4 respostas

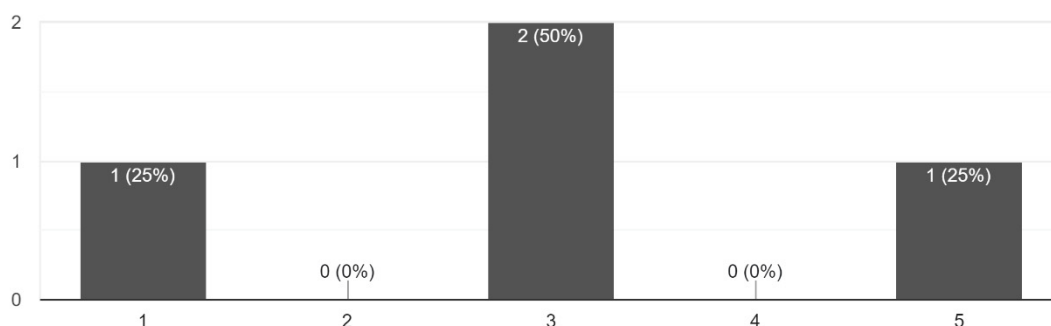


Figura 4.34: Respostas “Estou motivado a utilizar o Core Inc.”

que o apoio visual por diagramas, por desenhos, por é, enfim, que seja lá o tipo de instrumento que se use, eu acho que ele faz muita diferença."e "o visual nesse caso é importante porque é um conteúdo muito abstrato, é fora da nossa realidade.", um dos professores apesar de achar tais ferramentas potencialmente benéficas levanta uma ressalva quando diz "Eu acho que ela pode ser benéfica se ela tiver uma representação de fácil entendimento e correta do ponto de vista do que se quer passar.". Tais afirmações reforçam o resultado do questionário, ao mesmo tempo que levantam a preocupação dos professores com a qualidade das ferramentas desenvolvidas.

Com a compreensão de que os professores possuem uma opinião prévia positiva do uso de ferramentas de visualização de software e em sua maioria também concordam com ferramentas culturalmente contextualizadas, foi avaliado se os professores consideravam o conteúdo abordado pelo Core Inc, ou seja, a arquitetura de *von Neumann* relevante e os participantes foram unânimes ao concordar que ela é importante para o processo de aprendizado de algoritmos e programação, ainda que nem todos utilizem estes conceitos em seu dia a dia de trabalho, onde as seguintes

declarações confirmaram essa impressão: "Eu acho que a correta compreensão do ciclo de instrução de como é que as coisas acontecem, por que elas acontecem. Eu acho que ela ajuda bastante o aluno. ", outro professor que trabalha com arquitetura de software respondeu "Totalmente, totalmente porque em arquitetura de computadores é essencial saber o ciclo básico.", uma professora que trabalha diretamente com ensino de algoritmos e programação básica em sala de aula acrescentou que "você ter essa noção dessa arquitetura de von Neumann é importante porque facilita a compreensão na programação em si".

Sobre a introdução de elementos de contextualização ao software, utilizando a teoria da cognição situada para guiar o desenvolvimento de um contexto mais representativo do dia a dia dos alunos os professores foram indagados sobre suas opiniões acerca do tema e em sua maioria as respostas foram positivas, como quando o professor B comenta "Isso é sensacional, né? Porque eu acho que é um grande chamariz para os alunos, né?". A professora C cita Paulo Freire (Freire, 1987) ao trazer um dos exemplos presentes em seus estudos dizendo "então ele traz muito da realidade de termos e palavras que os agricultores estão acostumados para poder trabalhar alfabetização com eles." e compara diretamente com a proposta do Core Inc. afirmando assim que "(...) fazer uma analogia com a programação, que é o nosso caso, que é muito abstrato. Eu acho que isso vai facilitar ainda mais a compreensão.", o professor D também reforça o benefício do uso de ferramentas contextualizadas no momento proposto por este trabalho ao comentar "Acho interessante, principalmente para usar nas disciplinas introdutórias quando nesses conceitos que a gente tenta passar pra eles ainda são muito abstratos. ". Um dos professores, porém, apesar de acreditar no potencial deste tipo de solução contextualizada, afirma que não considera o ensino superior adequado a este tipo de solução, ao comentar que "Eu acho que faz e eu acho que em se tratando de ensino superior é preferível a representação mais abstrata.", contudo ele faz ressalvas quanto a possibilidade de tais métodos serem eficazes em outros ambientes de ensino e faixa etária de alunos. Tratando-se especificamente do Core Inc. a solução em software desenvolvida, os professores foram indagados sobre diversos aspectos do software, sendo um deles se seus tutoriais e usabilidade estão bons. As respostas recebidas foram positivas de maneira geral, com respostas sucintas como o professor A dizendo "Sim, eu acho que sim. Eu acho que sim." e o professor B "Ele foi fácil.". O resultado do questionário aplicado reforçou estas afirmações.

A temática e representação criadas no Core Inc. para abstrair os conteúdos do ciclo de instrução e modelo de *von Neumann* também foram comentadas pelos professores. Eles avaliaram que o trabalho feito é positivo, contudo alguns consideram que narrativa e seus personagens deve ser ainda mais densa, com mais diálogos, história e inserção de música e efeitos sonoros. O professor B comenta que "Você tem que dar uma pirada, os alunos vão falar assim, ah tá meio bobinho, mas aluno adora coisa bobinha assim, né? Nós adoramos coisas bobinhas.", a professora C elogia a escolha da temática ao dizer que "Porque a gente trabalha atuando em sistemas que vão estar dentro de fábricas ou vem alunos que a gente sabe dos que trabalham em fábricas, têm a noção, tem a ideia, o conhecimento, a prática. ". O professor D elogia especificamente da inserção de personagens humanoides, ao comentar "Bonequinhos é bem mais interessante. ". O questionário aplicou perguntas sobre o tema em relação aos visuais das animações e personagens, em que os professores reafirmam suas opiniões, com exceção do professor A que justifica que para o ensino superior o software não seria adequado ao público.

As entrevistas também forneceram direcionamento para a revisão do Core Inc. e trabalhos futuros que podem ser desenvolvidos no mesmo. As principais modificações citadas foram a necessidade de uma revisão dos elementos da animação e como eles abstraem e demonstram o que realmente acontece no modelo de *von Neumann*, sendo que no estado atual existem falhas conceituais na maneira com que a animação demonstra certas ações e elementos, isso embasa-se em frases ditas pelos professores ao longo das entrevistas, como quando o

professor A comenta “É, e esses conceitos que no meu entender são fundamentais eles não são apresentados com clareza, com fidelidade” onde ele acaba listando alguns dos defeitos de representação ao dizer “A coisa da memória apresentada de um jeito que a memória está fora do cenário. Não é assim.” e “A entrada e a saída estão no mesmo lugar na animação, não é? Isso não corresponde ao mecanismo elementar”, o professor B também apontou problemas em relação a representação do modelo ao dizer que “A entrada e saída ali está totalmente desconectado de memória, e não é o que acontece.” e “A posição visual e a organização dos componentes, eu colocaria de outra forma que seria mais benéfico”. Esta revisão acaba sendo necessária para identificar o método correto de ajustar a animação e o posicionamento dos elementos da animação em tela para que corrijam as críticas feitas majoritariamente pelos professores A e B e é uma prioridade em correções futuras, pois afeta diretamente o processo de aprendizagem dos alunos e a eficácia do Core Inc. ao ser aplicado em sala de aula.

Também foi comentado pela maioria dos professores a necessidade de áudio durante o vídeo de apresentação da história e durante as próprias animações, onde o professor B comenta “O rolê todo do teu vídeo eu achei que podia ter uma musiquinha. (...) Alguma coisa que desse a entender que você tinha algo que passasse alguma emoção”, ele ainda se estendeu dando exemplos de músicas e áudio que esperaria encontrar no Core Inc. e contribuiriam para a imersão dos alunos, por conseguinte a professora C comenta também “A única coisa que eu senti ali, eu não sei qual que é o objetivo de vocês nesse sentido, eu senti falta do som.” reiterando a opinião levantada pelo professor B, ela também acabou comparando a presença de músicas e efeitos de áudios extensos em um trabalho anterior do autor desta dissertação, vendo como uma necessidade estar presente no Core Inc. também. Além disto todos os professores ficaram confusos durante o vídeo de apresentação quando notaram a ausência de áudio, sendo mais um indicativo de que esta necessidade deve ser atendida para melhor atender as expectativas do usuário e permitir uma imersão adequada na narrativa proposta pelo Core Inc.

Além disto, a adição de mais elementos e opções de contextualização e enredo também foram sugestões que apareceram durante a maioria das entrevistas, o professor A comentou “Então é. É pra mim não ficou muito claro, é a temática da empresa, sabe?” e “Até se poderia ter uma história no início, no vídeo ter a história. (...) Para que tenha uma narrativa em torno disso sabe, uma narrativa mais clara.”, ele pede para que o enredo do Core Inc. seja mais aprofundado ao dizer que “Você tem que dar uma pirada, os alunos vão falar assim, ah tá meio bobinho, mas aluno adora coisa bobinha assim, né? Nós adoramos coisas bobinhas”. A professora C também comenta sobre a necessidade de uma narrativa mais detalhada quando comentou “Se tivesse uma narrativa bem curta, não precisava ser longa ali.” justificando essa necessidade com “Eu acho que facilita tanto principalmente para quem é iniciante.”, ela também comenta a possibilidade de personalizar o quanto de narrativa será inserida nas animações de acordo com o desejo do aluno ao explicar “E você dá a opção de ligar ou desligar a narrativa, porque quando ele tiver já habituado com o software, tem gente que nem vai querer ouvir.”. Estes pontos das entrevistas com os professores B e C demonstraram que atenção adicional a narrativa do Core Inc. é relevante e adições na percepção destes professores podem beneficiar o impacto final do software e deve ser uma das prioridades no processo de correção do Core Inc.

Outros comentários pertinentes quanto a modificações do Core Inc. dizem a respeito da adequação dos termos utilizados pelo software quando o professor B comenta “E deixa todo em português tchê, metade em português, metade inglês. Lá em cima código, lá em baixo input. Por que não é código de entrada? Escolhe uma língua e vai até o fim com aquela língua.”. Este foi considerado um apontamento relevante, pois alunos que estão iniciando no aprendizado de programação e algoritmos não possuem em sua maioria conhecimento de termos estrangeiros e

jargões da área de programação, além de que é uma adequação de nível de complexidade baixo a ser implementada.

A adição de um indicador no campo de texto do código-fonte de qual linha está sendo executada no momento da animação também foi uma funcionalidade comentada pelos professores A, B e D, sendo que o professor D especificamente solicitou esta funcionalidade ao dizer que “Uma coisa que eu achei que podia ajudar seria se você tivesse alguma indicação no código para dizer em qual linha está.”. Por ser comentada por quase todos os professores entrevistados considera-se que também é algo relevante a ser implementado no futuro. Junto disto o professor D também comenta alguma confusão com o funcionamento dos pontos de parada (*breakpoint*) de animação que o Core Inc. possui ao dizer que “Eu acho que no momento do tutorial, lá no início, se você colocar, por exemplo, um exemplozinho do que é o breakpoint?”, isto acaba por ser uma situação similar aos termos estrangeiros utilizados pelo Core Inc. atualmente, onde o conceito destes pontos de parada pode não ser claro para o aluno e devem ser demonstrados detalhadamente no futuro com a adição de informações adicionais no tutorial do Core Inc.

Junto das modificações sugeridas consideradas mais relevantes a partir da análise e conclusões obtidas das entrevistas, os professores também forneceram sugestões adicionais de funcionalidades ao Core Inc., que apesar de não serem consideradas necessárias ao escopo proposto do software podem ser benéficas ao expandi-lo no futuro.

Dentre elas o professor B comenta sobre o desenvolvimento de um manual de professor mais detalhado para o uso do Core Inc. ao dizer que “Ai eu não sei quais são as limitações do seu do seu modelo (...) é importante você deixar claro quais são as limitações em algum lugar.”, pois isto permitiria os professores a trabalharem melhor ao redor das limitações do Core Inc. O professor B também comenta que o software poderia ser aplicado em aulas de arquitetura de computadores, fornecendo suporte a assembly e outros modelos de arquitetura como ao dizer respectivamente “Poderia ser uma extensão interessante, sabe, talvez você escrever código em assembly.” E “A simulação do mips, por exemplo, a simulação do mips eu acho que seria totalmente passível assim de ser mostrado o que está rolando.”.

Adições a contextualização das animações também foram comentadas, quando a professora C levanta a possibilidade de trabalhar o Core Inc. com públicos diferentes dos propostos inicialmente, onde temáticas diversas seriam interessantes para se adequarem ao cotidiano de cada público, nisto ela diz “Sei lá, o contexto da escola, um contexto do supermercado, não sei, mas um contexto, mais cotidiano sabe que daí o software poderia ser usado, o teu software poderia ser usado em contextos diferentes, não só na graduação, na universidade”. Tal adição pode ser sim relevante caso no futuro o Core Inc. demonstre-se uma boa ferramenta em sala de aula e exista o desejo de expandi-lo para outros públicos.

Um modo com mais interatividade também foi proposto pelo professor A como uma alternativa para diminuir a passividade do software segundo ele, onde ele comenta “Por exemplo, se você tivesse, botões diferentes para ações diferentes e o aluno tivesse que acertar a próxima ação. Quer dizer o que vai acontecer agora, ela vai levar o bonequinho, vai levar a memória para a mesa da entrada e saída, então ele tem que apertar o X. E se ele apertar uma outra tecla é um erro.”, indicando o desejo que o Core Inc. possua elementos que exijam interação obrigatória do aluno para que a animação seja realizada corretamente. Apesar de tal funcionalidade ir além do escopo de um software de algoritmos animados, esta possibilidade também pode ser uma opção interessante, ainda mais se não for obrigatória, pois possibilita aos professores a escolha de abordagem ideal a cada ambiente de ensino.

Por fim, a possibilidade de controle de velocidade das animações além de pausá-las e o tratamento melhor das mensagens de erro também foi comentado por alguns entrevistados, sendo com mensagens de erro mais claras ou a própria animação possuindo cliques específicos para

demonstrar os erros através de seus personagens como o professor D comenta “Seria interessante, não é? Vai que dá várias luzes na fábrica e eles se enlouquecem”.

Quanto a utilização do Core Inc. em sala de aula os professores majoritariamente consideram necessário correções para o uso do software com alunos, contudo apontam a possibilidade de utilização futura. Os participantes envolvidos com a área de arquitetura de software comentaram sobre a aplicação do Core Inc. ser ainda mais adequada ao ensino de arquitetura de computadores, com a possibilidade de uso como modelo de *von Neumann* e adições futuras da simulação de outros modelos arquiteturais. De seu método de aplicação eles comentam a utilização do software para exemplificar o modelo de *von Neumann* durante a aula, servindo de apoio ou até mesmo substituindo os exemplos construídos com desenhos no quadro, dado a praticidade e melhor apresentação do conteúdo que o Core Inc. pode proporcionar aos alunos. Além disto, uma parcela dos professores cita o Core Inc. como ponto de partida para atividades diferentes das que executam hoje em sala de aula, com a proposta de desenvolvimento de discussões do funcionamento de códigos baseado nas animações do Core Inc.

Dado esta análise final, compreende-se que as entrevistas geradas com o apoio do questionário e o modelo de aceitação de tecnologia contribuíram ao identificar que os conceitos propostos nesta dissertação são de interesse dos professores e benéficos ao ensino de algoritmos e programação, assim como o enfoque no ensino do ciclo de instrução do modelo de *von Neumann*, que também foi considerado parte essencial do currículo segundo os entrevistados. Quanto a percepção do software proposto utilizando estes conceitos, foram gerados dados objetivos e quantitativos com opiniões detalhadas de cada característica do Core Inc. através das entrevistas realizadas, propiciando material para demonstrar que a recepção do software foi em geral positiva, apesar de ressalvas serem feitas quando sugestões de melhorias e aumento de escopo são propostas, o que permite delinear ao fim desta seção nas tabelas 4.5 as modificações necessárias ao Core Inc. segundo o grau de prioridade estabelecido pelo autor dada sua percepção das entrevistas e a análise final realizada nesta seção, enquanto na tabela 4.6 sugestões adicionais que podem ser benéficas no futuro são listadas também de acordo com seu nível de prioridade. No próximo capítulo as conclusões obtidas são demonstradas dados os objetivos inicialmente definidos.

Tabela 4.5: Sugestões de Modificação Necessárias

Prioridade	Correção/Modificação
1	Revisar e ajustar a abstração realizada dos conceitos do modelo de <i>von Neumann</i> para a animação, de forma a corrigir imprecisões quanto a representação do conteúdo abordado pelo Core Inc.
2	Revisar e ajustar o posicionamento dos elementos das animações e o movimento dos personagens para melhor corresponder como ocorre a execução de instruções no modelo de <i>von Neumann</i>
3	Adicionar elementos sonoros ao vídeo introdutório de contextualização do Core Inc. e nas animações.
4	Adicionar mais elementos de contextualização através do enredo, com um melhor desenvolvimento e exposição dos personagens e mundo criado.
5	Adicionar ao tutorial uma melhor apresentação do enredo e contexto da animação e seus personagens.
6	Adicionar uma funcionalidade a ferramenta de animação que retorna a linha em execução atualmente na animação para integração com editores de texto externos.

Tabela 4.5: Sugestões de Modificação Necessárias

Prioridade	Correção/Modificação
7	Revisar os textos do Core Inc. de forma a retirar palavras em inglês, estrangeirismos e termos técnicos, de forma a deixar o software 100% em português brasileiro.
8	Adicionar ao tutorial um melhor detalhamento da funcionalidade de pontos de parada.

Tabela 4.6: Sugestões Adicionais

Prioridade	Correção/Modificação
1	Desenvolver um manual de uso do Core Inc. para professores mais detalhado do que existe atualmente.
2	Adicionar a opção de elementos narrativos adicionais nos primeiros ciclos de animação, caso o aluno deseje.
3	Adicionar mais contextos de animação para escolha do usuário, permitindo uma adequação ao contexto de mais pessoas externas à comunidade de alunos da universidade.
4	Adicionar o tratamento de erros durante a execução da animação, demonstrando através da animação de forma lúdica e visual um erro ao invés de uma mensagem padrão.
5	Adicionar controle de velocidade da animação.
6	Adicionar o suporte a linguagem assembly.
7	Adicionar suporte a representação de mais modelos de arquitetura além da de <i>von Neumann</i> .
8	Adicionar um modo interativo a animação, onde o aluno necessita pressionar botões para a continuidade da execução da animação.

5 CONCLUSÃO

Nestas páginas observamos a identificação de uma problemática recorrente ao ensino de algoritmos e programação básica. Identificou-se que as características abstratas dos conteúdos a serem ministrados, juntos da complexidade do tema desmotivam alunos a prosseguir com seus estudos e causam dificuldades no aprendizado dos cursos que abordam tal tema. Motivado por esta problemática nesta dissertação foi proposto como objetivo utilizar-se de técnicas de algoritmos animados unidas com a teoria da cognição situada e características de jogos digitais para compreender se a utilização destes conceitos no ensino de programação podem ser benéficos aos estudantes e professores no processo de ensino-aprendizagem e como utilizar estes conceitos para construir uma ferramenta que auxiliasse neste processo educacional no momento do ensino da arquitetura de *von Neumann*, conhecimento essencial para o aprendizado de algoritmos e programação segundo a literatura pesquisada e a opinião de professores entrevistados durante experimentos realizados.

O estudo teórico realizado para o levantamento de conceitos que embasaram esta pesquisa foi feito de modo a compreender tanto seu embasamento educativo, o histórico do tipo de software proposto e conceitos fundamentais para a compreensão dos métodos de desenvolvimento da ferramenta de algoritmos animados e seu processo avaliativo. Portanto, um estudo taxonômico da área demonstrou as definições que limitam o que é uma solução de algoritmos e como ela está inserida em um âmbito mais abrangente de soluções da área de visualização de software. Na sequência um histórico do desenvolvimento de soluções de algoritmos animados que posteriormente foram suplantadas por um número muito maior de soluções de visualização de software em geral foi apresentado, estas que apesar de não realizarem diretamente o proposto por esta dissertação contribuíram para o desenvolvimento dos projetos na área.

Não só foi possível observar a construção das diversas ferramentas de algoritmos animados e visualização de software ao longo dos últimos anos, como também estudos sobre a recepção e resultados obtidos por este tipo de solução. Através destes estudos notou-se a dificuldade em comprovar a efetividade de softwares de algoritmos animados na melhora de desempenho dos alunos em sala de aula e a resistência de professores em utilizá-las dado a motivos de efetividade, organizacionais e dificuldade de aplicação em sala de aula. Além de identificar uma mudança de paradigma ao longo dos anos com soluções de visualização de software focando no potencial motivador que as ferramentas desta área possuem com os alunos. Neste levantamento apontou-se que a reprodução de ferramentas com propostas já construídas anteriormente com visuais mais chamativos e próximos a jogos digitais não seria suficiente para atingir os objetivos pretendidos.

Em busca de acrescentar a literatura já produzida na área a teoria da cognição situada foi demonstrada nesta dissertação como uma teoria cognitiva que considera que a construção do conhecimento se dá pela interação constante do indivíduo com o conhecimento e seu contexto, apresentando um novo paradigma relevante a ser considerado no processo educacional de construção de softwares de algoritmos animados, pois a cognição situada não embasou nenhum dos outros trabalhos demonstrados pela revisão de literatura realizada. Esta teoria cognitiva foi detalhada durante o levantamento teórico desta dissertação e utilizado como base para o desenvolvido dos conceitos a serem validados durante este trabalho.

Com as bases do referencial teórico coletado e as análises feitas do histórico de desenvolvimento e desempenho de software similares foi desenvolvido uma ferramenta de

algoritmos animados denominada Core Inc. com o intuito de validar os conceitos propostos e atingir o objetivo desta dissertação. Ao longo do texto também foi descrito o processo de concepção do software, seu escopo e como a teoria da cognição situada embasou e influenciou a tomada de decisão nos elementos visuais das animações, além de como se deu a estruturação deste software para que pudesse ser modificado e utilizado com mais facilidade posteriormente.

Como subproduto deste objetivo também foi gerada e disponibilizada¹ a ferramenta Core Inc. de algoritmos animados capaz de ser inserida em qualquer ambiente de software que suporte às tecnologias web mais atuais, sendo elas: HTML, CSS e Javascript e com suas animações independentes de linguagem e programação a ser interpretada previamente.

Devido ao processo de construção desta dissertação ocorrer durante a pandemia de COVID-19, não foi possível realizar experimentos com alunos, devido a ausência de aulas presenciais e disciplinas que se encaixassem com os temas da pesquisa. Por fim a junção de conceitos propostos nesta dissertação e o software desenvolvido para validá-los foi avaliado através de entrevistas qualitativas e um questionário do modelo de aceitação de tecnologia aplicado com quatro professores voluntários. Identificou-se através deste processo que os professores não possuíam opiniões prévias negativas sobre algoritmos animados em relação ao seu uso e efetividade, o que era uma preocupação segundo os estudos realizados anteriormente e um dos objetivos adicionais desta dissertação. Os conceitos levantados nesta dissertação como fundamentos do desenvolvimento de softwares de algoritmos animados foi bem recebida pelos professores entrevistados, havendo unanimidade ao considerar o apoio de ferramentas de visualização e algoritmos animados benéfica aos estudantes, assim como a aplicação de características visuais de jogos digitais e a contextualização das ferramentas educativas com o dia a dia dos alunos, com uma única ressalva sendo feita por um dos professores, ao considerar representações contextualizadas não adequadas aos alunos de ensino superior.

Quanto a avaliação do Core Inc. especificamente os professores tiveram opiniões positivas quanto aos elementos de tutorial e usabilidade, assim como sua temática, contextualização e animações, ainda que acreditem que há necessidade de correção de algumas representações conceituais e uma expansão da narrativa que contextualiza o Core Inc. Por fim, um deles considera a ferramenta apta para uso, dois deles acreditam que o Core Inc. para completo uso em sala de aula necessita das correções levantadas pela análise dos resultados e um dos professores não acredita que o software é adequado ao ensino superior, ainda que não descarte seu uso para outras faixas etárias.

Ao analisar o projeto desta dissertação entende-se que o principal objetivo de demonstrar a utilização dos novos conceitos propostos na área de algoritmos animados é atingido, assim como sua avaliação através de entrevistas e o software Core Inc.. Como subproduto desta dissertação o software proposto também é disponibilizado para uso da comunidade científica com a devida documentação da aplicação já disponível, além disto os objetivos adicionais que tratam de identificar a recepção dos professores entrevistados para com softwares de algoritmos animados e o Core Inc. também foi atingida através dos métodos avaliativos propostos, por fim o planejamento da sequência do trabalho, o que segundo as avaliações realizadas permitiria um aprimoramento da aplicação dos conceitos propostos e preparação para o Core Inc. estar melhor preparado para futura aplicação em sala de aula com alunos.

Acredita-se que o processo de desenvolvimento do Core Inc. desde seu levantamento teórico até sua avaliação final demonstra o potencial da utilização de algoritmos animados de forma contextualizada para o ensino de algoritmos e programação e que soluções embasadas na teoria da cognição situada possuem interesse e aceitação por parte dos professores e pesquisadores da área. As avaliações realizadas dos conceitos e software desenvolvido auxiliam a elucidar

¹Repositório Core Inc.: <https://github.com/Vinicius-Pereira/core-inc-animation>

quais são os pontos relevantes a serem aprimorados e corrigidos em trabalhos futuros, como as tabelas 4.5 e 4.6 demonstram, enquanto apresenta uma recepção positiva ao estudo, de maneira a incentivar o desenvolvimento de mais ferramentas que façam uso dos conceitos apresentados neste trabalho.

Ao retornar ao problema identificado no início desta dissertação e seu principal motivador, compreende-se que foi demonstrado o potencial da utilização de algoritmos animados de forma contextualizada para o ensino de algoritmos e programação e que soluções embasadas na teoria da cognição situada possuem interesse e aceitação pelos professores e pesquisadores da área, isto junto da aplicação destes conceitos no Core Inc. demonstra como é possível melhorar o ensino de algoritmo e programação através de softwares contextualizados, assim como levanta novas questões e propostas de expansão e melhorias para estudos futuros.

REFERÊNCIAS

- Al-Sakkaf, A., Omar, M. e Ahmad, M. (2019). A systematic literature review of student engagement in software visualization: a theoretical perspective. *Computer Science Education*, 29(2-3):283–309.
- Amorim, R. V. e de Rezende, P. J. (1993). Compreensão de algoritmos através de ambientes dedicados a animação. Relatório técnico, Departamento de Ciência da Computação - Universidade Estadual de Campinas, Campinas SP.
- Andrade, C. C. e Holanda, A. F. (2010). Apontamentos sobre pesquisa qualitativa e pesquisa empírico-fenomenológica. *Estudos de Psicologia Campinas*, 27(2):259–268.
- Baloukas, T. (2012). Javenga: Java-based visualization environment for network and graph algorithms. *Computer Applications in Engineering Education*, 20(2):255–268.
- Bateson, G. (1972). *Steps to An Ecology of Mind*. Chandler Publishing Company.
- Bentley, J. e Kernighan, B. (1991). A system for algorithm animation. *Computing Systems*, 4(1):5–30.
- Brown, J. S., Collins, A. e Duguid, P. (1989). Situated cognition and the culture of learning. *Educational Researcher*, 18(1):32–42.
- Brown, M. H. (1988). Exploring algorithms using balsa-ii. *Computer*, 21(5):14–36.
- Brown, M. H. (1991). Zeus: A system for algorithm animation and multi-view editing. Em *IEEE Workshop on Visual Languages*, páginas 4–9, Kobe, Japan.
- Brown, M. H. e Sedgewick, R. (1985). Techniques for algorithm animation. *IEEE Computer Society*, 2(1):28–39.
- Buchanan, S. e Laviola, J. J. (2014). Cstutor: A sketch-based tool for visualizing data structures. *ACM Transactions on Computing Education*, 14(1).
- Campos, R. (2010). Metodologia erm2c: Para melhoria do processo de ensinoaprendizagem de logica de programação. Em *XVIII Workshop sobre Educação em Computação*, páginas 961–970, Minas Gerais - Brasil.
- Carlson, P., Burnett, M. e Cadiz, J. (1996). A seamless integration of algorithm animation into a visual programming language. Em *International Wotlshop on Advanced Visual Interface*, páginas 194–202, Gubbio, Itália.
- Castilho, M., Silva, F. e Weingaertner, D. (2020). *Algoritmos e Estruturas de Dados 1*. UFPR.
- Clancey, W. J. (1997). *Situated Cognition: On Human Knowledge and Computer Representations*. Cambridge.
- da Rosa, A., de Jesus, A., Igarashi, G. V. e Pereira, V. S. (2020). Iron ears: primeiras impressões de um jogo educativo para ensino de estrutura de dados lineares. *Workshop sobre Educação em Computação*, 28(1):31–35.

- da Silva, P. M., Dias, G. A. e de Almeida, J. R. (2009). Modelo de aceitação de tecnologia (tam) aplicado ao sistema de informação da biblioteca virtual em saúde (bvs) nas escolas de medicina da região metropolitana do recife. *Informação e Sociedade: Estudos*, 19(1).
- da Silva, T. R., Medeiros, T. J., Medeiros, H., Lopes, R. e Aranha, E. (2015). Ensino-aprendizagem de programação: uma revisão sistemática da literatura. *Revista Brasileira de Informática na Educação*, 23(1):182–196.
- Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly*, 13(3):319–340.
- de B. Campos, L. F. e Venâncio, L. S. (2007). Perspectivas em (in)formação: tendências e tensões entre abordagens físicas, cognitivistas e emergentes. *TransInformação*, 19(2):107–118.
- de L. Junior, J. A. T., Vieira, C. E. C. e de P. Vieira, P. (2015). Dificuldades no processo de aprendizagem de algoritmos: uma análise dos resultados na disciplina de al1 do curso de sistemas de informação da faeterj - campus paracambi. *Cadernos UniFOA*, 10(27):5–15.
- Demetrescu, C. e Finocchi, I. (2001). Smooth animation of algorithms in a declarative framework. *Journal of Visual Languages and Computing*, 12(1):253–281.
- DiCicco-Bloom, B. e Crabtree, B. F. (2006). The qualitative research interview. *Medical Education*, 40(4):314–321.
- Dillon, A. e Morris, M. G. (1996). User acceptance of information technology: Theories and models. *Annual Review of Information Science and Technology (ARIST)*, 31:3–32.
- dos Santos, R. P. e Costa, H. A. X. (2005). Tbc-aed: Um software gráfico para apresentação de algoritmos e estruturas de dados aos iniciantes em computação e informática. Em *Congresso de Computação do Sul do Mato Grosso*, Mato Grosso, Brasil.
- Eck, R. V. (2006). Digital game-based learning: It's not just the digital natives who are restless.... *EDUCAUSE Review*, 41(2):1–16.
- Falkembach, G. A. M., Tarouco, L. R. e Amoretti, M. S. M. (2003). Aprendizagem de algoritmos: Uso da estratégia ascendente de resolução de problemas. Em *Taller Intercacional de Software Educativo*, Santiago, Chile.
- Farrugia, M. e Quigley, A. (2011). Effective temporal graph layout: A comparative study of animation versus static display methods. *Computer Applications in Engineering Education*, 10(1):47–64.
- Flick, U., von Kardorff, E. e Steinke, I. (2004). *A Companion to Qualitative Research*. Sage Publications.
- Freire, P. (1987). *Pedagogia do oprimido*. Editora Paz e Terra.
- Games, H. R. (2020). Ikenfell. <https://store.steampowered.com/app/854940/Ikenfell/>. Acessado em 11/13/2020.
- Garcia, I. C., de Rezende, P. J. e Calheiros, F. C. (1997). Astral: Um ambiente para ensino de estruturas de dados através de animações de algoritmos. *Revista Brasileira de Informática na Educação*, 1(1):71–80.

- Gerring, J. (2017). Qualitative review. *Annual Review of Political Science*, 20(1):15–36.
- Giraffa, L. M. M. e da C. Mora, M. (2013). Evasão na disciplina de algoritmo e programação: Um estudo a partir dos fatores intervenientes na perspectiva do aluno. Em *Tercera Conferencia sobre el Abandono en la Educación Superior*, Cidade do México, México.
- Goldman, K. J. (2004). An interactive environment for beginning java programmers. *Science of Computer Programming*, 53(1):3–24.
- Gomes, A., Mendes, A. J. e Marcelino, M. J. (2004). Avaliação e evolução de um ambiente de suporte à aprendizagem de programação. Em *VII Congresso Iberoamericano de Informática Educativa*, páginas 187–196.
- González, F. E. (2020). Reflexões sobre alguns conceitos da pesquisa qualitativa. *Revista Pesquisa Qualitativa*, 8(17):155–183.
- Hannabuss, S. (1996). Research interviews. *New Library World*, 97(1129):22–30.
- Helminen, J. e Malmi, L. (2010). Jype - a program visualization and programming exercise tool for python. *Proceedings of the ACM Conference on Computer and Communications Security*, páginas 153–162.
- Hosseini, R., Sirkiä, T., Guerra, J., Brusilovsky, P. e Malmi, L. (2016). Animated examples as practice content in a java programming course. Em *SIGCSE 2016 - Proceedings of the 47th ACM Technical Symposium on Computing Science Education*, páginas 540–545.
- Hundhausen, C. D., Douglas, S. A. e Stasko, J. T. (2002). A meta-study of algorithm visualization effectiveness. *Journal of Visual Language and Computing*, 13(3):259–290.
- Kölling, M. (2008). Greenfoot. *ACM SIGCSE Bulletin*, 40(3):327.
- Lampert, M. (1986). Knowing, doing, and teaching multiplication. *Cognition and Instruction*, 3(4):305–342.
- Lave, J. (1988a). *Cognition in practice*. Cambridge.
- Lave, J. (1988b). The culture of acquisition and the practice of understanding. Relatório técnico, CA: Institute of Research on Learning, Palo Alto.
- Lawrance, A. W., Badre, A. M. e Stasko, J. T. (1994). Empirically evaluating the use of animations to teach algorithms. Em *IEEE Symposium on Visual Languages*, páginas 48–54, Saint Louis, USA.
- Leydens, J., Moskal, B. e Pavelich, M. (2004). Qualitative methods used in the. journal of engineering education. *Journal of Engineering Education*, 93(1):65–72.
- Magro, M. C. (1999). *Linguajado o linguajar da biologia a linguagem*. Tese de doutorado, Universidade Estadual de Campinas - Instituto de Estudos da Linguagem, Campinas - Brasil. 287 pgs.
- Maloney, J., Resnick, M., Rusk, N., Silverman, B. e Eastmond, E. (2010). The scratch programming language and environment. *ACM Transactions on Computing Education*, 10(4):1–15.

Mandow, L., Villalba, F. e Coego, J. (2010). Uso de animaciones para la enseñanza de algoritmos de búsqueda en inteligencia artificialn. *XVI Jornadas de Enseñanza Universitaria de la Informática.*, páginas 461–468.

Manzini, E. J. (2012). Uso da entrevista em dissertações e teses produzidas em um programa de Pós-graduação em educação. *Revista Percurso*, 4(2):149–171.

Mathieson, K., Peacock, E. e Chin, W. W. (2001). Extending the technology acceptance model: The influence of perceived user resources. *The DATA BASE for Advances in Information Systems*, 32(3):86–112.

Maturana, H. e Varela, F. (1995). *A árvore do conhecimento*. Editorial Psy.

Merleau-Ponty, M. (1999). *Fenomenologia da percepção*. Martins Fontes.

Myller, N., Bednarik, R. e Sutinen, E. (2009). Extending the engagement taxonomy : Software visualization and collaborative learning. *ACM Transactions on Computing Education*, 9(1).

Naps, T. L., Fleischer, R., McNally, M., Rößling, G., Hundhausen, C., Rodger, S., Almstrum, Vickiand Korhonen, A., Velázquez-Iturbide, J. , Dann, W. e Malmi, L. (2002). Exploring the role of visualization and engagement in computer science education. Em *Proceedings of the Conference on Integrating Technology into Computer Science Education, ITiCSE*, páginas 131–152.

Prat, C. S., Madhyastha, T. M., Mottarella, M. J. e Kuo, C. H. (2020). Relating natural language aptitude to individual differences in learning programming languages. *Scientific Reports*, 10(1):1–10.

Price, B. A., Baecker, R. M. e Small, I. S. (1993). A principled taxonomy of software visualization. *Journal of Visual Languages and Computing*, 4(3):211–266.

Qu, S. Q. e Dumay, J. (2011). Qualitative research interview. *Qualitative Research in Accounting and Management*, 8(3):238–264.

Rogers, S. (2016). *Level Up: Um Guia para do Design de Grandes Jogos*. Blucher.

Roman, G.-C., Cox, K. C., Wilcox, C. D. e Plun, J. Y. (1992). Pavane: a system for declarative visualization of concurrent computations. *Journal of Visual Languages and Computing*, 3(2):161–193.

Saleh, A. M. (2004). Adoção de tecnologia: Um estudo sobre o uso de software livre nas empresas. Dissertação de Mestrado, Pós-Graduação em Administração - Universidade Federal de São Paulo, São Paulo - SP.

Schoenfeld, A. H. (1985). *Mathematical problem solving*. Academic Press.

Shaffer, Clifford A. andCooper, M. L., Alon, A. J. D., Akbar, M., Stewart, M., Ponce, S. e Edwards, S. H. (2010). Algorithm visualization: The state of the field. *ACM Transactions on Computing Education*, 10(3):1–22.

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 27(1):623–656. Reimpresso.

Silber, D. (2016). *Pixel Art for Game Developers*. CRC Press.

- Sorva, J., Karavirta, V. e Malmi, L. (2013). A review of generic program visualization systems for introductory programming education. *ACM Transactions on Computing Education*, 13(4).
- Stasko, J. T. (1990a). The path-transition paradigm: a practical methodology for adding animation to program interfaces. *Journal of Visual Languages and Computing*, 1(3):213–236.
- Stasko, J. T. (1990b). Tango: a framework and system for algorithm animation. *IEEE Computer*, 23(9):27–39.
- Tanembaum, A. S. (2013). *Organização Estruturada de Computadores*. Pearson.
- Taylor, D. S., Lurie, A. F., Horstmann, C. S., Johnson, M. B., Sharma, S. K. e Yin, E. C. (2009). Predictive vs. passive animation learning tools. Em *Proceedings of the 40th ACM technical symposium on Computer science education*, páginas 494–498, Chattanooga, USA.
- Varela, F., Thompson, E. e Rosch, E. (1991). *The Embodied Mind: Cognitive Science and Human Experience*. MIT Press.
- Venâncio, L. S. e Borges, M. E. N. (2007). Cognição situada: fundamentos e relações com a ciência da informação. *Pesquisa Brasileira em Ciência da Informação e Biblioteconomia*, 2(1).
- Venâncio, L. S. e Nassif, M. E. (2008). O comportamento de busca de informação sob o enfoque da cognição situada: um estudo empírico qualitativo. *Ciência da Informação*, 37(1):95–106.
- von Krogh, G. e Roos, J. (1995). *Organizational Epistemology*.

APÊNDICE A – PARÂMETROS DE ENTRADA DE COMANDOS CORE INC.

Abaixo são listados os parâmetros necessários para a entrada de cada tipo de comando suportado pelo Core Inc.

```

1 // Início do Programa
2 [
3     "program",
4     "linha de instrução",
5     "próxima linha de instrução/comando",
6     "nome do programa",
7 ]
8
9 // Declaração
10 [
11     "var",
12     "linha de instrução",
13     "próxima linha de instrução/comando",
14     "nome da variável",
15     "tipo da variável",
16     "valor da variável"
17 ]
18
19 // Leitura
20 [
21     "read",
22     "linha de instrução",
23     "próxima linha de instrução/comando",
24     "nome da variável",
25     "tipo da variável",
26     "valor da variável"
27 ]
28
29 // Escrita
30 [
31     "write",
32     "linha de instrução",
33     "próxima linha de instrução/comando",
34     "nome da variável",
35     "tipo da variável",
36     "valor da variável"
37 ]
38
39 // Escrita Complexa
40 [
41     "write(y + x)",
42     "linha de instrução",
43     "próxima linha de instrução/comando",
44     "nome da variável",
45     "tipo da variável",
46     "valor da variável",
47     "nome da variável 2",
48     "tipo da variável 2",
49     "valor da variável 2"(...)
50 ]
51

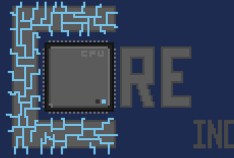
```

```
52 // Calculo e Atribuição
53 [
54     "x = y + z",
55     "linha de instrução",
56     "próxima linha de instrução/comando",
57     "expressão do cálculo",
58     "nome da variável",
59     "tipo da variável",
60     "valor da variável",
61     "nome da variável 2",
62     "tipo da variável 2",
63     "valor da variável 2"(...)
64 ]
65
66 // Controle de Fluxo - Se Senão
67 [
68     "if",
69     "linha de instrução",
70     "próxima linha de instrução/comando",
71     "expressão da comparação",
72     "resultado da comparação",
73     "nome da variável",
74     "tipo da variável",
75     "valor da variável",
76     "nome da variável 2",
77     "tipo da variável 2",
78     "valor da variável 2"(...)
79 ]
```

APÊNDICE B – CARTA DE APRESENTAÇÃO

Convite para Pesquisa

Vinicius S Pereira e Marcos Castilho



Olá!

Eu sou o **Vinicius S. Pereira** e estou desenvolvendo como dissertação de mestrado sob a orientação do professor **Castilho** um software chamado "**Core Inc.**", uma solução do que chamamos de algoritmos animados. Para os não familiares com o termo, é um tipo de programa que transforma algoritmos e código em uma animação. Neste caso, o Core Inc. transforma códigos simples em animação com a ideia de tentar ajudá-los a compreender melhor os conceitos do modelo de von Neumann e como o computador funciona executando os programas que vocês estão aprendendo agora. Neste caso, cada elemento do modelo vira um bonequinho, desse jeito aqui:

Ol. Eu sou a Úrsula!
Meu trabalho é de
Unidade Lógica e Aritmética.



Eu sou a Lucy.
A Unidade Controle.
Ao seu dispor!



Meu nome é M3M8.
Minha função é de
Barramento de
memória.



RR! Eu sou Bus.
O barramento de
entrada e saída.
RR! RR!



A grande questão é: **Não sabemos se o Core Inc realmente funciona ou ajuda os alunos.** Pra saber se o Core Inc. tem potencial pra ser útil no futuro um passo super importante é coletar a opinião justamente do público-alvo dele, no caso **você**. Pra isso gostaria que você tirasse uns minutinhos pra testar o Core Inc. e responder o questionário que desenvolvemos. Lembrando que você não precisa saber programar ainda pra usar o Core Inc., ele tem um tutorial que explica como ele funciona e um botão pra trocar entre programinhas prontos só pra você ver como funciona. Por último, mas não menos importante é importante dizer que a participação é **opcional e voluntária**.

Logo abaixo fica o link do site com acesso ao Core Inc. e ao questionário pra responder no final:

<https://vinicius-pereira.github.io/portal-core-inc/>

Figura B.1: carta de Apresentação

APÊNDICE C – CÓDIGOS DE EXEMPLO CORE INC.

Abaixo estão listados os códigos de exemplo utilizados no Core Inc.

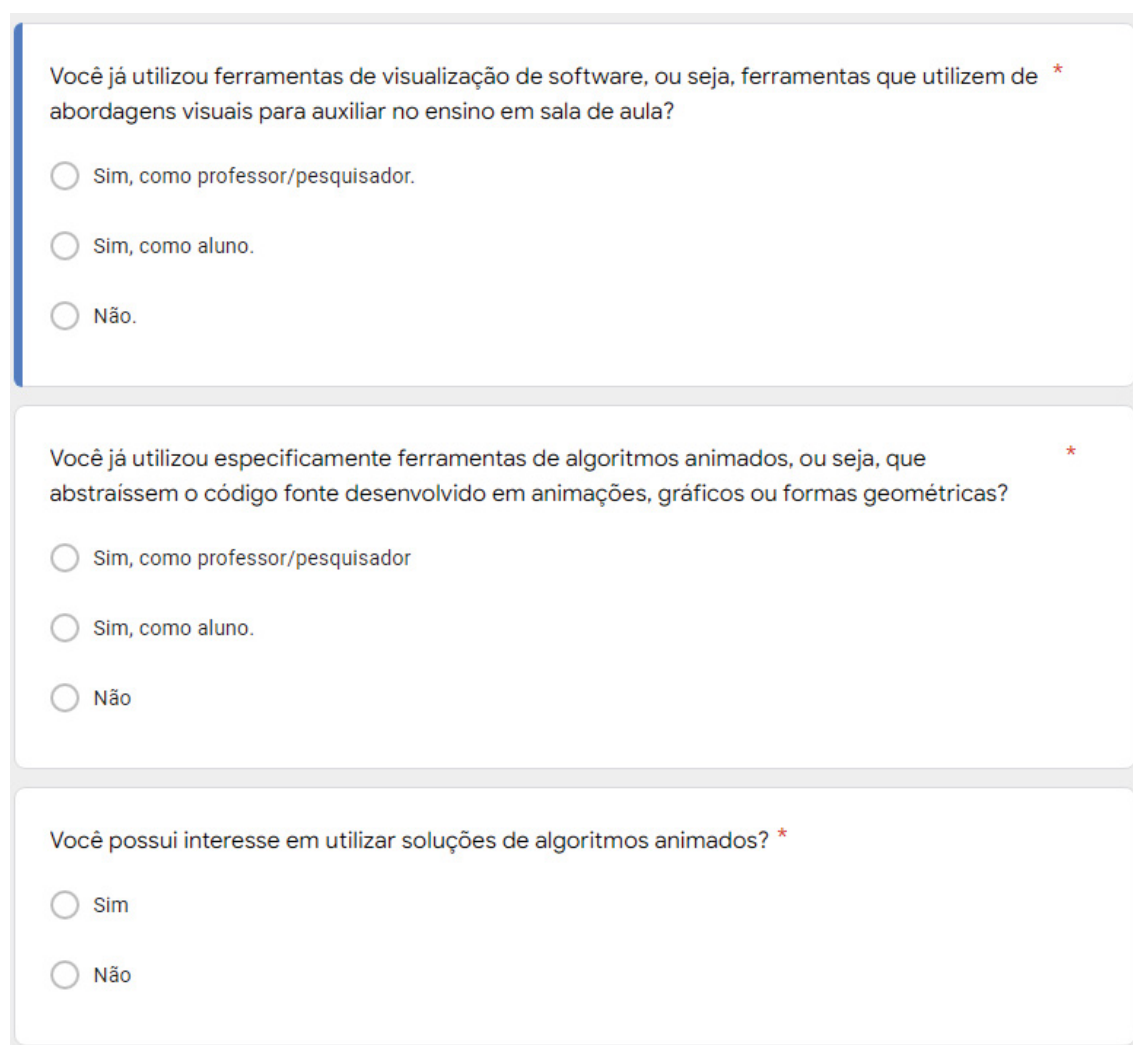
```

1 // Programa Exemplo 1
2 program Bhaskara;
3 var b, c, raizdiscriminante: real;
4 begin
5   read (b);
6   read (c);
7   raizdiscriminante := sqrt(b*b - 4*c);
8   write ((b - raizdiscriminante )/2);
9   write ((b + raizdiscriminante )/2);
10 end.
11
12 // Programa Exemplo 2
13 program entrada_saida;
14 var a: integer;
15 begin
16   read(a);
17   write(a);
18 end.
19
20 // Programa Exemplo 3
21 program incremental;
22 var i: integer;
23 begin
24   i:= 0;
25   write(i) ;
26   i:= i + 1;
27   write(i) ;
28 end.
29
30 // Programa Exemplo 4
31 program soma2numeros;
32 var a, b: integer;
33 begin
34   read(a);
35   read(b);
36   write(a+b);
37 end.
38
39 // Programa Exemplo 5
40 program imprime_se_positivo;
41 var a: integer;
42 begin
43   read(a);
44   if a > 0 then
45     writeln(a);
46 end.
47
48 // Programa Exemplo 6
49 program Hello;
50 begin
51   writeln('Bem vindo a Core Inc.!!!');
52 end.

```

APÊNDICE D – QUESTIONÁRIO EXPERIMENTO COMPLEMENTAR

Abaixo são demonstrados o questionário aplicado aos professores no experimento complementar em suas duas etapas de coleta de histórico de utilização de ferramentas de visualização de software e algoritmos animados e o modelo de aceitação de tecnologia.



Você já utilizou ferramentas de visualização de software, ou seja, ferramentas que utilizem de abordagens visuais para auxiliar no ensino em sala de aula? *

☐ Sim, como professor/pesquisador.

☐ Sim, como aluno.

☐ Não.

Você já utilizou especificamente ferramentas de algoritmos animados, ou seja, que abstraíssem o código fonte desenvolvido em animações, gráficos ou formas geométricas? *

☐ Sim, como professor/pesquisador

☐ Sim, como aluno.

☐ Não

Você possui interesse em utilizar soluções de algoritmos animados? *

☐ Sim

☐ Não

Figura D.1: Questionário Professores

Com que frequência você utiliza ou utilizou soluções de algoritmos animados e similares? *

	1	2	3	4	5	
Pouco Frequente	☐	☐	☐	☐	☐	Muito Frequente

Quanto softwares de algoritmos animados e similares você já utilizou? *

- ☐ 1
- ☐ 2
- ☐ 3
- ☐ 4
- ☐ 5 ou mais

Figura D.2: Questionário em Comum Parte 1

...

As soluções que você utilizou abordavam quais dos temas abaixo: *

- ☐ Introdução a Algoritmos
- ☐ Estruturas de Dados Lineares
- ☐ Árvores
- ☐ Grafos
- ☐ Orientação a Objetos
- ☐ Banco de Dados
- ☐ Engenharia de Software
- ☐ Outros...

De maneira geral, o quão satisfatória você considera que sua experiência com estas soluções foi? *

1 2 3 4 5

Pouco Satisfatória ☐ ☐ ☐ ☐ ☐ Muito Satisfatória

Casos se lembre, liste os softwares que você já utilizou.

Texto de resposta longa

.....

Figura D.3: Questionário em Comum Parte 2

Quais são as razões pela qual você nunca utilizou soluções de visualização de software ou algoritmos animados? *

- ☐ Não conhecia esse tipo de solução.
- ☐ Estas soluções não são eficazes.
- ☐ Não tive oportunidades.
- ☐ Não tenho os recursos necessários.
- ☐ Não tenho interesse.
- ☐ Outros...

Figura D.4: Questionário em Comum Parte 3

Utilizar o Core Inc. é satisfatório. *

1 2 3 4 5

Discordo Totalmente ☐ ☐ ☐ ☐ ☐ Concordo Totalmente

Consigo utilizar o Core Inc. sem o auxílio de um instrutor. *

1 2 3 4 5

Discordo Totalmente ☐ ☐ ☐ ☐ ☐ Concordo Totalmente

A interface do Core Inc. é de fácil entendimento. *

1 2 3 4 5

Discordo Totalmente ☐ ☐ ☐ ☐ ☐ Concordo Totalmente

Figura D.5: Questionário Método TAM Parte 2

Utilizar o Core Inc. pode melhorar o processo de ensino durante a aula. *

1 2 3 4 5

Discordo Totalmente ☐ ☐ ☐ ☐ ☐ Concordo Totalmente

Usar o Core Inc. auxiliaria no ensino de algoritmos e programação em sala de aula. *

1 2 3 4 5

Discordo Totalmente ☐ ☐ ☐ ☐ ☐ Concordo Totalmente

O Core Inc. auxiliaria durante as aulas. *

1 2 3 4 5

Discordo Totalmente ☐ ☐ ☐ ☐ ☐ Concordo Totalmente

Consigo atingir meus objetivos como professor mais facilmente com o auxílio do Core Inc. *

1 2 3 4 5

Discordo Totalmente ☐ ☐ ☐ ☐ ☐ Concordo Totalmente

Figura D.6: Questionário Método TAM Parte 3

Utilizar o Core Inc. pode melhorar a qualidade de meu trabalho. *

1 2 3 4 5

Discordo Totalmente ☐ ☐ ☐ ☐ ☐ Concordo Totalmente

Utilizar o Core Inc. pode tornar as aulas mais produtivas. *

1 2 3 4 5

Discordo Totalmente ☐ ☐ ☐ ☐ ☐ Concordo Totalmente

Utilizar o Core Inc. pode melhorar a eficiência do tempo gasto em sala de aula. *

1 2 3 4 5

Discordo Totalmente ☐ ☐ ☐ ☐ ☐ Concordo Totalmente

Figura D.7: Questionário Método TAM Parte 4

Recomendo a utilização do Core Inc. *

1 2 3 4 5

Discordo Totalmente ☐ ☐ ☐ ☐ ☐ Concordo Totalmente

Prefiro utilizar o Core Inc. em sala de aula para auxiliar no ensino de algoritmos e programação. *

1 2 3 4 5

Discordo Totalmente ☐ ☐ ☐ ☐ ☐ Concordo Totalmente

Acho que o Core Inc. deveria ser utilizado nas disciplinas introdutórias de algoritmos e programação. *

1 2 3 4 5

Discordo Totalmente ☐ ☐ ☐ ☐ ☐ Concordo Totalmente

Estou motivado a utilizar o Core Inc. *

1 2 3 4 5

Discordo Totalmente ☐ ☐ ☐ ☐ ☐ Concordo Totalmente

Figura D.8: Questionário Método TAM Parte 5

O Core Inc. possui um visual engajante. *						
	1	2	3	4	5	
Discordo Totalmente	☐	☐	☐	☐	☐	Concordo Totalmente

Os personagens do Core Inc. são bonitos. *						
	1	2	3	4	5	
Discordo Totalmente	☐	☐	☐	☐	☐	Concordo Totalmente

O cenário das animações de Core Inc. é de fácil entendimento. *						
	1	2	3	4	5	
Discordo Totalmente	☐	☐	☐	☐	☐	Concordo Totalmente

O cenário das animações de Core Inc. é bonito. *						
	1	2	3	4	5	
Discordo Totalmente	☐	☐	☐	☐	☐	Concordo Totalmente

Figura D.9: Questionário Método TAM Parte 6

⋮

As animações do Core Inc. são engajante. *

	1	2	3	4	5	
Discordo Totalmente	☐	☐	☐	☐	☐	Concordo Totalmente

As animações do Core Inc. são de fácil entendimento. *

	1	2	3	4	5	
Discordo Totalmente	☐	☐	☐	☐	☐	Concordo Totalmente

O tutorial do Core Inc. é de fácil acesso. *

	1	2	3	4	5	
Discordo Totalmente	☐	☐	☐	☐	☐	Concordo Totalmente

As instruções do tutorial são de fácil compreensão. *

	1	2	3	4	5	
Discordo Totalmente	☐	☐	☐	☐	☐	Concordo Totalmente

Figura D.10: Questionário Método TAM Parte 7

APÊNDICE E – ROTEIRO ENTREVISTA QUALITATIVA

Abaixo é demonstrado o roteiro da entrevista qualitativa aplicada de acordo com o método fenomenológico:

- Introdução
 1. Informar o entrevistado que a entrevista está sendo gravada.
 2. Qual é a relevância que o modelo de von Neumann e os temas abordados pelo Core Inc. tem no seu dia a dia de trabalho.
 3. Qual a sua percepção sobre ferramentas de visualização de software e algoritmos animados.
 4. Qual a sua opinião sobre conteúdos adaptados a realidade do aluno, seus gostos e cultura.
- Sobre o Core Inc.
 1. Quais foram suas primeiras impressões utilizando o Core Inc.
 2. Quais foram suas impressões da usabilidade do Core Inc.
 3. Quais foram suas impressões sobre o tutorial do Core Inc. Como você avalia a temática do Core Inc. e o modo que ela busca representações através de conhecimentos do dia a dia prévios dos alunos.
 4. Como você avalia a representação visual das animações do Core Inc.
 5. Qual a sua opinião sobre a abstração que o Core Inc. faz dos processos de execução de um algoritmo em suas animações.
 6. Após usar o Core Inc., qual seria sua opinião sobre ele ser utilizado em sala de aula?
 7. Caso o Core Inc. fosse aplicado dentro do currículo de ensino dos alunos, qual seria a abordagem que você utilizaria.
- Conclusões
 1. Quais modificações ou correções você indicaria para o Core Inc.
 2. Você possui algum apontamento, sugestão ou crítica que acha relevante e nós não abordamos até o momento?
 3. Você estaria disposto a contribuir na validação do processo de análise desta entrevista que vai ser realizado posteriormente?
 4. Você se sentiu desconfortável durante a entrevista em algum momento, tem alguma crítica ou sugestão ao processo?
 5. Agradecimento

APÊNDICE F – TRANSCRIÇÃO ENTREVISTAS EXPERIMENTO PRINCIPAL

Abaixo é demonstrado na íntegra a transcrição das entrevistas qualitativas aplicada de acordo com os procedimentos do Experimento Principal, onde o entrevistador foi o autor desta dissertação (Vinicius Struginski Pereira). Não foram transcrevidos os momentos que antecederam as entrevistas, como a resposta dos questionários no Experimento Complementar e conversas não relevantes ao propósito do mesmo, como, por exemplo, assuntos sobre problemas técnicos ou temas pessoais do professor e entrevistador.

F.1 TRANSCRIÇÃO ENTREVISTA PROFESSOR A

VINICIUS: Ok, eu não preciso essencialmente que você só responda ela, você pode falar mais é algo mais aberto, certo?

PROFESSOR A: Ok.

VINICIUS: Primeiro, eu queria saber para você no seu dia antes-

PROFESSOR A: Acho que eu tenho outra pergunta, qual que é o propósito desta nossa interação, dessa avaliação aqui. Tá, para quem, para que ela vai servir no teu trabalho?

VINICIUS: Bem, ela vai servir para gente validar inicialmente, se o Core Inc. tem na visão dos docentes, na visão dos professores, ela tem possibilidade de ser benéfica em sala de aula para os alunos, certo? Essencialmente, a gente tá fazendo isso primeiramente, porque a gente tá numa pandemia e é mais difícil aplicar com os alunos diretamente, né?

E segundo também porque esse tipo de solução, segundo a literatura e a revisão bibliográfica que foi feita ela tem uma aceitação menor com professores e muitos deles dizem que talvez não funcione, talvez seja muito difícil de utilizar em sala e tudo mais. Então a gente também quer verificar essa barreira, sabe que não são só os alunos o público alvo disso, mas os professores também são, né? Aqueles que decidem utilizar isso.

PROFESSOR A: Entendi, sim, obrigado.

VINICIUS: Tá?

Então, primeiro queria perguntar para você, qual que é qualquer importância relevância para você no seu dia a dia no quando está em sala de aula e tudo mais desses temas que o Core Inc. aborda, da arquitetura de von neumann, do ciclo dele, dessa aprendizagem de algoritmos inicial.

PROFESSOR A: É. Bom, eu considero muito importante. Vamos dizer assim, a ideia de utilizar isso em cursos introdutórios sempre foi defendida por mim aqui. Alguns alguns professores aqui gostam de fazer isso, outros não, né? Mas vamos dizer assim, que eu que inicialmente incentivei isso e quando eu dou disciplinas introdutórias, eu sempre uso essa ideia.

Então considero ela muito importante. Eu acho que a correta compreensão aí desse mecanismo, né? De como é um mecanismo básico. Do ciclo de instrução de como é que as coisas acontecem, por que elas acontecem. Eu acho que ela ajuda bastante o aluno.

Aí falo inclusive da minha experiência como aluno, porque eu aprendi dessa maneira, né? É eu sempre defendi a ideia de usar isso daí como abordagem. Inclusive pela minha experiência positiva como aluno, tanto eu como os meus colegas.

Fixamos logo no início da formação em computação, fixamos algumas ideias fundamentais. Aí através dessa, através disso é que a gente depois na convivência com outros colegas, gente vinda de outras informações, etc via diferença que fazia no sentido de esclarecer as ideias. Certo, eu considero muito importante.

VINICIUS: Então, então você realmente acredita que se o aluno não compreender direito, essa parte arquitetura de von neumann, o ciclo de instrução dele, ele pode ter dificuldades no futuro?

PROFESSOR A: Sim, eu acho que ele vai ter dificuldades e certamente e essa compreensão não vem de algum lugar, não é?

É, a gente tem disciplinas aí que são mais centradas hardware, por exemplo, a gente e outras graduações também tem, não é?

Onde isso também é explicado, não é? Mas eu acho que no ensino de programação, uma ideia ao menos esquemática que depois vai ser mais aprofundada em disciplinas dedicadas a esse tipo de coisa. Uma ideia esquemática, eu acho que sim, faz falta, faz diferença, deixa um buraco aí na formação, se não for corretamente apresentado.

VINICIUS: Certo, e mudando um pouquinho, o que você acha desse tipo de solução? Você falou mais cedo que você tinha utilizado já esse tipo de solução de visualização de software, né? De apresentar de forma visual.

PROFESSOR A: É, então é, talvez eu quando eu respondi o questionário estava é, eu não estava pensando tanto em visualização de softwares, mas eu estava pensando em visualização de algoritmos, certo, né?

Existe uma área, vamos dizer assim, de interesse, aí de animação de algoritmos, algoritmos animados os nomes às vezes variam um pouquinho, mas que se dedica justamente a encontrar maneiras efetivas de apresentar visualmente algoritmos. É especialmente algoritmos que não são triviais.

É, eu acho que o apoio visual por diagramas, por desenhos, por é, enfim, que seja lá o tipo de instrumento que se use, eu acho que ele faz muita diferença. Essa também é minha experiência, tanto como aluno como professor. É, de que você ter esse apoio visual de uma forma ou de outra, facilita o entendimento, complementa o entendimento. Especialmente quando a gente está lidando aí com algoritmos não triviais. Mesmo de uma certa maneira, mesmo em nível mais elementar, aí em disciplinas introdutórias e tudo se se o aluno desconhece mesmo aquilo que é, objetivamente elementar, para o aluno que está sendo apresentado pela primeira vez a ideia, aquilo também pode não ser trivial.

Então, acho que mesmo nesse tipo de situação eu acho que também faz diferença, eu acho que o apoio visual faz diferença.

VINICIUS: E você acha que esse tipo de conteúdo visual, quando ele está sendo apresentado, se ele tenta ser produzido, ser demonstrado para o aluno de forma que interaja mais com a realidade dele, com a cultura dele. Você acha que isso que isso é benéfico também ou você acha que isso não faz tanta diferença?

PROFESSOR A: No contexto de programação ou de algoritmos? Eu não tenho muita certeza de que a quem você se refere.

VINICIUS: Eu posso dar um exemplo. Entre um software que vai, por exemplo, apresentar o funcionamento de um algoritmo, algo parecido. Vamos supor que o software tá demonstrando isso através de formas geométricas e diagramas simples, certo? Tem uma bolinha, tem alguma linha, um gráfico trabalhando. Enquanto o outro software tá trabalhando talvez é transformando essa bolinha em algo que o aluno vê na realidade dele. Talvez seja uma pessoa essa bolinha, talvez ele goste de jogar futebol e seja um jogador de futebol ou algo parecido. Você acha que essa diferenciação de trocar elementos mais abstratos para algo que está mais para a realidade dele você acha que isso pode impactar, que faz diferença para você?

PROFESSOR A: Eu acho que faz e eu acho que em se tratando de ensino superior é preferível a representação mais abstrata. Eu acho que assim representações antropomórficas, representações lúdicas, etc. eu acho que atrapalham em vez de ajudar no processo de aprendizado

quando a gente tá falando de ensino superior. Eu não tenho experiência, não tenho nenhum conhecimento assim, não me atrevo a falar no de ensino médio de ensino fundamental, porque assim, escapa completamente da minha competência, não é? Não tenho nenhuma experiência com essas coisas, não tenho filhos, né? Então é pelo menos no meu tempo de aluno não se ensinava computação nessas idades, não é? Então é zero a minha autoridade para falar fora do ensino superior, não é?

Agora, em se tratando de ensino superior, é, eu acho que não. Eu acho que o objetivo deveria ser desses apoios visuais, desse tipo de ferramenta o objetivo deveria ser deixar o mais limpo possível o conceito que você está querendo apresentar. O mais limpo de qualquer distração.

É, quer dizer um algoritmo é uma coisa abstrata, as relações entre as coisas que acontecem em um algoritmo, as variáveis, etc. são coisas abstratas. E o objetivo do ensino superior é fazer o aluno se acostumar a pensar de maneira abstrata. Você puxar isso pro outro lado, na minha opinião, é trabalhar contra o objetivo, o que deveria ser o objetivo do ensino. A gente tem que facilitar o raciocínio abstrato do aluno, não é? Não tentar substituir ele por outra coisa, como eu vou chamar, mais palatável ou mais próxima do aluno.

De novo, eu insisto eu me refiro aqui a ensino superior, isso é a minha experiência, essa minha percepção aí de como as coisas funcionam ou não funcionam. Eu acredito perfeitamente que se alguém que entende do assunto me disser que para idades mais baixas, é o contrário do que eu estou falando, que é o que é o adequado, eu acredito perfeitamente que possa ser. Mas eu não me atrevo nem a dar uma opinião.

VINICIUS: Você falou que você já teve experiência mesmo como professor utilizando esse tipo de software e você também tá falando pra mim que você acha que a gente não deve abstrair o conteúdo de outra forma para o aluno, a gente tem que apresentar de forma clara mesmo de forma visual, né? Esses outros softwares que você utilizou anteriormente, eu gostaria de saber qual foi a sua experiência com eles e se eles eram desse modo que está falando de maneira mais clara e direta possível, como você disse.

PROFESSOR A: É mais do que software é? Eu vi o resultado do software, né? Eu nunca usei um software que produza animações ou que anime algoritmos né, o que eu vi foi o resultado de animações que eu nem sei se foram produzidas totalmente com software. Talvez elas tenham sido produzidas aí com software em mais alguma coisa não é? É assim. Não seja um processo totalmente automatizado, não é? E foram relativamente poucos exemplos, então nas minhas respostas no questionário, quando eu falo de experiência pregressa com essas coisas, eu estou me referindo a isso. Eu não usei o software, eu só assisti o resultado do que o software produziu, não é?

VINICIUS: A gente também pode chamar isso de um software, sabe? Tipo, você está usando um resultado pronto, mas ele é o produto do software, então a gente trata como se fosse um trabalho de visualização de software, né?

PROFESSOR A: Tá ok? É, então de novo, é o que eu vi foi alguns exemplos de animação de algoritmos. Algoritmos de ordenação, deixa eu ver o que mais assim, alguns algoritmos de busca, coisas assim, coisas que você aprende aí no começo aí da sua formação em computação, não é? Animados acho que mais de uma maneira, alguns deles animados de mais de uma maneira, e eu achei instrutivo. Quando eu vi essas coisas, eu não usei porque eu não estava encarregado de nenhuma disciplina. Eu vi aquilo porque, sei lá, alguém me apresentou, eu achei na internet, enfim, já não lembro mais. Assim é, uma experiência bem limitada.

E tem várias coisas que eu gostaria de ver, né? Como eu disse, algoritmos mais mais sofisticados, coisas que você ensina em disciplinas mais mais adiante, não é? Na graduação, etc. Se eu soubesse de animações que ilustrassem certamente eu usaria, nem que fosse só como

indicação pra que os alunos assistissem, não é? É, mas enfim, foram coisas desse tipo, eu não sei se eu respondi com a pergunta, ou se eu saí meio pela tangente.

VINICIUS: Eu entendi, eu entendi. E me diga sim, quando você abriu o Core Inc. você começou a mexer nos tutoriais, qual foram suas primeiras impressões?

PROFESSOR A: Então eu fiquei com a minha primeira impressão, foi que isso seria dirigido a uma faixa etária menor né? Por isso eu fiz a pergunta. Eu pensei que isso era dirigido a ilustrar, a motivar etc. Não de ensino superior, não é? Acho que mais assim de ensino fundamental, não é? É, então isso foi a primeira coisa, a segunda coisa que eu que me impactou, vamos dizer assim, o fato de a experiência ser muito passiva da parte de quem está usando não é? Você tem um programinha escrito em Pascal. Você tem uma animaçõzinha correndo ali e você fica assistindo aquilo só não é você, é? Por exemplo, pelo que eu entendi não haveria a possibilidade de eu experimentar com valores diferentes lá na entrada, na entrada do algoritmo que está executando.

VINICIUS: Você poderia digitar ali qualquer código dentro dos comandos, aqueles códigos estavam prontos ali, para facilitar a nossa vida aqui. O aluno poderia ali digitar um programa do zero, certo? Botar as entradas que você quiser e ele vai animar.

PROFESSOR A: Tá, entendi, entendi. Mas quer dizer, depois que você é apertou aquele botãozinho de executar a coisa é muito passiva. Você só assiste a animação. É como os comandos, ainda mais uma linguagem de alto nível, o conjunto de comandos é relativamente limitado a partir de um certo ponto, você já entendeu, você já sabe o que esperar da animação. Eu já sei o que que vai acontecer na animação, não é? Depois de 2 exemplos, não é?

É agora ela vai pegar o valor e colocar na variável, não é? Também já sei o que vai acontecer. Agora vai somar 2 números, eu também já sei o que vai acontecer na animação. Rapidamente, eu acho que é o usuário entende o mecanismo básico ali, do que está acontecendo, não é? Quer dizer, qual que é a tradução que está acontecendo no software para para os personagens ali que estão se mexendo e os percursos e as ações que eles fazem rapidamente onde você compreende, não é? E nesse momento aí que você compreende isso daí, eu tenho a impressão que, do ponto de vista de um aluno a coisa fica desinteressante assim, não tem um desafio, não tem um engajamento ali, não tem, quer dizer, ele não tem por exemplo, não é como estamos falando de uma animação. Eu não sei qual é o nome que se dá isso, assim antropomorfizada porque tem bonequinhos e tal. Ele não tem sequer o estímulo de falar. "E se eu fizer isso aqui, será que eu consigo fazer" ele programar animação por meio do código que está escrito, entendeu? Fazer acontecerem coisas diferentes na animação experimentando com o código. Nem sequer isso ele tem a expectativa de que vai acontecer porque não vai acontecer. O bonequinho sempre corre pela esquerda ou sempre corre pela direita. É enfim, a coisa rapidamente se torna previsível demais. Que eu temo, não é que no uso ela se torne desinteressante, quer dizer, e quando eu falo que ela se torna desinteressante assim ela vai fazer pouca diferença naquilo que é o objetivo que é a compreensão mais fácil, mais rápida, mais eficiente aí dos conceitos fundamentais.

E a terceira coisa não é? Também me chamou a atenção. É que a animação para mim trabalha contra o objetivo de uma coisa dessas. A animação em vez de expor, ela esconde conceitos na minha opinião, não é? Quer dizer, a variável é uma caixinha onde eu não vejo o que está escrito nela? E só se eu parar. A animação para ver, não é? A variável ela não tem valor num determinado momento não é assim. Quando você vai ver isso está conceitualmente furado, não é? Quer dizer, não existe posição de memória sem valor. Eu acho que se o objetivo é refletir os conceitos reais e esse tipo de atenção tem que tomar, tem que tem esse tipo de cuidado, tem que tomar, não é?

A coisa da memória apresentada de um jeito que a memória está fora do cenário. Não é assim. Eu interpreto assim porque aquele bonequinho vermelho lá que recebe as memórias, eu

interpretei aquilo como sendo uma espécie de caixa, uma espécie de barramento, enfim, ele joga o valor lá num negócio e aquele negócio vai para a memória, não é? E é armazenado na memória, mas eu acho que é muito importante nessa visualização você ter a memória lá, não é? Quer dizer, é uma parte assim fundamental, eu acho, do que você pretende expor e desmistificar para o aluno. Ao expor a ele esse tipo de ideia, é que cada posição de memória tem um endereço e através desse endereço que você escreve e recupera, não é? Então, esse conceito está completamente escondido, não é? Ele não é aparente na ação, não é?

E a entrada e a saída estão no mesmo lugar na animação, não é? Isso não corresponde ao mecanismo elementar, não é? A entrada e saída são 2 buraquinhos separados, não é? São independentes. A memória, por sua vez, não é um barramento só, mas na animação elas são 2 coisas separadas, tem um por onde chega e um outro por onde a memória sai. Então essas coisas me chamaram muito atenção, porque é, eu tinha essa ideia de que o propósito é educacional, no sentido de facilitar a compreensão de conceitos. É desses conceitos aí que são os conceitos assim protagonistas aí quando você fala, quando você quer explicar a arquitetura von neumann pelo menos esquematicamente, não é? É, e esses conceitos que no meu entender são fundamentais eles não são apresentados com clareza, com fidelidade não é? Então, isso pra mim mata o propósito, o objetivo que você tem quando você quer falar dessas coisas, não é?

Então é essas 3 coisas assim chamaram a atenção mais fundamentalmente, mais de cara.

VINICIUS: Entendo seu ponto que você mostrou que existem problemas fundamentais na representação, ali de como coisas são dispostas na tela e tudo mais, né? E, por outro lado, quando você olha essa temática antropomorfizada dos personagens e tudo, essa tentativa de trazer para o real entre aspas, do aluno, com bonequinhos, uma fábrica, coisas que ele está acostumado antes de entrar no curso normalmente, né? É conhecimento do dia a dia dele, não é? Você sente que isso está bem conseguido, está bem feito ali? E você sente que isso faz alguma diferença para os alunos, para você ou não?

PROFESSOR A: Veja só. De novo, estou falando aí da perspectiva de ensino de ensino superior, eu sempre faço questão de dizer isso, porque eu entendo que as mesmas perguntas, se você tivesse pensando em outra faixa etária e conversando com quem entende do assunto, é, podia ter uma análise, uma discussão muito diferente, não é? Então, nessa altura dos acontecimentos, nessa altura da vida do aluno, nós estamos falando aí de alunos na faixa dos 17 18 19 anos tipicamente não é? Eu realmente acho que representar as coisas com personagens, com bonequinhos, com animais, ou seja lá o que for, não faz muita diferença. Quer dizer não faz muita diferença no seguinte sentido, você não vai capturar a atenção dele, o interesse dele mais ou menos fortemente, não é? É está a maneira como eu percebo isso, você está trocando uma representação simbólica por outra. Tá, tem a representação simbólica do código escrito, que é uma parte do propósito das disciplinas introdutórias aí de programação, algoritmos, etc. é ensinar o aluno a interpretar corretamente aquele código escrito não é? E você está trocando, não é? Está traduzindo aquilo lá por uma outra representação simbólica que são os personagenszinhos, é atuando, interagindo num certo cenário. Mas essa outra representação simbólica na minha apreciação dessa pequena experiência que eu fiz ela talvez esteja um pouquinho mais perto da realidade concreta de como é o funcionamento do computador, mas ainda está longe. Ela ainda é mais uma alegoria do que uma explicação ou do que uma ilustração. Eu acho que uma representação visual ela ajuda, ela é importante, ela ajuda na medida em que ela me a vencer essa distância. Ela me dá um ponto intermediário aí entre o abstrato e o concreto. Se você me dá uma representação visual que ainda está muito longe do conceito, a ajuda que ela vai me dar é muito limitada. Então com relação a primeira pergunta a minha apreciação é essa. Quer dizer, ela está mais para uma alegoria da arquitetura interna do computador do que para uma ilustração, para uma representação. E eu acho que para o aluno do ensino superior, especialmente, que está

tendo o seu primeiro contato aí com o assunto e tudo mais eu acho que é muito mais útil, mais produtivo ele ter uma ilustração do que simplesmente uma outra representação simbólica não é? Quer dizer, tentando dizer a mesma coisa de outra maneira, você sabe o que são fluxogramas? Quando você estudou isso ainda existia?

VINICIUS: Sim, sim, sim, sim, espera aí, tá.

PROFESSOR A: Então o fluxograma eu não uso, nunca usei.

VINICIUS: Respondendo sua pergunta, a gente ainda usa isso na graduação.

PROFESSOR A: Tá, entendi, OK, então tá, então você conhece fluxogramas, então a um tempo atrás, um tempo que para você é muito e para mim é só um tempo se discutia o uso de fluxogramas e se isso ajuda no aprendizado, se não ajuda, etc. As vezes se discutia de maneira interminável e tudo mais, não é? E às vezes a discussão era um pouco fora dos trilhos porque as pessoas perdiam de vista que o fluxograma é outra representação simbólica com apelo mais visual etc. da linguagem de programação. Você tem um outro conjunto de convenções simbólicas para representar o algoritmo, que é baseado lá em em figuras geométricas com texto escrito dentro de cada uma delas e as linhas que ligam ela. Muito bem, mas é outra representação simbólica, aquilo não te traz mais para perto ou mais para longe de como é que é que o computador funciona de verdade, né? O objetivo daquele troço lá, alguns acreditam que aquilo facilita o teu aprendizado, outros acreditavam que não facilita, não é? Nesse momento, tanto faz a minha opinião, não é disso que se trata, não é?

Eu só estou usando como um análogo, quer dizer, é uma representação simbólica proposta para ajudar, explicar como é que funciona o computador. O propósito ali é tornar mais mastigável, por assim dizer, os conceitos básicos da programação, noção de execução condicional, de loop, teste de condição variável, esses conceitos que fazem parte da programação, não é? Um defeito que tem os fluxogramas é o seguinte, se você só consegue representar problemas muito pequenos, se o seu programa cresce o fluxograma fica inviável porque dá muito trabalho de desenhar e dá muito trabalho de compreender, não é? Se você pegar um fluxograma lá que tem, sei lá, centenas de elementos, você lê aquilo e tenta entender o que aquilo significa. É discutível se é mais difícil ou mais fácil do que ler o código que correspondente ao fluxograma desse tamanho, não é?

Enfim, aí você tinha feito uma segunda pergunta que era...

VINICIUS: Bem, eu tinha perguntado para você... Você respondeu praticamente no total.

PROFESSOR A: Já não é, tá? Então tá.

VINICIUS: Eu tinha perguntado para você justamente se você sentia que essa abstração ela ia auxiliar ou não? E a segunda pergunta se essa abstração está bem conseguida, bem feita ao ponto de se aproximar da realidade mais do aluno.

PROFESSOR A: É, eu, é como eu disse para você, pensando em alunos de 17, 18, 19 anos não me parece que estava particularmente próxima, nem particularmente longe. Quer dizer, ele tá acostumado a ver representações assim desse tipo, não é no dia a dia dele. Isso é, comunicação visual hoje em dia está ficando cada vez mais gráfica, não é? Então ícones, avatares, tal fazem parte do dia a dia, não é? Nesse sentido tá próximo. Como que eu vou dizer, quando eu ouço falar nesse negócio de estar próximo da experiência do aluno...

VINICIUS: Do dia a dia do aluno.

PROFESSOR A: Eu normalmente sou levado a pensar em coisas menos superficiais. Vamos dizer assim para mim quando se fala em estar próximo do dia a dia do aluno, tem a ver com o tipo de questão de vivência que ele tem, no seu dia a dia ou na sua experiência, etc. não é? Então tem a ver com você estar falando de coisas mais ou menos compreensíveis nesse sentido, não é? Isso daí, a maneira como que as coisas estão representadas ali, o uso dos personagens e

tal, para mim fazem parte do daquilo que hoje em dia é a realidade de todo o mundo que chega ao ensino superior. Então está próximo no sentido que o alfabeto está próximo, a eletricidade está próxima. São coisas que fazem parte da experiência de todo mundo. Ninguém vai ter nenhuma dificuldade em reconhecer o que está acontecendo ali.

VINICIUS: É, você falou muito, você citou os fluxogramas, fez uma analogia ali com o próprio Core Inc., que de como se sente que a gente só está abstraindo de formas diferentes, mas tá ficando a mesma distância. Por outro lado, você acha que por mais que seja a mesma distância, você acha que existir o fluxograma, existir o Core Inc., ele pode ser um benefício por si só apresentando isso de uma maneira diferente de uma abstração diferente, tipo que o aluno tenha opções entende? Que ele não tem só uma linha.

PROFESSOR A: Eu acho que ter opções é bom, nem todo mundo aprende do mesmo jeito, não é? Então, para a grande maioria das pessoas ter mais de um jeito de compreender facilita a compreensão, não é? Então eu acho que ter opções sempre é bom, não é? É, eu não acho que do jeito que está, do jeito que eu experimentei aí o Core Inc., eu não acho que ele faça diferença significativa no processo de aprendizado. Assim, o que não ficou claro para o aluno, em boa parte por causa daquela coisa que eu falei que eu acho que tem conceitos fundamentais que não estão adequadamente representados é, ele continua não sabendo com o uso da animação. Ela não complementa. Eu acho que significativamente não terá alguma diferença. Assim, alguma diferença vai fazer, não é que vai fazer zero de diferença, o ponto aqui para mim na minha percepção é que será pequena? Essa diferença não será significativa. Então, um outro aluno vai por causa disso, vai realmente capturar um conceito com mais clareza ou rapidez, etc, do que 100, não é? Mas é, acho que é isso.

VINICIUS: Quando você vê os bonequinhos se movendo dentro do software e tudo mais, é deixando um pouco de lado a parte que você considera que ele não está está conceitualmente acurado, tudo mais. Quando você vê se os bonequinhos se movendo, fazendo as ações, você acha que a animação dos bonequinhos em si ela está clara? O jeito que eles se movem, para onde eles vão é compreensível? Para onde eles vão, o que eles estão fazendo?

PROFESSOR A: Sim, eu acho que sim. Eu acho que sim. Eu não sei se eu sou, se essa pergunta feita para mim, não é, professor, se ela serve para o que você quer, não é para avaliar esse aspecto da coisa, porque vamos dizer assim, é, eu sei o significado da coisa, do símbolo que você está me mostrando, porque eu já sei antes, então eu acompanho a movimentação ali, não é? Eu li o código do programa, tinha 3 linhas, eu acompanho a representação, eu leio o texto que aparece no começo. Com a minha experiência, é muito imediato associar uma coisa à outra não é? Quer dizer, compreender o que aquela animação está representando, não é? Eu acho que talvez a resposta seja diferente para o público ao qual isso daí está dirigido.

VINICIUS: Você tem a impressão então que talvez se isso for apresentado para um aluno, ele não vai ser claro o jeito do modo que é pra você?

PROFESSOR A: Eu não chego a dizer que não vai ser claro, eu só vou dizer que, assim como eu já sou muito velho, vamos dizer assim, eu já estou muito acostumado com esses conceitos, eu penso nesses conceitos, eu ensino esses conceitos, etc, etc. Então, para mim é imediata associação entre uma coisa e outra, não é? A tal ponto que eu não tenho sensibilidade para saber se para o aluno aquilo está tão claro quanto está para mim. Entendeu? É uma questão de falta de sensibilidade por causa da experiência, por causa da profissão, vamos dizer assim, não é para mim. Está fácil para mim. É, está tranquilo. Agora você pega um aluno que nunca aprendeu nada de programação, etc., está no seu primeiro dia de aula ou sei lá no dia de aula em que pelo menos o primeiro dos probleminhas de 3 linhas lá o Hello World. Lá ele consegue ler e entender o código e você mostra animação para ele, vai ser tão imediata a associação? Eu não

tenho sensibilidade para dizer, eu acho que é melhor você fazer essa pergunta para gente mais próxima do público alvo.

VINICIUS: Eu faço essa pergunta e eu acho que é uma pergunta interessante também, porque entendo o senhor dizer que você sente que não tem a sensibilidade, ou você não é o público alvo, mas também é aquela ao mesmo tempo, você está convivendo com o público alvo diariamente, né? Você tá está ensinando a eles, né? E daí essa percepção sua também é importante.

PROFESSOR A: É preciso dizer que faz tempo que eu não dou aula introdutória de computação. É a última turma que eu peguei assim de uma disciplina dessas pode fazer mais de 12 anos. Então, enfim, os alunos com quem eu estou convivendo aí nos últimos 12 anos é, já não estão mais aprendendo a programar, não é? Por isso que a sensibilidade é baixa. Se eu tivesse dando aula recentemente dessas coisas, talvez eu pudesse dar uma, dar uma resposta aí com um pouco mais de valor para você.

VINICIUS: Justo justo é. Vamos supor aqui, vamos ter uma hipótese que o Core Inc. vai ser aplicado em sala de aula como uma avaliação. Algo parecido, né? Você acha que teria alguma abordagem específica que seria mais adequado aplicar ele em algum momento? Você acha que seria importante alguma introdução aos alunos antes?

PROFESSOR A: Eu entendo que o momento de usar ele você não tem muita escolha, não é? Quer dizer, é o momento em que o aluno está sendo apresentado esses conceitos, não é? E isso tem um momento determinado ali no andamento da disciplina, não é? Então acho que o momento está determinado aí, pela maneira como a disciplina está montada, construída, não é. É. Eu não sei, é aí, eu não sei, não sei porque não pensei no assunto.

Cenários de uso que eu imagino, não é? É um cenário completamente offline, no sentido de você dar o link lá para o aluno e fala, faça os exercícios lá, faça o tutorial lá, enfim dar uma lição de casa para ele, para usar. Ou uma versão mais online onde o professor está presente quando eles usam, não é? Não, não sei que cenário seria mas são os 2 que eu consigo imaginar. Não sei se tem algum terceiro que você tem em mente assim de cara, não é, mas são as únicas 2 diferenças que eu consigo pensar e eu não tenho um palpite a respeito de qual seria mais interessante ou se faria alguma diferença. É por conta desse elemento que falei de que me pareceu passivo demais a interação, talvez eu me atrevesse a dizer que se você der a coisa offline, como uma lição de casa, ele vai ser menos efetivo porque o aluno vai se desinteressar mais rapidamente, não é? Talvez com um instrutor, com o professor, um monitor, assim, alguém ali encaminhando a experiência talvez a efetividade seja um pouco maior, mas é um chute de alguma coisa que estou pensando agora não é?

VINICIUS: Você também falou de passividade, né? Quando a gente está executando o código por si só, não é, talvez só o aluno pausar animação e ver os os bonequinhos e cada um tá fazendo é não é interessante, não é engajante o suficiente, estou correto?

PROFESSOR A: Sim, na minha opinião.

VINICIUS: Você acredita que, como você disse, fazer as animações serem mais variadas, elas se modificarem, isso tornaria a ação menos passiva ou você imaginou alguma coisa que se tornaria menos passivo essa interação entre o software e o aluno?

PROFESSOR A: Para mim, seria menos passivo... Quer dizer, simplesmente modificar animação não, não modifica, não é. Quer dizer, se continua sendo um negócio que você aperta o botão e não ponha a mão naquilo até acabar, por assim dizer, não é? É para mim, é o mesmo grau de passividade, não é? É para mim, por exemplo seria menos passivo se o aluno precisasse, sei lá, apertar um botão para que a próxima coisa acontecesse, para que a próxima linha seja executada ou para que o próximo pedaço de ação acontecesse, não é?

Agora eu vou chutar coisas de novo aqui que eu estou pensando na hora, não é? Mas é assim, não que sejam boas ideias, mas servem para ilustrar. Por exemplo, se você tivesse, botões diferentes para ações diferentes e o aluno tivesse que acertar a próxima ação. Quer dizer o que vai acontecer agora, ela vai levar o bonequinho, vai levar a memória para para a mesa da entrada e saída, então ele tem que apertar o X. E se ele apertar uma outra tecla é um erro. Então quer dizer, é aí já vira um exercício de ver a compreensão dele do que está acontecendo. Ele ia adiantar o que vai acontecer. O que que vai ser o próximo passo aqui? Seriam coisas desse por essa linha, vamos dizer assim, que não tivesse uma resposta única, não é? E o aluno, de alguma forma é, está, vamos dizer assim, interagindo com a animação. Ele colocar diferentes entradas e ele tem que pôr os 2 números e aí apertar o executar. Eu acharia mais interessante se você aperta o executar e na hora que o número vai ser lido aparece lá uma janelinha, alguma coisa assim pra ele digitar o número. Ok, tá bom, isso já é alguma interatividade com a coisa. É que aí ele pode fazer experiência de digitar o que acontece se ele digitar um número negativo, não é?

VINICIUS: Certo? É? Você acha que durante nossa conversa você pensou em mais alguma parte que você acha que não está boa no Core Inc.? Que você ainda não abordou ou alguma modificação que você acharia interessante do que a gente conversou?

PROFESSOR A: Não, acho que não. Acho que assim as coisas que eu achei importantes, vamos dizer assim, elas foram saindo aí. Deve haver detalhes aí que sei lá, não é? Mas são detalhes, não é?

As coisas assim que eu me preocupo mais, a questão do propósito do negócio é esse, ele está na direção de cumprir esse propósito. Acho que não teria mais nada a dizer.

VINICIUS: Essa entrevista, ela vai passar por um processo de revisão, transcrição, a gente vai verificar o que o senhor falou e vai fazer apontamentos, identificar padrões e tudo mais com demais professores, né? Você estaria disposto no futuro a revisar essa análise da sua entrevista, se a análise que a gente fez da sua entrevista vai estar curada, segundo o seu ponto de vista, no futuro?

PROFESSOR A: Pode contar comigo, tá? Eu não faço questão no sentido de que é, vamos dizer assim. A minha opinião foi pedida, ela foi dada, você faz com ela o que você quiser, não é? Não faço questão de assim fiscalizar se você transcreveu a minha opinião fidedignamente, não estou preocupado com isso agora. Mas se isso daí for de alguma forma útil para você no processo aí do trabalho que você está desenvolvendo, sim, tuas ordens.

VINICIUS: É entendo, senhor, não querer fiscalizar mais. Ao mesmo tempo, a gente tem que nos fiscalizar. Daí a gente procura.

PROFESSOR A: Estou as ordens, no que precisar ou se for útil, é só dar um toque e a gente faz com todo o gosto.

VINICIUS: E para finalizar, agora uma pergunta sobre a entrevista em geral, não é sobre o software. Você acha que hoje, tirando os nossos probleminhas técnicos no começo você se sentiu desconfortável? Tem alguma coisa que o senhor não gostou?

PROFESSOR A: Não, não de forma nenhuma, Tudo ok. A conversa foi produtiva, foi bastante objetivo, não é? Quer dizer, você tinha um roteiro aí, um objetivo a desempenhar. Fazer eu ter contato com o software etc., responder ao questionário, depois a entrevista, não é? Tanto o questionário quanto a entrevista as perguntas eram bem objetivas e bem pertinentes eu acho, não é? Tudo ok, não tem absolutamente assim nenhum incômodo não é pelo contrário, a gente fica contente de poder colaborar com o trabalho dos alunos, não é? Seja qual for, sejam quais forem, sempre que a gente pode colaborar de alguma forma, a gente sempre fica contente.

VINICIUS: Fico feliz, então é porque é a primeira vez que a gente tá fazendo isso. Isso não é muito comum, não é? Não sei se o senhor sabe, mas não é muito comum a gente fazer esse tipo de entrevista já na área de TI, né? E daí é um pouco novo, é novo para mim.

PROFESSOR A: Eu já tive. Como é que foi? Mas aí acho que era um trabalho de conclusão de curso. Mais ou menos recentemente. Que para que um aluno me pediu passar por um processo parecido, não é? É, acho que era um aluno do professor Andrei, se não me engano. Mas, enfim, ele tinha também tinha um só assim, um protótipo, não é? Já de um software que tinha a ver com o ensino de algoritmos em grafos. Não é, e ele queria que eu tivesse um primeiro contato, assim, um teste drive assim. Talvez menos bem estruturada do que a sua, não é? Mas o objetivo era mais ou menos esse também, de alguma forma validar e ouvir a opinião etc. quanto a utilidade didática daquilo que ele estava fazendo, né?

Mas sim, é, eu acho que eu só tenho essas 2 experiências desse tipo de processo assim, por isso que eu perguntei no começo qual que era o propósito para tentar colaborar.

VINICIUS: Foi uma ótima experiência, se eu prosseguir assim com as demais entrevistas, eu vou ter resultados interessantes. Muito obrigado.

F.2 TRANSCRIÇÃO ENTREVISTA PROFESSOR B

VINICIUS: Então, o seguinte, eu tenho algumas perguntas para fazer e tudo mais, mas elas são, no geral, abertas. Você pode se sentir confortável para adicionar, falar o que você acha importante que tudo tem, não só fazer só um bate volta, entendeu?

É, primeiro, eu queria saber de você qual que é a relevância para você do modelo de von neumann, do ciclo da instrução para quando você está trabalhando dando aula na sua pesquisa, se você está envolvido com isso bastante ou não, como é que é?

PROFESSOR B: Totalmente, não é totalmente porque em arquitetura de computadores é essencial saber o ciclo básico. E inclusive, o visualizador eu acho que ele é muito mais útil para a arquitetura do que para algoritmos, porque o pensamento algorítmico eu não vejo assim ele tão dependente daqueles detalhes arquiteturais do barramento de memória e tudo, mas para arquitetura sim, totalmente. Então, para projetos digitais ou arquitetura seria muito interessante.

VINICIUS: Certo, e o que que você acha, qual que é a percepção sua sobre esse tipo de software que você testou, esse tipo de software de algoritmos animados, de visualização de software mesmo, de tentar representar as coisas de uma maneira mais visual? Qual a sua opinião sobre isso?

PROFESSOR B: Eu adoro isso, na verdade, porque assim tem até um aluno meu que está tentando fazer alguma coisinha nisso. Para a gente visualizar o que está acontecendo dentro de um simulador de um processador. Então a ideia é uma visualização funcional do que está acontecendo para ajudar a debugar, por exemplo. Então, debugar o comportamento do computador, então assim, eu já vi algumas abordagens, uma delas até de realidade aumentada, que você pegava o livro de arquitetura de computadores e ele tinha ali a parte de busca de instruções ou a parte de execução, você colocava o celular e ele animava as instruções, acontecendo as coisas. Então era bem interessante, então eu sou totalmente favorável, assim eu acho que é uma ferramenta interessante.

VINICIUS: E você sente que esse tipo de animação seja em realidade aumentada, como você falou, ou como Core Inc. está funcionando, isso tem um impacto, em geral, para os alunos, para o desempenho deles em sala de aula.

PROFESSOR B: Depende, depende. É, depende se os conceitos representados na animação estão bem representados, né? E se essa visualização vai trazer mais entendimento ou dúvidas. Sabe, então assim se é uma simplificação da realidade, eu acho que funciona super não é? E depende também do professor adotar, né? Se o professor adotar e sugerir para os alunos não é, porque hoje em dia existem várias dessas ferramentas, então eu acho que depende desses fatores assim, né? Porque às vezes o aluno fica perdido no meio de tanta coisa, do que ele vê,

aula do YouTube, tem não sei o que, tem visualizador, tem várias coisas, né? Então, depende de um pouco de instrução assim por parte de quem está lecionando.

VINICIUS: Então você acha que esse tipo de ferramenta, por si só ela talvez não seja benéfica, mas é importante que tenha uma combinação dessa ferramenta com alguma metodologia que o professor vai utilizar.

PROFESSOR B: Não, não é bem por aí. Eu acho só que ela é benéfica, diferente do que você afirmou, eu acho que ela é benéfica, com certeza ela por si só ela não traz o conhecimento. Ela precisa de uma base por trás, isso sim. É, mas eu acho que ela pode ser benéfica se ela tiver uma representação de fácil entendimento e correta do ponto de vista do que se quer passar, né?

VINICIUS: E o que você acha desse tipo de solução quando ela está tentando adaptar esses conteúdos mais para a realidade do aluno. O Core Inc. está tentando adaptar isso, antropomorfizando os personagens, trazendo um visual entre aspas mais chamativo, né? Mais atual e tudo.

PROFESSOR B: Isso é sensacional, né? Porque eu acho que é um grande chamariz para os alunos, né? Quase que uma gamificação do conhecimento, né? Até hoje ferramentas desse tipo são muito boas.

VINICIUS: E você acha que esse tipo de ferramenta que está tentando pegar mais da cultura do aluno do dia a dia dele para adaptar o conteúdo, isso pode ter um impacto positivo no aprendizado dele. Pode ter chance de ter um impacto mais positivo do que uma aula expositiva?

PROFESSOR B: Com certeza, com certeza, sem dúvida.

VINICIUS: Tá, então fala pra mim quais foram as suas primeiras impressões assim, quando você estava utilizando o software, o Core Inc.?

PROFESSOR B: As primeiras impressões foi, eu achei interessante poder ter um ar retrô, não é? De de games mais retros assim, eu acho que isso é bacana. E eu fiquei um pouco perdido ali na organização dos componentes da animação. Então eu entenderia que a animação ela precisaria de uma melhor organização dos componentes. Para realmente facilitar o entendimento do aluno. E trazer uma uma visão mais próxima da execução real talvez, porque se você pensar que que você está tratando do modelo de von neumann, ele é aplicado nos processadores.

Então, se você pensar em um processador bem bem simples, não é? E o fluxo dos dados, fluxo de controle que acontece ali, eu acho que ele poderia ser mais bem adaptado no cenário das animações, não é um jogo. É para que ele traga um entendimento mais sólido para o aluno, com menos chances de ter um mal entendimento, sabe? É, e uma adoção maior até, né? Porque aí ele vai ser algo mais... você pode usar o como um objeto de transição, né? Entre várias disciplinas, porque o cara de algoritmos ele pode ensinar sobre algoritmos, o cara de arquitetura pode explicar sobre a arquitetura usando a mesma coisa. Então esse objeto poderia servir como essa ponte, né? De troca de conhecimento entre arquitetura e algoritmos e talvez compiladores mais pra frente, quem sabe?

VINICIUS: Quando você fala que a representação, ela poderia ser mais acurado e tudo mais, né? Você consegue, elaborar mais isso? Se você está falando da posição visual das coisas, literalmente dos desenhos.

PROFESSOR B: O desenho. Não, a posição visual e a organização dos componentes, eu colocaria de outra forma que seria mais benéfico. Por exemplo, a entrada e saída ali está totalmente desconectado de memória, e não é o que acontece não é? A entrada e saída acontece pela memória. Tudo bem que no modelo von neumann, a entrada e saída acontece pelo barramento, mas ali também não acontece isso, né? Não acontece a entrada de saída pelo barramento como o modelo von neumann previa. Então ele tá um modelo que não é nenhum modelo de von neumann e nem um modelo um pouco mais aplicado do modelo de von neumann, né? Um modelo que se vê em uma arquitetura de processador simplificada, sabe? Não estamos falando aqui de um

processador muito complexo. Outra coisa seria o controle, né? Você ter o controle feito por um único personagem, eu acho que isso talvez não seja tão benéfico porque, afinal de contas, os controles são especializados, né? Então se poderia ter mais do que um personagem ou metade da tela sendo feito por um personagem, metade sendo feito por outro, para dar essa impressão de especificidade não é, de o controle está acontecendo. Olha, você tem uma parte que gerencia, só busca de instrução, outra parte só faz instrução e vai para outro lugar, outra parte que isso outra parte que aquilo. Então eu acho que eu, eu não sei assim, do ponto de vista de algoritmos e do modelo de von neumann, quando ele é ensinado, mas do ponto de vista de arquitetura de computadores alguns ajustes poderiam trazer ele muito, muito próximo para realidade de uma arquitetura simplificada, sabe? Um processador genérico qualquer e aí eu acho que teria um potencial maior ainda, sabe, de impacto da ferramenta. E eu entendo também de entendimento do aluno também. Porque nota que se lá em algoritmo você ensina um com um conceito próximo de arquitetura, mas você ensina ele de uma forma não totalmente correta, esse é o primeiro contato que o aluno tem com aquele conceito, né? Então, para depois que eu ou algum professor corrigir esse conceito para um conceito mais correto, ele sempre tem uma dificuldade maior do que se eu tivesse ensinado diretamente o conceito correto na primeira vez. Então, nesse sentido é buscar essa harmonia entre o modelo de von neumann e arquitetura simplificada. Eu acho que seria muito, muito benéfico, porque se não você pode estar fazendo um favor para o aluno em um primeiro momento, mas você pode fazer estar fazendo um desfavor para o aluno no segundo momento. Entende? Então assim, eu tenho esse ponto de vista assim, muitas coisas estão muito, muito bacanas, assim você fez o grosso, sabe, mas a eu acho que faltava agora só pensar um pouquinho mais, porque eu entendo que assim animação, o grosso você já fez, sabe se você fez o trabalho pesado agora, poderia ser feito um ajuste mais fino que aí ele ia ficar tipo, uau.

VINICIUS: Ok. Entendo.

PROFESSOR B: Assim de professor para aluno, não é? Não entenda assim, ah, pô, então tudo que eu fiz está errado? Não, não, não é nada disso que eu tô querendo dizer e passar, né? O que eu estou querendo dizer é você fez um baita de um trabalho, né? Pelas horas trabalhadas, dá pra dá pra se ter uma noção disso, né? E eu sei que não é fácil fazer isso tudo, mas eu acho que uma conversa mais abrangente com outros professores de outras áreas, por exemplo, conversar, sei lá, com Daniel Oliveira, de conversar comigo, com o Bona talvez, poderia trazer uma visão mais diversa sobre o que você está tentando mostrar e talvez você poderia ter as vezes simplificações da realidade, ou aperfeiçoar o teu cenário para que o aluno depois quando a gente mostrasse uma memória de instituições, é o PC, buscando instruções e tal, ele conecte mais rapidamente sabe esses conceitos.

VINICIUS: Usamos apontamentos de conversar com professores de outras áreas diferentes é um pouco que a gente está tentando fazer, não é? Assim, é difícil conversar com todo mundo, que todo mundo é muito ocupado, mas a gente está tentando justamente onversando com você e mais um professor da área de arquitetura e a gente está pegando outros 2 professores que não são da área de arquitetura, para ter uma noção diferente, né? A gente está tentando alcançar isso.

Então você acha assim que é interessante abstração que está sendo feita mas talvez ela se perdeu um pouco no meio do caminho.

PROFESSOR B: É uma forma de dizer. As vezes é uma falta de um passo atrás, de eu olhar, tá, vamos, vamos pegar esse negócio aqui, vamos ignorar o que foi feito e vamos pensar como que acontece na vida real? E aí você vai começar a ver assim você vai começar a adaptar, não é do tipo assim, é isso que acontece na vida real está acontecendo ali quando o bonequinho vai de lá para cá. E aí eventualmente eu imagino o que vai acontecer assim que você em algumas partes essa abstração aqui eu não estou conseguindo representar.

Então, assim eu consigo algumas partes num nível muito detalhado e outras eles estão se perdendo. Então, o ideal que se todo o nível de abstração estivesse no mesmo nível e tudo o que acontece realmente tivesse realmente representado. E outras coisas talvez uma pequena mudança de organização já supre sabe, não são mudanças grandes são mudanças pequenas, né?

Então eu acho que que algumas conversas, assim. Então tá, vamos como que funciona numa um processador ultra simples, e funciona assim, assim, assado, sabe desenhar no quadro branco o pacote vem aqui, vai acontece isso, depois, aquilo, aquilo aí você vai falar um olhe, dá para adaptar isso daqui, de um jeito ou de outro e tal. Então nesse sentido, não é que eu que eu imagino que daria para para aperfeiçoar.

VINICIUS: É fala para mim, você disse que você achou interessante esse visual retrô e tudo mais, né? Como você acha que a temática que foi escolhida pra pra abstrair isso, para abordar, né? Transformar numa empresa e tem uns bonecos, você acha que que isso conecta de alguma forma com os alunos, com a realidade que eles vivem hoje. Você acha que é interessante para eles esse tipo de temática ou não?

PROFESSOR B: Então é. É pra mim não ficou muito claro, é a temática da empresa, sabe? Me parecia mais uma indústria igual uma indústria nuclear assim que os bichinhos têm que ficar fazendo lá, não pode falhar nunca, né?

VINICIUS: É justo, é uma comparação justa.

PROFESSOR B: Então me pareceu que se você tivesse puxado mais assim para temática sobre o que é uma indústria de execução de instruções né ao invés de uma empresa genérica, sim, eu acho que ia ser mais claro a relação, sabe? Então, essa esse contexto da empresa, eu acho que foi importante, mas eu acho que faltou um pouquinho de foco, sabe, com mais foco você saberia o que escrever dos personagens, não ia ficar tão, tão genérico. Muito se fala assim, isso aqui é uma indústria de execução de instruções estelares. Ah, então o bicho tem um olho só estao por conta disso e até se poderia ter uma história no início, no vídeo não é ter a história, do tipo, olha, então existe uma estrela né, que tem uma execução de instruções e tal. Ela Foi criada pelo não sei o quê, para que tenha uma narrativa em torno disso sabe, uma narrativa mais clara, né? Assim, você foi colocando detalhes, mas em nenhum momento você bateu o martelo, você falou claramente, sabe? Então, no vídeo de tutorial, por exemplo, se não não começou falando, olha, é isso. Então assim, conta uma história, cria uma história, ah existe uma estrela há tantas anos luz da Terra que foi descoberta bababábababá é que inspirou os cientistas a fazer os processadores e babababa e assim, cria uma narrativa mesmo.

O rolê todo do teu vídeo eu achei que podia ter uma musiquinha você sabe que no YouTube tem músicas grátis lá.

VINICIUS: Isso eu posso explicar porque eu decidi não colocar música, né? A gente está pensando que os alunos normalmente vão utilizar isso em sala de aula, né?

PROFESSOR B: Aham.

VINICIUS: Em sala de aula é, no geral, experiência que eu tenho é, obviamente não foi feito uma análise sobre isso, mas experiências que eu tenho aqui, música em sala de aula atrapalha. Porque o áudio em sala de aula, o aluno vai mutar, vai incomodar ele, ele provavelmente não está com fone de ouvido, entendeu? Eu não quis me ater a um áudio para explicar isso, sendo que se fosse um áudio talvez fosse perdido no meio.

PROFESSOR B: Há não, eu concordo contigo, eu concordo contigo. Até mesmo a retrô não encaixa com o áudio de você falando, mas encaixa com uma música de fundo, sabe? Não sei se já viu, deve ter no YouTube a introdução do acho que era Dune.

VINICIUS: É, eu sei qual é.

PROFESSOR B: Mas é muito antigo, sabe?

VINICIUS: Eu sei, é o primeiro RTS.

PROFESSOR B: É que você tinha que construir a base o caramba a quatro, se você lembrar dele ou se pegar, por exemplo outro de construção assim, o command and conquer, os primeiros não tinha, nunca tinha voz, né? Mas é uma musiquinha de 8 bits ali É puro, né? Alguma coisa que desse a entender que você tinha algo que passasse alguma emoção, né? Ou de urgência ou de tranquilidade ou de alegria não é, mas você entende, uma musiquinha. Então eu acho que se isso ia ser bom assim, sabe porque fica aparecendo isso é que tá quebrado, meu meu fone, meu fone não está funcionando, não tem áudio, né? Então você poderia tentar achar alguma bobeirinha assim, converter para 8 bits, sabe? E eu acho que ia ser interessante assim. É algo que eu imagino que não dê tanto trabalho e seja prazeroso de fazer, sabe? Tipo assim, você pegar uma música lá de uma música clássica que você fala, nossa, essa daqui é legal ia encaixar, pegar um convertor para 8 bits que aí vai ficar só uns, né?

Umas músicas, umas coisas bem rústica, combinaria totalmente, né? Com o design todo que você colocou, né?

Outra coisa que eu fiquei me perguntando é se não faria sentido você traduzir o código de Pascal em um código assemblylike sabe? Porque o código assemblylike ele tem sempre 2 operadores e um store. Então? Talvez possa ser um trabalho futuro, sabe, que se possa sugerir. Porque, sei lá, você poderia pegar o código binário que você usou fazer e fazer o dessassembly dele e, mas eu concordo que é uma coisa assim complexa de ser feita, eventualmente, mas poderia ser uma extensão interessante, sabe, talvez você escrever código em assembly.

VINICIUS: Quando você quando você fala da tradução, só para esclarecer, você fala o aluno ainda escrever o código em Pascal e na animação e tudo você fazer essa tradução em assembly ou diretamente você até mesmo deixar o aluno escrever.

PROFESSOR B: Eu acho que as 2 abordagens são válidas. Eu vejo não como uma coisa só, mas eu vejo as duas podendo ocorrer. Porque de verdade, assim, o processador dele e a máquina de von neumann, ele não era com mais de 2 operandos, não é? Então por isso, sabe, então ou você fazer uma simplificação da lógica, não é do que você tiver aí com mais de 2 operandos ele quebrar em mais. Porque o aluno fala, então olha lá quando eu escrevo aquilo então compilador vai quebrar isso em um código de 3 endereços, que é como lá o pessoal de compiladores chama. E aí você poderia, outra sugestão que é paralela a essa você ter uma setinha mostrando com a instrução em execução, sabe? Sim, conforme ser que você mostra ali, não é? Qual que começou a ser buscado e tal, então poderia ter uma setinha do lado do código mostrando. É aqui, ó. É o que isso imagino que é uma super simples de você fazer, não é?

VINICIUS: Sim, sim, sim, não é difícil.

PROFESSOR B: É que facilitaria, porque na hora que você estava executando ali, você viu que o pause não funciona quando aparece a janela com o nome da instrução, né? Depois se você colocar o mouse ele não pausa. Então ali achei que às vezes quando eu tava entendendo eu queria olhar mais tempo daqui para entender com a instrução, tudo bem. E aí, o que eu percebia assim se eu tivesse uma instrução do ladinho sendo apontada ou negrito, aí fica mais fácil assim do tipo, "ah pisquei e perdi alguma coisa"né? Sabe, é uma bobeirinha do lado.

VINICIUS: Justo, um dos motivos que eu não implementei isso ainda até, não é? É porque a tese é sobre o que você está vendo do lado direito, não é. É sobre a animação em si, certo?

PROFESSOR B: Sim, sim, sim, sim.

VINICIUS: É como se esse site fosse um arcabouço para isso. Então assim, você consegue utilizar aquele conjunto de animação que você está vendo e independente de todo o resto que você está olhando, tipo, você consegue baixar no github e utilizar então, a gente focou muito em funcionalidades por enquanto dentro daquele quadrado, sabe qual é? Porque o que está

fora não necessariamente vai se aplicar. Se outra pessoa botar em outro lugar, porque ela vai usar talvez outro editor de texto do lado.

PROFESSOR B: É tipo uma APIzinha que você fez?

VINICIUS: Isso! Exatamente! Tanto que a ideia é assim, né? Hoje a gente faz com o Pascal, mas a animação em si, ela serve para absolutamente qualquer linguagem, desde que tenha a entrada correta tá valendo. É bacana, é um apontamento interessante, tipo esse destaque e tudo mais.

PROFESSOR B: É um ponto que eu fiquei pensando assim, tudo bem, parece que são 2 interfaces, é uma bem nova, que é onde você escreve o texto e outra com arte antiga, que é onde você realmente tem animação acontecendo. Se você for ver, tem uma boa parte da tela que está inutilizada ali que você fala em qualquer momento você pode pedir ajuda tal.

VINICIUS: A parte está falando mais a direita.

PROFESSOR B: Isto, mais a direita é o textinho dos personagens, você poderia talvez quando a pessoa não quer ajuda e tal, ali você poderia utilizar com carga útil, sabe? Poderia, por exemplo, a mostra o que você quer mostrar e depois aquilo fica fixado ali na direita. Então, a animação está acontecendo, mas eu sei a que se refere aquela animação. Alguma coisa assim, eu fiquei pensando tá? Tem que tentar achar uma solução, não é? Então eu, eu iria nesse ponto assim, porque se você for ver o restante da tela toda é muito útil, mas ali fica parado, né? Então você poderia, tipo, fica parado por um tempo e diminui falando, ó, preciso de ajuda, aí você passa mouse ele aparece alguma coisa de ajuda, mas usar com carga útil mesmo a tela.

VINICIUS: É, me fala também o que que você achou da... a gente está falando de interface e tudo não é, mas fala especificamente o que que você achou da usabilidade do software? Se o tutorial você fez até um apontamento no começo, se o tutorial para você explicou o suficiente para você saber usar o Core Inc.?

PROFESSOR B: Sim, isso ajudou. Eu tenho um apontamento que seria assim, o tutorial não apresentou o cenário. É ele, precisaria apresentar o cenário né, aqui é a entrada e saída, aqui é onde acontece tal, coisa, etc. Sabe, então isso poderia ajudar ainda mais, mas ele foi fácil. É, mas eu acho que isso cairia fichas mais rápido.

VINICIUS: Ok, ok, e você acha que mesmo um aluno utilizando que ainda não está tão familiar com esse tipo de interface, está aprendendo.

PROFESSOR B: Justamente nesse ponto que a explicação do cenário seria mais importante ainda.

VINICIUS: Ok. É, vamos ter uma situação hipotética aqui que o Core Inc. que ele fosse ser aplicado em sala de aula, certo? Você imaginaria alguma abordagem específica que você abordaria em algum momento? Pode ser até que você comentou que talvez ele fosse interessante em arquitetura, né? Algo que até a gente não estava cogitando antes. Você pode explicar isso melhor? Onde você acha que você aplicaria? Como você aplicaria?

PROFESSOR B: A simulação do mips, por exemplo, a simulação do mips eu acho que seria totalmente passível assim de ser mostrado o que está rolando, sabe? Então você teria que ter um cenário um pouquinho diferente, verdade né? Com memória de instrução, memória de dados, mas o mips é sem pipeline, sem nada. Eu acho que ele seria uma adaptação não difícil de ser feita, eu imagino, né, até mesmo porque você falou de API. Ia ser muito legal, porque aí eu poderia escrever um assembly do mips, não é? Porque isso eu acho que é um ponto, assim que cada vez mais eu vejo que os alunos têm uma dificuldade. E como professor, eu tenho uma dificuldade também porque eu tenho que explicar, eu tenho que escrever umas 5 instruções lá no quadro, aí eu desenho mips e aí eu vou executando cada uma no braço. Agora o PC aconteceu isso aí. O PC veio para cá, aí veio para cá e veio para cá. Bom, aí eu apago tudo. Agora a próxima instrução começou aí, está aqui, acontece isso, acontece aquilo, aconteceu, sabe? Então

ia ser muito legal assim se o aluno tipo pô, não sei, eu dormi, não estava bem atento, não sei o quê. Eu ia ficar um pouco mais tranquilo, né? De não precisar certificar várias vezes que o aluno viu a explicação e tudo mais, né? E você pode imaginar que é uma explicação que demora bastante tempo assim, consome bastante da aula. Você desenhar desenhar todo um caso de teste. Eu não consigo colocar casos mais ricos, por exemplo, um laço de repetição e nada disso assim. Que talvez o aluno me olhando, assistindo, os alunos que tem um pouco mais de dificuldade talvez ali cairia as fichas para ele, sabe? Porque é o que eu vejo acontecendo, assim, quando o aluno vai lá na minha sala, eu não entendi muito bem isso aí. Eu falo, tá, vamos ver aqui, ó, aí eu começo a fazer passo a passo ali no quadro, e o cara "ah agora entendi". Sabe, então acho que seria uma ferramenta muito boa e principalmente para a disciplina de projetos digitais, porque em projetos digitais tem uma conexão ali que não é uma conexão trivial, que é quando eu cara sai do circuito que ele estava fazendo cada circuito separado, ele sabe fazer uma memória, ele sabe fazer um registrador, ele sabe fazer uma ula, ele sabe fazer um mux, um demux. E aí agora a gente fala assim, cara, agora nós vamos orquestrar uma execução de uma instrução usando tudo isso. Ah, daí os caras batem a cabeça na parede, é sangue ranger de dentes, sabe? Então talvez uma ferramenta dessa seria muito útil assim não é porque isso daqui é o que vai inspirar a nossa execução. Né? E aí o aluno entendeu a visão geral, não é? E aí os componentes eles já sabem mais ou menos os componentes, não é? E tudo bem ter abstrações, não é igual você usar analogias, né? Então, eu acho que que faz muito sentido.

VINICIUS: Daí me diz assim, você acha que o que tem hoje, tirando os apontamentos que você botou as animações e tudo mais, você acha que essas animações, que o bonequinho está indo fazer e tudo ele andando, é claro, assim é facilmente identificável pro aluno?

PROFESSOR B: O que o bonequinho está indo fazer e tudo mais não é por ser poucos pixels, né? São poucos pixels andando. Eu acho que sim, acho que está claro. Teria que focar assim, que se eu pensasse em fazer um refinamento. Tipo assim, nossa, vamos tentar ver se tem um refinamento que a gente consegue fazer, aí eu pegaria cada ação específica daria um zoom nela e tentaria identificar o que está acontecendo ali. Mas aí seria, sabe é um nível de detalhamento assim que a nível de pixel assim, né, ficaria melhor ali, igual o olho do bicho, né? O bicho só tem um olho, não é isso ou de baixo ali.

VINICIUS: Ele tem um LED no rosto, né?

PROFESSOR B: É então, mas não acho que no vídeo parecia que ele só tinha um olho e aí embaixo parecia que ele tinha um LED, então ali parece que se cortasse 2 pixel parecido com a animação, sabe aqui que eu pensei assim. Mas eu acho que está bom, sabe? Tá bom.

VINICIUS: É, além dessas, dessas modificações, correções que você comentou até agora. Você acha que tem mais alguma coisa que você queira falar mais? Uma correção ou algo parecido que a gente não discutiu até agora?

PROFESSOR B: Ai eu não sei quais são as limitações do seu do seu modelo, sabe? Nota que como trabalho de mestrado você não tem que resolver o problema do mundo, não é? Mas só estou resolvendo o problema da animação. Não, nem esse problema você tem que resolver tudo, né? Então assim, é importante você deixar claro quais são as limitações em algum lugar, sabe? É quase que eu não fui clicando ali na ajuda tal pra ver se tem um lugar que mostra as limitações ou não. Você poderia trazer explicações. Você fala das limitações do que o software pode fazer, né? Ele, por exemplo, eu consigo fazer chamada recursiva nele funciona?

VINICIUS: Então é isso, se você vai no sobre lá em cima, no menuzinho, a gente tem. A gente não botou isso tutorial porque a gente achou que isso no momento não seria relevante ao aluno, porque provavelmente o professor ele está orientando. Mas a gente tem assim lá que está escrito assim, a gente suporta tais comandos. Em Pascal hoje a gente suporta o programa

atribuição basicamente, a gente vai, a gente vai até o IF, a gente não chega no loop. A gente parou no loop por causa que no começo eles não estão vendo loop, né?

Tá escrito, só não está na primeira página, né.

PROFESSOR B: Tudo bem, mas é importante que esteja escrito, né? E essas questões mais técnicas sabe do tipo, assim para o professor entender as limitações, porque assim se o professor entende a limitação ele vai achar um exemplo que guia o aluno na direção certa, não é? Ou se o aluno fizer uma divisão por zero, ele vai falar, "olha, não é compilador que deu erro, na verdade, é uma execução que gerou uma exceção". Então esse tipo de coisa seria importante você falar, sabe, tipo assim, olha, não espere a gente.. então as construções dentro de Pascal a gente não faz construções de laço, a gente não faz construção de chamada de função, a gente não faz construção disso e daquilo. A gente só suporta as funções de write e read. Então assim, sabe sempre falar, delimitar assim, ó, meu trabalho é esse escopo aqui, então tudo o que está aqui dentro, você pode brincar, se achar um bug, me avisa, e o que está fora é tudo um bug, então não use porque não é feito pra isso, né? Então, nesse sentido, eu acho que é importante sim, se não tem ainda, né? Eu não li a parte do help, como você está falando, mas é importante você falar isso como cientista da computação, sabe como um conhecedor de todo o processo que o código faz.

VINICIUS: Um manual mais voltado para o educador ou os cientistas da computação, né?

PROFESSOR B: Exato, exato. Então eu acho que isso seria interessante assim trazer, porque a gente sempre está numa estrada, assim a cabeça nossa é do tipo eu quero mostrar que eu resolvi o problema e não que eu tenho problemas ainda ser resolvidos. Não é, tipo, eu quero mostrar que resolveu o problema lá, isso acabou, entendeu? Mas de verdade, quando você fala abertamente, olha essa daqui é uma limitação, essa é outra limitação, essa outra limitação, de verdade, você está apoiando os próximos amigos que podem resolver essas partes, né?

O seu orientador, que pode já ter claramente o que precisa ser feito, né? E os educadores, né? Falando assim, olha, então tá, então ele não consegue laço, então tá, então laço é uma coisa que eu não posso usar, tudo bem? É para deixar mais claro assim, igual ali faz tempo que eu não programo em Pascal. Então em C eu diria para você, olha ali é uma divisão por zero, ele não sabe que é uma divisão por zero em C, só na hora de executar que levantaria uma exceção, aí poderia fazer um trap, poderia fazer outra coisa, né?

Então, assim você usar ajustar essas nomenclaturas é importante, sabe?

VINICIUS: Tudo bem, tudo bem.

PROFESSOR B: E deixa todo em português tchê, metade em português, metade inglês. Lá em cima código, lá em baixo input. Por que não é código de entrada? Escolhe uma língua e vai até o fim com aquela língua.

VINICIUS: Tudo bem, tudo bem. Mas você não se incomoda não se incomoda com o nome do software ser Core Inc.?

PROFESSOR B: Eu me incomodo também, eu achei também que ele poderia ter um outro nome. É, é que assim você não chamou a atenção para esse nome. Sabe na tua historinha da narrativa que eu falei, está o rum, não é tipo assim, há existe uma estrela, tem uma indústria de execução chamada Core Inc. Agora estaremos entrando dentro do Core Inc, vamos ver o que esses funcionários misteriosos fazem. Aí você faz lá, entendeu? Você tem que dar uma pirada, os alunos vão falar assim, ah tá meio bobinho, mas aluno adora coisa bobinha assim, né?

Nós adoramos coisas bobinhas assim porque, sabe, a gente não quer uma coisa super, não é? Tem que ter que ter uma liberdade ali porque é uma brincadeira, não é? Aí fica mais leve, né?

VINICIUS: Tudo bem. Me diz assim, a gente está fazendo essa entrevista aqui, não é? Essa entrevista vai passar por uma análise, transcrição e tudo mais dos apontamentos, né?

Eu gostaria de saber... eu não estou exigindo de forma alguma nem nada, mas se você estaria disposto a fazer uma avaliação da análise das entrevistas mais à frente, tipo, verificar se a análise que foi feita, do que a gente conversou, tá acurada.

PROFESSOR B: Eu acho que isso é muito legal. Sim, sim, sim, sim. Eu acho que esse é um cuidado super bacana que você que vocês estão tendo. Sim, se tem essa proposta, sim. A mesma coisa que poderia ser feita uma análise do questionário que você aplicou.

VINICIUS: Então o questionário, ele é baseado na literatura, né?

PROFESSOR B: Tem algumas partes que eu achei assim, parece que será que está tentando me induzir alguma coisa? Ou ele não estava sendo preciso suficiente, não é? Então tem uma parte ali que me chamou atenção que ele falou assim é, tal coisa é razoável. É que é razoável. É razoável uma palavra é tipo, quando você vai pro Estados Unidos, alguém fala assim, essa comida é OK, aham, ele tá falando, a comida é razoável. É boa? É o que é ruim, não é?

VINICIUS: OK, aham, entendi.

PROFESSOR B: É esse conceito que você quer passar, né? Então, esses tipo de cuidado assim, né? Para se pensar assim, né, tipo., ficou um pouco meio genérico assim, não é?

VINICIUS: É justo, dá para fazer uma análise também. Só que isso aqui todo mundo já usa faz 5 mil anos.

PROFESSOR B: Tá todo mundo fazendo igual, perfeito, tá dizendo que funciona tudo isso? Se você tomou esse cuidado, tá ótimo, tá? Você tomou esse cuidado, está ótimo. Eu ia até sugerir, então fala com o pessoal de estatística, mas justamente para ter isso, né? Ter um questionário que não influencia a pessoa, né? Isso seria até interessante passar por alguém de estatística, sabe esse questionário? Né? Tem uns parceiros aí do departamento, Wagner, Valmes. Do ponto de vista estatístico, porque em estatística eu acho que você não pode falar assim a estar razoável, sim ou não, sabe? É meio que assim, você tem que falar que você achou disso, aí você dá como opções razoável, bom ou ruim, né? É faz muito tempo que eu fiz estatística descritiva tal, né? Então, então, assim, se não outro cuidado que você poderia tomar também, sabe, tipo assim, mas se você já tomou esse cuidado de eu estou seguindo modelo e tal. Está defendido, sabe? O importante é isso, não é você ter argumento para cada coisa, para cada decisão que você tomou, né?

VINICIUS: Só pra fechar, uma pergunta aqui que não é sobre o Core Inc agora nem nada, mas assim, durante o processo da entrevista hoje, tirando os problemas técnicos claro, você se sentiu desconfortável em algum momento teve alguma coisa que você não gostou? Porque a gente vai aplicar com mais professores, né? Aprimorar o processo é sempre interessante.

PROFESSOR B: Eu tenho alguns pensamentos confusos sobre se precisaria ser síncrono ou não, sabe?

VINICIUS: É isso que a gente está fazendo hoje?

PROFESSOR B: É. É. É A Entrevista. Tudo bem, eu entendo que tem que ser síncrona, mais a será que o teste tudo mais não é? Mas assim você só vai garantir se for síncrono né. Então acho que não, acho que não tem nenhuma ressalva, não é?

VINICIUS: É, existe uma questão de quando eu tô vendo sua tela. Talvez você não tenha notado algo, tenha perdido algo no tutorial, algo parecido que talvez o tutorial realmente não esteja claro. Eu preciso que você faça tais coisas para a entrevista. É um dos motivos, né? E daí o segundo é que, no geral, quando eu estava conversando com o Castilho e tudo mais, era mais fácil tipo assim, o professor vai entrar aqui a gente vai separar um horário e ele já vai ver, vai fazer.

PROFESSOR B: É verdade.

VINICIUS: É daí fica mais fácil, né? E daí justamente a entrevista é impossível que ela não seja síncrona, porque até porque várias perguntas que eu fiz aqui, elas não estão escritas, né? Eu decidi com base no que você fala.

PROFESSOR B: Mas eu achei que foi isso, foi muito bom processo né.

VINICIUS: Então eu queria agradecer você.

PROFESSOR B: Que isso Vinicius, eu que agradeço.

F.3 TRANSCRIÇÃO ENTREVISTA PROFESSORA C

VINICIUS Então é o seguinte, o meu projeto de mestrado ele é um software de algoritmos animados. Eu não sei se você tem noção que que é isso, mas é a ideia dele é para auxiliar no ensino superior, no ensino de programação básica, algoritmos. Então, a ideia dele é que o aluno, quando ele está começando a entender como funciona o computador e quando ele vai começar a programar e tudo, ele tem essa ferramenta de auxílio para tentar ajudar ele, né? Então, o que você vai ver é que essa ferramenta funciona em Pascal, mas você não precisa, não lembra assim sintaxe, não precisa saber de Pascal, ele tem códigos exemplos lá só pra você testar e ela funciona em Pascal e basicamente a ideia é que o aluno bota o código dele lá né? Dado devido às limitações de comandos, são coisas mais simples, que ele tá vendo no comecinho. E esse código, ele vai ser animado, vai ter bonequinhos que vão animar ele de acordo com com a teoria, não é? De acordo com a arquitetura de von neumann. Então tem um bonequinho representando cada área e eles vão eles vão se animar conforme você bota um código lá dentro.

Então a ideia que você testa ferramenta pelo tempo que você achar necessário, quando você achar que você viu tudo, se você tiver alguma dúvida na hora que você está usando, eu tiro as dúvidas para você, para você usar ela. Daí depois disso é eu tenho um questionário no Google forms que você responde e por último eu faço uma entrevista com você, eu tenho uma série de perguntas e tudo daí a gente conversa sobre a ferramenta. Basicamente é isso, alguma dúvida?

PROFESSOR C Não por é por enquanto não.

VINICIUS Então tá, eu vou te mandar o link aqui no chat aqui do negócio.

PROFESSORA C Posso entrar nele, então.

VINICIUS Compartilha a tela para eu ver, é só essa primeira parte do software que eu quero ver.

PROFESSORA C Aonde que é o compartilha a tela? Que esse aqui eu não usei ainda.

VINICIUS Tem o quarto botão ali.

PROFESSORA C Compartilhar sua tela.

VINICIUS Foi?

PROFESSORA C Foi.

[Uso do software]

VINICIUS Está aqui? Top, tá, agora eu tenho a Entrevista para gente fazer. Eu tenho tópicos, perguntas para te fazer tudo, mas elas são bem abertas, mas tá, você pode sentir a vontade para discorrer, falar, se você acha que tem algo para adicionar, não precisa ser super objetivo, né?

PROFESSORA C Certo.

VINICIUS Então, uma coisa que eu queria saber é, qual que é a relevância para você do modelo de von neumann, dessas coisas que o Core Inc., que a solução está tentando abordar, qual é a relevância dela pra você quando você está em sala de aula ou menos em sala de aula que você falou que agora está focando doutorado, né? Quanto isso é importante na sua pesquisa? Você acha que está diretamente conectada, relevante ou está mais fora da sua área?

PROFESSORA C Da minha pesquisa em si, ela está fora, mas do meu trabalho em sala de aula ela está dentro, né? Se a gente for pensar em na arquitetura de von neumann, sempre

quando eu começo a trabalhar a introdução a algoritmos, já eu faço essa analogia com essa arquitetura para que a os alunos consigam compreender o que é uma variável, como que ela se comporta, onde ela é, digamos assim, enviada, né? Que nem você coloca lá no jogo o pacote na memória, de onde eu leio para onde eu envio? Então você ter essa noção dessa arquitetura de von neumann é importante porque facilita a compreensão na programação em si, né? O que você está fazendo.

VINICIUS Tá, e o que que você acha também de desse tipo de solução que o Core Inc. tenta ser, esse tipo de solução mais como uma ferramenta visual, que tenta abstrair de uma forma visual esses conceitos.

PROFESSORA C Eu sou 100% adepta e concordo com esse tipo de solução. É lógico que a gente tem que ter aquela noção de que os alunos aprendem de maneira diferente, né? Uns são mais visuais, outros mais auditivos, né? Outros gostam de escrever e tal, mas na minha opinião, sendo professora da área de programação aí há quase 20 anos. O visual nesse caso é importante porque é um conteúdo muito abstrato, é fora da nossa realidade. Então quando a gente traz um jogo, uma tela dinâmica com desenhos com animação, sai um pouco dessa abstração e faz essa analogia com a realidade. Isso facilita a compreensão do aluno, do conceito em si, né?

Então às vezes eu estou desenhando no quadro, porque quando a gente não está trabalhando com esse tipo de software eu estou fazendo analogias. Quando você teve na aula de estruturas de dados comigo você lembra que eu fazia analogia lá da corrente de portão para a lista encadeada né? Então é sempre interessante você trazer essa parte dinâmica de analogia de animação para essa área que é muito abstrata, porque isso realmente facilita a compreensão.

VINICIUS Tá, e o que você acha de quando esses conteúdos, se vale a pena esses conteúdos tentarem ser mais adaptados para a realidade do aluno, assim, para a cultura dele, o que ele já vê no dia a dia antes mesmo de ele ter tido contato com qualquer tipo de conteúdo, nesse caso específico, com algoritmos e tudo. Você acha que vale a pena você adaptar esse conteúdo para tentar aparecer algo mais real? Tipo como por exemplo, você já deu o exemplo da corrente agora?

PROFESSORA C Eu acho que sim. Eu é posso até agora fazer um gancho com o que eu estou estudando agora no doutorado, que meu doutorado é para eu analisar e investigar como que se dá, como está se dando o processo de inclusão digital em projetos de extensão universitária, e um dos autores que eu estudo agora que trabalha extensão é Paulo Freire e o Paulo Freire fala muito do tema gerador e que que é o tema gerador? Ele trabalha a parte de alfabetização, né? De adultos, principalmente, não só de adultos, mas o foco principal é adultos.

É esse tema gerador, tem a ver com a realidade, né? Teve uma parte de um livro dele que eu tava lendo lá, que ele traz muito o exemplo dos agricultores, então ele traz muito da realidade de termos e palavras que os agricultores estão acostumados para poder trabalhar alfabetização com eles. Não só isso, mas também análise crítica do mundo deles, ali do contexto, né? Então, se a gente consegue desenvolver um software que traga a realidade de um grupo de alunos, uma determinada localidade com os termos que eles estão acostumados a usar, que eles já conhecem, já absorveram esse tema onde é natural o entendimento, fazer uma analogia com a programação, que é o nosso caso, que é muito abstrato. Eu acho que isso vai facilitar ainda mais a compreensão. Pensando na linha de raciocínio de Paulo Freire, né?

VINICIUS E falando do software em específico, que você testou do Core Inc. assim, quando você começou a mexer nele quais foram as suas primeiras impressões assim?

PROFESSORA C Para ser sincero, eu lembrei do Iron Ears. Eu já sei, o autor é o mesmo. Mas é eu acho que eu me senti familiarizada mais do que se fosse o primeiro contato. Porque eu conheci o Iron Ears e eu conheço a lógica de pensamento de vocês. A única coisa que

eu senti ali, eu não sei qual que é o objetivo de vocês nesse sentido, eu senti falta do som. Sabem de alguma narrativa, O Iron Ears também tem som né?

VINICIUS Tem, tem, o Iron Ears tem.

PROFESSORA C Então acho que por isso que eu senti falta, eu senti só essa falta que na hora que eu li o tutorial lá eu entendi, tá o funcionamento da tela é assim, tais são os comandos que tem beleza. Aí quando eu começa a animação, por ser a primeira vez que eu tô vendo a animação, eu fico um pouco perdida, né? Onde que está as coisas? Se tivesse uma narrativa bem curta, não precisava ser longa ali. Nas primeiras vezes que a gente visse já ajuda para a gente se situar dentro da animação.

VINICIUS Você diz um pouco como o vídeo que você viu, a história que você viu no começo se ela aparecesse nos primeiros ciclos da animação?

PROFESSORA C Isso

VINICIUS Tá, entendi. E daí você acha que a própria animação em si os bonequinhos se movendo e tudo seria legal se eles também que fizesse um barulhinho, pegasse caixas e fizesse um barulhinho.

PROFESSORA C Isso não só para o meu entendimento, pra eu que enxergo me situar, mas para quem não enxerga e vai ouvir e vai conseguir também acompanhar, entendeu?

VINICIUS Entendi.

PROFESSORA C Pega um aluno com deficiência visual, se ele se jogar ele, ele vai... Tá certo que acho que ainda não tem, é isso ali no lado onde a gente digita o código onde vai digitando tivesse lá, mas não sei se vai conseguir passar um leitor de tela que Lê pra ele ouvir o que ele está digitando, mas se ele coloca um código lá e ele consegue ouvir a tua animação, ele também vai conseguir entender o objetivo do software, vai conseguir entender animação e consequentemente ele também vai entender o código que está ali sendo animado, entendeu? Eu acho que facilita tanto principalmente para quem é iniciante. Porque às vezes a gente fala da ULA, da memória, do processador, mas aonde que está, até o aluno se situar sabe quando você vai narrando e ele vai acompanhando com a narrativa. Eu acho que vai facilitar o entendimento. E você dá a opção de ligar ou desligar a narrativa, porque quando ele tiver já habituado com o software, tem gente que nem vai querer ouvir, "Já sei como funciona", ele vai lá e desliga, entendeu?

VINICIUS Entendi, entendi. E o que que você achou é do processo de tutorialização do software em geral, da usabilidade.

PROFESSORA C Eu achei bem clean, bem fácil, do meu ponto de vista, tá tudo ali e não é uma tela cheia de botão, monte de coisara, deixa até eu voltar nela aqui.

Como que eu vou dizer? Está tudo muito claro, ela está bem dividida. No meu ponto de vista, onde que estão as informações, né? Em locais bem definidos, se mantém o padrão, você tem o tutorial e ele é claro e explica. São poucos os comandos que você precisa saber para poder navegar nela. Eu achei bem tranquilo. Eu acho que mesmo sem tutorial a gente conseguiria no primeiro momento ia se bater um pouco mas depois ia pegar, né? O tutorial facilita o entendimento inicial, mas a tela ela é tão pra mim ela está tão simples, a usabilidade, ela tá tão simples que vai ter gente que vai nem vai ver o tutorial e vai usar direto.

VINICIUS E o que que você achou da temática em geral da solução. Assim, o que a gente está tentando fazer, que é tentar representar as coisas aproximando do mundo real. A gente faz uma fábrica, tem elementos ali mais reconhecíveis. O que que você achou dessa temática? Como se avalia ela?

PROFESSORA C Eu acho interessante, porque eu penso assim, a gente usar o software dentro do contexto da universidade para o ensino de graduação, eu acho bem interessante. Porque a gente trabalha, por exemplo, se for usar no curso, a gente vai estar atuando em sistemas que

vão estar dentro de fábricas ou vem alunos que a gente sabe dos que trabalham em fábricas, têm a noção, tem a ideia, o conhecimento, a prática, né? A vivência da fábrica em si, então isso já ajuda nessa analogia aí. Mas se eu, por exemplo, eu Professora quisesse usar o software, num projeto de inclusão digital, com pessoas idosas ou sei lá, até mesmo adolescentes que não têm essa vivência de fábrica de contexto, seria interessante eu poder também ter outro contexto para fazer as animações. Sei lá, o contexto da escola, um contexto do supermercado, um contexto, não sei, mas um contexto, mais cotidiano sabe que daí o software poderia ser usado, o teu software poderia ser usado em contextos diferentes, não só na graduação, na universidade.

A extensão universitária é uma atividade da universidade, mas que é aplicada numa comunidade. Então eu poder pegar o contexto do dia a dia eu poderia aplicar de uma forma mais fácil. A adesão seria mais fácil pra essa comunidade que eu fosse atender.

VINICIUS Então você acha que seria interessante, se a gente tivesse vários ambientes diferentes ali naquele no momento que eu abro e eu posso escolher?

PROFESSORA C Isto. É lógico que eu acho que isso vai ser um trabalho futuro para você, né? Não vai ser um trabalho para a tese.

VINICIUS Não seria pra agora.

PROFESSORA C Mas eu acho que seria interessante que dependendo do público alvo que você for trabalhar, quanto mais próximo da realidade deles a gente conseguir fazer analogias, mais fácil vai ficar a compreensão desse conteúdo para essa comunidade, então eu ter temas variados aí que eu pudesse escolher. Poderia manter os mesmos personagens, mudaria só a localidade, o contexto, o ambiente, sabe.

VINICIUS E o que que você acha da parte visual em si? Tirando a parte temática, né, que você acabou de falar. Como estão os desenhos, como os bonecos estão animados, como eles se movem? Você acha que está está claro quando o bonequinho se move, faz uma ação?

PROFESSORA C Que nem quando eu vi a primeira vez eu me bati um pouco para entender o que estava acontecendo. Depois eu peguei, né? Eu não sei se para um aluno iniciante se ele pegaria com rapidez o que está acontecendo, por isso que eu achava interessante ter a narrativa ali, né? Mas depois que se compreende, eu acho que fica bem claro, né? E eu gosto desse visual bem, já digo vintage né? E é da minha época, né, da época dos ataris. Que é uma tendência que está voltando até para os adolescentes aí que estão retomando esse tipo de tela e tal. E olha que eu nem jogo, eu só vejo a piaçada falando aí, né? E então eu acho que isso é uma forma também de se aproximar do que o pessoal que gosta de jogar, né?

VINICIUS E você, acha que essa abstração que foi feita como foi pego um modelo de von neumann, o código ali que você viu que foi colocado e como ele foi transformado na animação, você acha que essa abstração ela tá bem feita, ela tá atingindo o objetivo? Você vê problemas ali?

PROFESSORA C Eu acho que atinge porque você consegue acompanhar certinho, porque ele vai mostrando, né? E aí, você vai visualizando a dinâmica, né? Pegando se é uma entrada de dados, ele vai lá, busca da entrada para jogar na memória, né? Pega no pacote e joga no registrador, aí tem a ULA que vai fazer o processamento e tal, então isso acho que fica bem claro que você consegue identificar cada instrução que ele executa na animação, ele mostra e você consegue identificar quando é a memória que está sendo acessada, quando é a ULA que é chamada quando é enviado ou para entrada ou para saída. Então isso pra mim ficou bem claro.

VINICIUS Esse software ele está sendo testado agora desse jeito aqui, com professores, né? O que você acha de testar o software em sala de aula. Se você fosse aplicado em sala de aula, como é que você utilizaria? Qual é a melhor abordagem? Ou se você acha talvez que ele não está pronto ainda para ser utilizado também, que falta alguma coisa.

PROFESSORA C Não, eu acho que em sala de aula eu utilizaria ele nas minhas aulas, primeiro nas minhas aulas introdutórias, utilizando os exemplos ali que são bem simples e são exemplos que a gente usa nas primeiras aulas de algoritmos, né? É na hora que eu fosse explicar a questão da arquitetura de von neumann, que eu abordo nessa disciplina. Eu me lembro que eu coloco o código no quadro, eu vou fazendo o passo a passo e vou explicando quem vai para a memória, quem vai tal, né? Então nesse momento eu abordaria o que é a arquitetura de von neumann, como que ela funciona e invés de eu ficar escrevendo o quadro, eu pegaria os exemplos do Core Inc. e já ia mostrando pela execução ali pela animação e falando, né? Olha a arquitetura funciona assim, o acesso a memória a ULA e tal, fazendo essa analogia. Daí depois eu iria trabalhar os algoritmos mais complexos com eles e passaria e pediria para que eles desenvolvessem os algoritmos deles usando o Core Inc. e testando a animação. Uma coisa que eu perguntar que eu não vi, se eu faço um algoritmo errado a animação mostra?

VINICIUS Se eu faço algo que não compilaria ele é como um compilador. Ele disse que não funciona.

PROFESSORA C Ele diz que não funciona.

VINICIUS A diferença é que ali é no momento eu não tô dando tão específico quanto o compilador se o código der errado, eu não vou informar linha tal erro tal. Eu vou falar sim ou não.

PROFESSORA C Beleza, mas não, não, tudo bem, então mas aí eu usaria: Desenvolva os algoritmos de vocês se quiserem em outra plataforma, mas depois testem para verificar qual é a animação que o teu código vai gerar no Core Inc. para você entender o que você fez e como que está funcionando lá em baixo nível lá na máquina. Então seria um outro tipo de exercício que eu poderia fazer e que sem o Core Inc. eu não consigo fazer, entendeu? Se eu conseguisse consigo pedir para os alunos, vai lá e monta o teu algoritmo e a gente testa, né? Discutir em sala, mas eles verem a dinâmica do algoritmo que eles próprios desenvolverem sendo animados e vendo como ocorre, é diferente, eu acho que seria interessante. Porque daí isso ajuda também na compreensão do próprio raciocínio do aluno para ele pensar sobre o que ele está raciocinando e isso é interessante. Fora que a gente pode ir para um mesmo algoritmo para o mesmo problema, gerar algoritmos diferentes, correto? E que jogando no Core Inc. ele vai gerar animações diferentes e a gente pode fazer uma análise dessas animações, uma animação que ficou mais longa, mais curta, né? E isso já mostra a questão do algoritmo que todos resolveram o problema, mas qual é o mais rápido? Porque é mais rápido? Qual usa menos memória? Então vendo na animação, fica mais fácil de compreender e identificar esse tipo de diferença.

VINICIUS Ok. Você vê além do que a gente está conversando aqui, que você já falou, você vê alguma modificação, alguma correção que você faria no Core Inc.

PROFESSORA C Além da narrativa, né?

VINICIUS Da narrativa e de mudar o ambiente que você falou de mudar temática.

PROFESSORA C Acho que o principal pra mim seria colocar a narrativa e ter a possibilidade de eu escolher uma temática, de resto para mim tá ok.

VINICIUS Você acha que tem alguma coisa, algum tema que eu não abordei até agora que eu não falei alguma coisa do software que você acha que você queria comentar?

PROFESSORA C Qual é a base teórica que vocês, a base teórica educacional, se tem alguma?

VINICIUS Chama teoria da cognição situada.

PROFESSORA C Nossa, essa eu não conheço.

VINICIUS Então ela não é muito conhecida mesmo. Ela surgiu com uma matemática lá nos anos 60 70, ela tentava fazer parecido com os exemplos do Paulo Freire que você deu, ela estava tentando fazer ensino de matemática básica, tipo levando uma pessoa no supermercado. E

daí tem 2 chilenos que escreveram um livro de biologia junto com a educação sobre isso que eles fundamentam basicamente que a gente entende que o conhecimento a gente simplesmente não vê uma coisa, pega processo dentro da nossa caixinha e expõe conhecimento, né? Não pega informações e simplesmente expõe conhecimento. A gente entende que a construção do conhecimento se dá pela cultura do aluno. A interação dele com o ambiente a todo o momento está modificando isso, então a pessoa interage com a cultura, essa cultura interage com ela daí isso vai interagir com outro conhecimento que ela está obtendo, pode ser algoritmos, por exemplo. E o jeito que ela vai interpretar isso depende com o ambiente que ela vive agora. Então, a ideia do Core Inc. é que justamente a gente está tentando abordar coisas mais mundanas, trazendo mais para a realidade do aluno, porque a gente quer que ele tente usar as ferramentas culturais que ele já tem para interpretar um conhecimento novo, em vez de a gente ficar se batendo tentando fazer ele aprender conceitos completamente novos, para daí começar a entender essa coisa.

PROFESSORA C Ai que legal!

VINICIUS Esse é um resumo do embasamento teórico.

PROFESSORA C Isso depois me manda aí, você tem um link desses materiais? Eu to interessada.

VINICIUS Eu cito tudo, né? Tá tudo lá, eu te mando.

PROFESSORA C Manda pra mim que eu dou uma olhada, porque se a gente se falou em cultura né, a gente tá a nossa sociedade hoje está mergulhada na na cultura digital se a gente parar para pensar. A sociedade e as pessoas estão sendo influenciadas culturalmente pelo ambiente digital, né? Isso também gera modificações na forma de pensar e raciocinar. Então, a partir do momento que eu vou aprender um novo conhecimento via um jogo como o Core Inc. é uma forma diferente de eu aprender. E quando eu faço via esse jogo uma analogia com uma realidade fora da cultura digital, que é a nossa realidade real, isso também me ajuda a entender melhor não só o conceito abstrato, mas também o funcionamento da lógica da cultura digital. Então, se você parar para pensar, eu comecei a pensar nisso, quando você falou em cultura, porque eu tô lendo sobre isso aí, né? O quanto a cultura digital está influenciando nas relações das pessoas e até na forma delas praticarem a cidadania, né? Isso tem a ver quando você pensa lá em inclusão digital, eu preciso pensar nisso também, porque dependendo da atividade que eu propor no meus projetos, eu vou estar excluindo em vez de estar incluindo a pessoa, né? E o Core Inc. me fez pensar nisso também. O interessante que é uma dialética, você altera a realidade que é a cultura digital e ao mesmo tempo eu trago a minha realidade real para dentro do ambiente virtual que é essa analogia que você está tentando fazer para que as pessoas consigam aprender esses conceitos abstratos, né? Então nossa, é uma liga bem interessante.

VINICIUS Sim, a gente tem que fazer essa mistura, os próprios personagens que você vê em tela, por exemplo, eles teve várias interações, não é a gente tipo, eles são inspirados em coisas que a gente já conhece, né?

PROFESSORA C O Blue, você pensa no filminho?

VINICIUS O Blue é uma das inspirações, né? Daí, por exemplo, a cultura evolui muito rápido, não é? A gente está vendo os alunos encontrando referências novas a todo momento e as coisas ficam antigas, rápido, não é? A gente tem isso também. Então, por exemplo, a gente tenta pegar esses personagens, como você disse, o Blue ainda é algo recente, a gente espera que os alunos conheçam. Você pega, por exemplo, a ULA, ela é inspirada em filme do Homem-Aranha. Então, a unidade de controle é inspirada num seriado de xadrez da Netflix, que é igualzinha a mocinha da Netflix. Então a gente tem essa ideia de deixar tudo bem similar, né?

PROFESSORA C E você falando em cultura, a gente pensa na cultura tradicional que não tem a ver com a cultura digital. E daí quando eu digo lá que seria interessante a gente ter que temáticas diferentes? Essas temáticas me permite acessar culturas tradicionais diferentes. Eu

poder levar lá o teu jogo dentro da cultura Nordestina, que é totalmente diferente da cultura lá da galera do Rio Grande do Sul, em termos de palavras, de de dialetos de dia a dia de clima, de comida, né? Isso tudo é a cultura tradicional que a gente fala. E eu poder ter temáticas diferentes me possibilita também trabalhar, acessar culturas tradicionais diferentes, porque a cultura digital, não me aprofundei muito ainda nisso, tá? É uma coisa que é uma ideia uma compreensão que eu tô começando a ter agora. A cultura digital já não, ela é universal, que todo mundo entra nesse mundo virtual e todo mundo tem que se enquadrar dentro dele para interagir com todo mundo. Aí que se você não entende os artefatos, as dinâmicas, os processos da cultura digital, você não consegue, você fica excluído. Diferentemente da cultura tradicional, que ela varia de região para região de país para país, né? E aonde as pessoas se sentem em casa, digamos assim, né?

VINICIUS Faz sentido. É, tá. Daí eu tenho uma última pergunta para você, é mais um pedido, não é? Não tem a ver totalmente com o Core Inc. nem nada. Essa entrevista aqui, como eu falei para você, ela vai passar por um processo qualitativo de validação, né? Eu vou transcrever isso e daí vai ser feito uma análise temática em cima do que a gente falou e separado frases, eu vou escrever sobre tudo mais, né? Eu queria saber se está disposta a revisar o material que vai ser gerado a partir dessa entrevista. É uma forma de validação.

PROFESSORA C Não, pode mandar para mim que eu reviso sim.

VINICIUS Então, é da ideia que daí eu mando para você e você me diz se eu não mal interpretei o que você falou.

PROFESSORA C Tranquilo, pode mandar.

VINICIUS Tá, e por último, uma pergunta sobre a entrevista em geral em si, você acha que teve alguma parte do processo que você não gostou, você sentiu que se sentiu desconfortável em algum momento da entrevista ou algo assim?

PROFESSORA C Não, para mim foi bem tranquilo. Eu gostei desse processo de interagir com o software primeiro, daí responder, porque quando eu estou respondendo o questionário, a gente vai pensando sobre o software. Quando chega na entrevista com você, eu já sei do que nós estamos conversando. Que é diferente quando você cai numa entrevista direta assim, não tem toda essa, mesmo que seja uma vivência no mesmo momento, digamos assim. Mas é um momento anterior, a priori, um pouquinho antes, mas que me ajuda a me dar tranquilidade em responder tuas perguntas com mais veracidade do que eu estou achando.

VINICIUS Então é isso!

F.4 TRANSCRIÇÃO ENTREVISTA PROFESSOR D

VINICIUS A primeira coisa que eu queria saber. É para você, no teu trabalho ou na tua pesquisa, ou quando o senhor está dando aula alguma coisa assim, qual que é a relevância para você do modelo de von neumann, que o Core Inc. está tentando abstrair, é desses temas que o software está tentando abordar. Qual que é a relevância deles para você no seu dia a dia no seu trabalho?

PROFESSOR D Eu acho que nas disciplinas que eu já lecionei, nesse nível de detalhe que o software está chegando, eu nunca usei, nunca cheguei a basear assim nenhuma explicação, não é nesse nível de detalhes que você está chegando no software, mas eu acho que porque nas disciplinas de programação que eu já lecionei até hoje, era disciplinas um pouquinho mais para frente. Já lá pelo terceiro quarto semestre da faculdade, quando já a gente já não desse tanto assim para explicar esse tipo de detalhes. Era programação orientada a objetos, então até já não vinha tanto ao caso, eu falar tão baseado assim como que funciona a arquitetura de von neumann no que a gente estava fazendo. Agora que eu estou lecionando uma disciplina de arquitetura e de

algoritmos e programação, que é um pouquinho mais básica, aí talvez viesse mais a calhar, mas ainda estou começando então eu não tenho a experiência para te dizer.

VINICIUS Ok, então o nível de detalhe que você viu no software, é só para confirmar mesmo, ele é maior do que o que normalmente você usa?

PROFESSOR D Sim.

VINICIUS E sobre o tema assim, de ferramenta para fazer visualização de software, a próprio ferramenta de algoritmos animados, que é o que o Core Inc. que você viu agora, o que você acha em geral desse tipo de ferramenta, você acha que ela é legal? Você não gosta muito? Você utiliza? Como é que é?

PROFESSOR D Eu acho muito legal. Eu já usei algumas ferramentas de visualização na disciplina de orientação a objeto que elas não necessariamente animavam o código, mas ela tinha um componente visual que facilitava de você compreender os conceitos e coisinhas que eu estava passando para os alunos e a resposta dos alunos costuma ser muito positiva, costumava ser muito positiva para isso. Então eu acho muito legal.

VINICIUS Ok? E o que que você acha de quando esses conteúdos, os próprio softwares de visualização, eles estão tentando se adaptar a realidade do aluno aos gostos dele, a própria cultura dele, o que o Core Inc. está tentando fazer é um pouco disso, trazendo um pouco mais para a nossa realidade, né? O que você acha de soluções que tentam isso, não só software, mas talvez o professor em sala de aula trazer o conteúdo de uma forma mais contextualizada e tudo o que você acha desse tipo de abordagem?

PROFESSOR D Acho interessante, principalmente para usar nas disciplinas introdutórias quando nesses conceitos que a gente tenta passar pra eles ainda são muito abstratos. As vezes a gente procura muitas formas diferentes de explicar, mas acho que tendo uma coisa para eles verem, acontecendo e seguirem um passo a passo visualmente eu tendo a achar que vai funcionar, vai ser uma ferramenta interessante para para o aluno que vai ter mais facilidade de entender né? Enxergando alguma coisa acontecendo na frente dele.

VINICIUS Falando do software específico do Core Inc, quais que foram pra você as suas primeiras impressões quando você abriu o software e começou a utilizar ele?

PROFESSOR D Como você me falou que era pra disciplina de algoritmos e programação e bem para o início, eu esperava que fosse mais simples. Depois quando eu comecei a usar, eu entendi o que é que estava acontecendo. Talvez eu esteja falando isso, porque eu estou considerando o meu público aqui, que costuma ter muita dificuldade com programação, principalmente no início.

Eu não tive a impressão que a forma como estava funcionando seria necessariamente para o aluno que está começando ou para um aluno que está fazendo a primeira disciplina de algoritmos de programação eu acho que talvez fosse mais útil para para o aluno que está que está aprendendo arquitetura do que para o aluno que está aprendendo programação.

VINICIUS O senhor comentou em nível do aluno, você acha que a forma como o conteúdo é ministrado aí onde o senhor da aula agora é diferente de onde a gente está? Tipo, eu me baseio aqui nos conteúdos e nos materiais da Federal do Paraná, né? Você acha que as coisas são apresentadas mais devagar, onde você está?

PROFESSOR D Sim!

VINICIUS Entendi. Então, tipo, esse tipo de conteúdo agora o aluno não estaria havendo em algoritmos 1, esse tipo de tipo de tela e o código ali?

PROFESSOR D O código sim, mas essa ideia de ele entender da forma como é que o algoritmo está funcionando dentro da máquina, né? De buscar as instruções, fazer solicitações da memória e sair, não aconteceria aqui no primeiro semestre.

VINICIUS Entendo a parte conceitual da arquitetura e tudo não é abordada logo de início aí.

PROFESSOR D Não. Um dos nossos cursos a disciplina de área de arquitetura só vem no quinto semestre.

VINICIUS Um, tá? Interessante. Aqui a nossa ideia é esta porquena matéria de algoritmos você tem apresentado o básico, a arquitetura mais simples, né?

E daí me diz o que que você achou em geral da usabilidade do software, dos tutoriais dele? Você acha que é fácil de usar, foi fácil de entender o que estava ali ou não?

PROFESSOR D Olha, eu achei no início quando, quando rodou o primeiro teste eu fiquei um pouquinho confuso, acho que é porque a tela não é tão grande, a tela e você tem que prestar atenção no que está acontecendo, apesar de que quando aparece aquelas telinhas lado que está acontecendo agora com a instrução, que a unidade de controle está buscando agora, isso aqui é o que está sendo buscado na memória, aquelas paradinhas que ele dá para mostrar cada estágio, é legal. Acho que quando chegou no segundo teste, "ahh tá é isso aqui que está acontecendo". Aí eu consegui ir prevendo qual era a próxima coisa que ia acontecer, porque eu já estava entendendo como era o funcionamento. Mas assim, tem uma leve curva para você compreender, mas aqui também acho que é porque eu estava esperando uma coisa mais simples, então eu não estava esperando aquele nível de detalhamento, mas acho que se você já for sabendo, ah isso aqui é o que você vai esperar está bem claro. Até porque depois que você vê acontecendo um ou dois testes, você entende bem certinho quem é quem naquele cenário e qual é a função de cada de cada personagem.

VINICIUS OK, e você acha que o tutorial explica bem o suficiente esse processo? Você acha que falta alguma coisa?

PROFESSOR D Pelo que eu me lembro, não aguardei a informação de que você podia colocar o mouse em cima dos personagens do jogo e ver o que eles estão fazendo. Mas me pareceu... porque mesmo se não tivesse o tutorial e tivesse me dito aperta o botão azul e depois aperta executar eu teria entendido que está acontecendo, não é? Não é tão contra intuitivo, dá pra ele entender só vendo acontecer. Uma coisa que eu achei que podia ajudar seria se você tivesse alguma indicação no código para dizer em qual linha está. Tem aquele momento que você anuncia, quando ele busca a instrução lá no canto esquerdo, mas depois não tem nenhum jeito de você acompanhar enquanto está acontecendo, em qual linha está na execução. Daí eu acho que poderia ter isso aí, ficaria mais intuitivo.

VINICIUS No próprio código, você diz?

PROFESSOR D Sim, no código, talvez um a setinha que vai andando, ou então um destaque em qual linha que está.

VINICIUS Tá. E o que você que você acha, da temática do software? Do jeito que ele tenta pegar esses conhecimentos mais abstratos do modelo e tudo mais e abstrair para essa temática com uma fábrica, bonequinhos e tudo mais. Você acha essa temática interessante? Você acha que ela está muito distante dos alunos? Qual que é a sua opinião?

PROFESSOR D Acho que é bem claro, acho que é bem interessante, inclusive quando eu costumava dar aula de introdução à computação eu muitas vezes, quando eu ia explicar mais ou menos esse funcionamento em alto nível de como é que esta as operações ocorrem dentro do computador o exemplo que eu dava era uma coisa parecida de coisas conversando entre si e fazendo solicitações uns aos outros. Então, para mim esse contexto que você criou ele é bem intuitivo.

VINICIUS E você acha que esse contexto ele pode ser benéfico? Assim é entre você demonstrar essa mesma estrutura com por exemplo, só formas geométricas e aqui demonstra ela com bonequinhos e tudo, você acha que tem uma diferença significativa entre os dois?

PROFESSOR D Bonequinhos é bem mais interessante. Também com os bonequinhos, tem mais uma facilidade assim que você entender que é uma coisa que está fazendo uma ação. Por que usar só formas geométricas, como sei lá um fluxograma, você ainda tem mais uma camada de abstração por cima, que o bonequinho deixa claro que é uma coisa dentro do computador que tem uma ação que causa uma modificação. Acho bem bem mais Claro.

VINICIUS Ok. O que você acha, tirando em si a parte temática em geral, o que você acha da representação visual em si das animações? O jeito que os bonequinhos se movem, para onde eles vão, as ações que eles fazem, você acha que é claro quando você visualiza os bonequinhos, o que eles estão fazendo no momento?

PROFESSOR D Eu acho que você podia deixar mais claro algumas coisas, tipo onde é que é o quê na memória? Porque é que às vezes a bonequinha da unidade de controle ela deixa o pacotinho do lado esquerdo ou do lado direito. Eu acho que a tela podia ser um pouquinho maior para ver as animações e acho que você podia ter um controle de velocidade.

VINICIUS OK.

PROFESSOR D Mas eu acho que no início, principalmente se eu pudesse ver as animações mais devagar, mostrar para o aluno as animações mais devagar seria bastante benéfico.

VINICIUS O que você acha da abstração que o Core Inc. fez desse processo da execução de um algoritmo? A gente sabe que não é exatamente assim que acontece no computador, não é? Existe um nível de abstração do que acontece dentro de uma arquitetura de von neumann do que está na animação. Você acha que esse processo de abstração, de transformação, para ser esa animação, ele foi bem feito ou você tem problemas com ele?

PROFESSOR D Eu achei um pouquinho confuso, já que como você coloca aqui que a sua intenção é que seja para alunos dos primeiros dos primeiros semestres. Porque você está presumindo aí que os elementos de entrada e saída, eles estão mapeados para a memória, quando você tem um read lá, ele vai primeiro buscar na memória para depois levar para a entrada para o personagem de entrada e saída. Isso aí eu achei que é um pouquinho confuso para o aluno que está vendo isso pela primeira vez. Porque eu estava pensando, tipo, quando você dá aula programando em Portugol, e aí tem lá leia e escreva. Então, quando a gente coloca para o aluno o leia, é uma coisa de entrada e saída. Então é uma informação que vem do teclado. Então esse pulo que você dá, primeiro buscar na memória e depois buscar na entrada e saída não seria intuitivo para o aluno iniciante. Daqui! Não sei na UFPR, mas em relação ao detalhamento das coisas e a abstração que você está fazendo da arquitetura de von neumann, eu acho que, principalmente para a disciplina de arquitetura está num ótimo nível para um aluno que está vai ver isso pela primeira vez.

VINICIUS OK? Dito isso, a gente está comentando as diferenças entre as duas universidades, não é mesmo, nem eu sei responder se se a minha impressão está completamente correta, porque a gente não testou com alunos, né? Pode ser que... O que o senhor acha da utilização do Core Inc. em sala de aula, você acha que ele está apto para ser utilizado em sala de aula atualmente? E se ele estiver, como você utilizaria ele em sala de aula?

PROFESSOR D Essa foi uma coisa que eu fiquei pensando, porque depois que eu entendi como é que estava funcionando, o que estava acontecendo, eu achei super legal, mas eu fiquei pensando como é que eu organizaria uma aula ao redor dele? Que seja, além de mostrar os exemplos e explicar o que está acontecendo nos exemplos. Como é que eu faria isso para ele ter mais utilidade para mim. E primeiro eu acho que assim para usar para apresentar para o aluno para ensinar seja lá o que for, eu acho que seria uma excelente como base pra mostrar para ele "Olha isso aqui vai acontecer e isso isso e isso", e depois eu destrinchar.

Ah, essa unidade de controle ela serve para isso aqui, então por isso que que ela está funcionando desse jeito, por isso que ela tenha essa ação na animação. Se eu fosse usar ele do

jeito que ele está agora eu faria desse jeito para mostrar a cara de cada coisa e explicar o que está acontecendo e por que ela faz aquilo, na ordem que ela faz aquilo, considerando as outras coisas que estão acontecendo. Porque a unidade de controle faz isso primeiro e depois a memória responde desse jeito e depois a ULA responde. Se eu fosse usar ele do jeito que está agora seria dessa forma.

Para eu organizar uma aula, por exemplo, para colocar o aluno para escrever um programa ali dentro do próprio software e observar o que está acontecendo eu acho que esperaria até ter algum controle da velocidade.

VINICIUS Então, o senhor pensa que para você botar o software na mão do aluno do jeito que está agora não está legal. Você gostaria que tivesse o controle da velocidade das animações e seria o suficiente para o aluno já poder utilizar o software?

PROFESSOR D Ou um controle de velocidade, ou então um controle de controlar o passo a passo, assim cada coisa e cada ação que está acontecendo, pelo que está acontecendo no sistema, como por exemplo, deixar mais claro assim: primeiro a gente buscar instrução, esse é o primeiro passo, depois faz isso, depois busca... Entendeu? Se a gente tivesse um controle assim de cada ação nesse destrinchar do que está acontecendo na arquitetura de von neumann, eu acho que seria realmente sensacional, principalmente para ensinar o início da parte de arquitetura ou então já que você aí, vocês abordam isso no início de programação. Aí eu acho que seria realmente muito legal.

VINICIUS Você diz, por exemplo, no começo naquele vídeo é citado os 7 passos do ciclo de instrução, né? Você acha que se esses 7 passos fossem divididos e se ele pudesse avançar passo por passo em cada ciclo da animação seria interessante, é isso?

PROFESSOR D Sim.

VINICIUS É você pensou em outras modificações em outras correções, que o software poderia ter que a gente não comentou ainda alguma coisa assim?

PROFESSOR D Ele já observa quando acontece com uma exceção? Uma divisão por zero.

VINICIUS Ele funciona quase como um compilador nesse caso. Se você cometer um erro, tipo uma divisão por zero, a animação não vai executar, não é? Eu vai aparecer uma mensagem que deu erro ao compilar ou executar o código, né? Mas no momento ele não é tão específico quanto um compilador. Se você pega um compilador de Pascal, por exemplo, ele vai disparar uma exceção e vai falar, ó na linha tal, deu erro tal, né? O Core Inc. não é específico nesse nível, ele só fala que deu erro.

PROFESSOR D Mas se for uma situação de que ele não consegue detectar que dá uma exceção e na hora de rodar acontece uma sessão.

VINICIUS Isso é impossível.

PROFESSOR D No Pascal?

VINICIUS Vou explicar, é que isso é uma parte técnica.

PROFESSOR D Ah, é porque como você já diz a entrada antes de rodar não tem como uma variável acabar chegando em um valor...

VINICIUS Eu tenho que pedir a entrada antes primeiro por questões técnicas, já eu não consigo manter uma máquina esperando um processo, esperando uma requisição vim pra daí eu passar. Ao mesmo tempo, o que está acontecendo é que eu estou pegando aquele código que o senhor está vendo e transformando em uma entrada válida para meu módulo de animação, né? E daí quando já vem todas as entradas e tudo é a realidade é que o código não está sendo executado em tempo real de verdade junto com animação. O que aconteceu é que quando você apertou aquele botão, eu executei o código, eu gerei outros processos para gerar as entradas para animação e eu passei tudo para ele, então é impossível você ter uma exceção no meio porque eu

já executei tudo antes, se ele deu uma exceção quando eu estava executando e eu já avisei o aluno que deu errado.

PROFESSOR D Seria interessante, não é? Vai que dá várias luzes na fábrica e eles se enlouquecem.

VINICIUS E PROFESSOR D [risadas]

PROFESSOR D Deu ruim...

VINICIUS É uma possibilidade. No fim é uma possibilidade. As luzes só não podem aparecer do meu lado aqui eu tenho que resolver esse problema antes. Tirando essa parte dos erros você pensou em mais alguma coisa, em mais alguma modificação ou correção.

PROFESSOR D Não.

VINICIUS É, você acha que tem alguma coisa, algum apontamento, alguma crítica ou sugestão de algo que você achou relevante até agora e que eu não perguntei que eu não citei até o momento?

PROFESSOR D Eu acho que no momento do tutorial, lá no início, se você colocar, por exemplo, um exemplozinho do que é o breakpoint? Eu fiquei me perguntando, se eu não colocar nenhum breakpoint e aí na próxima mensagem diz que se você não colocar nenhum ele vai executar tudo, né? A única coisa que eu pensei de diferente foi isso. Eu até eu ia fazer isso, de colocar um breakpoint para olhar qual é diferença, de quando tem um breakpoint para quando não tem nenhum, mas eu fiquei entretido com os exemplos, esqueci de fazer isso. Só acho que é uma coisa que me ocorreu agora que no tutorial não ficou claro para mim.

VINICIUS É só esclarecendo então, se você coloca um ponto de parada, se você coloca um breakpoint, a animação vai executar somente as linhas que você marcou, né? Então, tipo, se você vê aquele código de Bhaskara inteiro lá que tinha e você botasse um breakpoint na última linha dele, que é o último write, a animação ia começar direto do último write, ia mostrar ele e ia acabar.

PROFESSOR D Beleza.

VINICIUS É isso que ia acontecer. Se o senhor realmente ficou curioso, você pode lá testar porque o negócio está online, não é? Enfim, mas é basicamente essa ideia, o aluno pode escolher o que vai ser animado e o que não. E daí é basicamente como um compilador ou uma IDE mesmo, se você botar um breakpoint em uma linha que está dentro de um if e daí o if nem entrou, daí ele vai pular o breakpoint porque não executou a linha.

É, acho que eu fiz todas as perguntas que eu queria sobre o software e tudo mais, não é? Então eu queria perguntar para o senhor., é o seguinte, a gente está fazendo esse processo aqui de entrevista, eu estou falando com você, foi respondido o questionário e tudo. A entrevista em especial, ela vai passar por um processo de análise, não é? Eu vou transcrever isso, eu vou fazer uma análise temática em cima, escrever um texto e tudo mais. Eu gostaria de saber se o senhor está disposto isso não é obrigatório, tá? Se o senhor está disposto a contribuir na validação desse material de análise da entrevista que vai ser gerado. A contribuição seria basicamente, eu vou escrever tudo e daí eu vou mandar para o senhor daqui a alguns dias eu mandaria para o senhor daqui alguns dias a análise da nossa entrevista e a transcrição dela, e daí você leria e de alguma forma com alguma outra reunião dessa ou no formato de texto, o que o senhor preferisse você diria para mim, pô "isso aqui não foi bem o que eu quis dizer.", se não tá tudo certo, para evitar a confusão do tipo você falou uma coisa e daí na hora que eu fui analisar eu entendi que o senhor falou outra, entendeu? Eu queria saber se você está disposto a fazer esse processo ou se não, não lhe interessa.

PROFESSOR D Posso, posso claro. Mas vai ter prazo?

VINICIUS Assim. A minha experiência, eu fiz isso com outros professores até agora, não é? Eu provavelmente vou levar em torno de aproximadamente uma semana para devolver

isso para o senhor, talvez menos se tudo der certo e daí o processo de avaliação normalmente é rápido com os outros professores, não é? Eles lêem rapidamente, fala não tá certo, isso aqui está errado. O que mais demorou para mim foi um professor que marcou uma reunião comigo e ficou 15 minutos falando, né?

PROFESSOR D Muita coisa pra transcrever.

VINICIUS Ah sim, eu automatizo a transcrição, não escrevo absolutamente tudo. Senão ficaria difícil, existem softwares que auxiliam nessa parte.

PROFESSOR D Ah sim imagino. Mas pode mandar que eu dou uma olhada assim e a gente pode conversar de novo, se necessário.

VINICIUS Por último, durante a entrevista, em algum momento que você acha que você se sentiu desconfortável, tem alguma coisa no processo aqui que você não gostou? Se tem alguma sugestão, alguma crítica?

PROFESSOR D Não, foi tranquilo.

VINICIUS Então é isso, muito obrigado por participar.

APÊNDICE G – DEMONSTRAÇÃO DETALHADA DA IMPLEMENTAÇÃO DO CORE INC.

Este apêndice é responsável por demonstrar de maneira detalhada alguns dos pontos de maior destaque da implementação do Core Inc., com os desafios encontrados e como foram superados ao longo do desenvolvimento. É abordado aqui o processo de captura do código do aluno, as mutações feitas ao mesmo e a execução das animações de um ponto de vista de codificação.

G.1 MANIPULAÇÃO DE CÓDIGO-FONTE

O primeiro passo para a construção dos algoritmos animados da ferramenta Core Inc. ocorre com a captura dos dados necessários para a animação ser executada. Estes dados são: o código-fonte digitado pelo aluno; os valores que serão utilizados na entrada de dados (quando necessário); e uma lista de valores inteiros indicando quais linhas do código serão animadas. No caso da lista de linhas estar vazia, são animadas todas as linhas de código.

Este conjunto de dados é enviado através de campos de texto e marcações no código através de uma requisição do tipo POST. O servidor desenvolvido em NodeJS + Express é responsável por capturar estes dados e manipulá-los de forma a serem úteis para o módulo de animação do Core Inc.

O primeiro passo de tratamento de código é validar se o código em Pascal submetido está dentro das regras de utilização do Core Inc.. A função que verifica se somente estruturas e comandos válidos são utilizados no código é demonstrada no listing G.1. Esta etapa é importante pois o código do aluno está sendo executado diretamente no servidor da aplicação. Portanto, medidas de segurança são necessárias para garantir que nenhum código malicioso que afete o funcionamento do Core Inc. seja executado.

Listing G.1: Checagem de Código Válido

```

1 function CheckLegal(script){
2     const regexIllegalFunctions = /\b(assign|append|ChDir|DumpStack|Erase|
      KillThread|Get ([a-zA-Z]) *|Sys ([a-zA-Z]) *)\b/gi;
3     const regexIllegalUses = /\b(uses)\b/gi;
4     var found;
5
6     found = regexIllegalFunctions.exec(script);
7     if (found) {
8         return true;
9     }
10    found = regexIllegalUses.exec(script);
11    if (found) {
12        return true;
13    }
14    return false;
15 }
```

Posteriormente, é verificado se o código submetido é compilável. Um requisito primordial para a construção das animações é que o código esteja livre de falhas que impeçam sua compilação. Portanto, o código é salvo em um arquivo dentro do servidor e então é compilado. Caso falhe ao compilar o processo é cancelado e o usuário notificado, como demonstrado no listing G.2.

Listing G.2: Verifica Compilação

```

1 try {
2   fs.writeFileSync(filepath + filename + ".pas", params.body.script,
3     function (err) {
4       console.log('success to create ' + filename + ".pas");
5     });
6   result = exec(execCompile + filepath + filename + ".pas" + ' -o"' +
7     filename + '.exe"').toString();
8 }
9 catch (error) {
10   console.log(error.message);
11
12   fs.unlinkSync(filepath + filename + ".pas");
13
14   return callback([], new Error("Falha ao compilar!"));
15 }

```

Após garantir que o código submetido está apto para ser convertido em uma animação é realizado um processo de mutação do código original, para que o resultado final sejam os dados necessários para a execução da animação. Este processo é ilustrado através da figura G.1, onde a primeira etapa é a previamente demonstrada de validação do código.

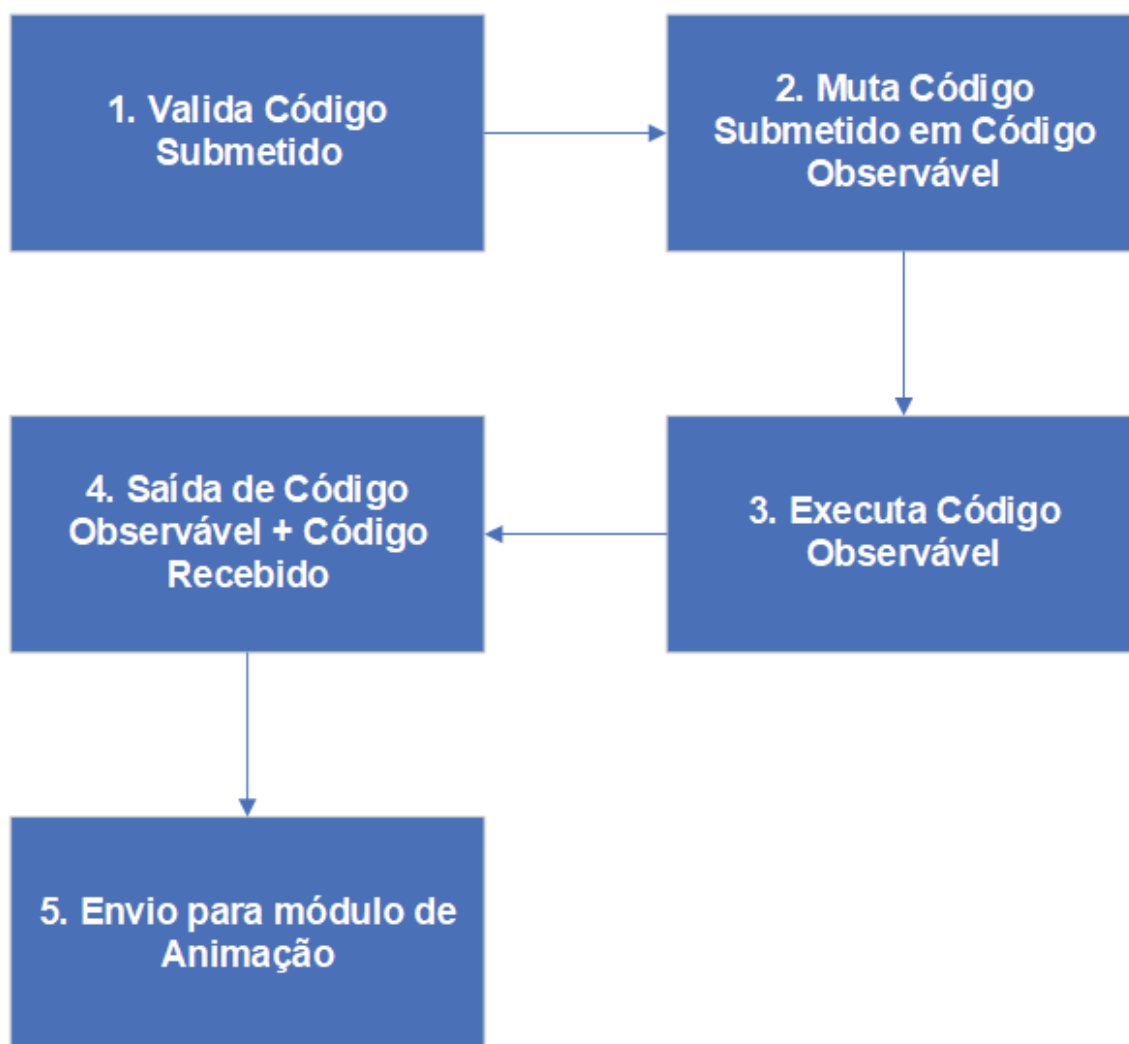


Figura G.1: Processo de Manipulação do Código

A segunda etapa foi construída para gerar o que é chamado de "Código Observável". Este é necessário para verificar o conteúdo das variáveis antes e após a execução de instruções que modificam seus valores. O processo é essencial, pois substitui a necessidade de depurar o código submetido pelo aluno. Depuração esta que não foi viável de ser executada de forma automática, devido a dificuldade do compilador da linguagem Pascal interagir com NodeJS. Portanto, esta etapa soluciona o problema de executar uma depuração tradicional do código através do compilador.

A construção do código observável se dá pela análise linha a linha do código submetido, de forma a mutá-lo em um novo código que retire qualquer interação necessária de entrada de dados, ao mesmo tempo em que verifica os valores dos operadores presentes na linha antes e depois da execução da mesma. Por exemplo, uma linha *read* com uma entrada específica como no listing G.3 ao passar pela mutação de código submetido para código observável fica como demonstrado no listing G.4.

Listing G.3: Código Submetido

```
1 program imprime_se_positivo;
2 var a: integer;
3 begin
4   read(a);
5 end
6
7 Entrada:
8 42
```

Listing G.4: Código Observável

```
1 program imprime_se_positivo;
2 var a: integer;
3 begin
4   writeln(a);
5   a := 42
6   writeln(a);
7 end
```

Este processo de mutação é codificado de maneira complexa, pois cada comando possui uma lógica diferente de tratamento para torná-lo em código observável. Enquanto entradas de dados precisam ser substituídas por comandos *write* intercalado com uma atribuição, o tratamento do comando *if* para controle de fluxo faz uso de pilha para armazenar corretamente a ordem dos comandos executados. Contudo, um dos pontos em comum entre todos os comandos é o método que são identificados, através de expressões regulares (*regex*). As expressões utilizadas para esse processo são demonstradas no listing G.5

Listing G.5: Expressões Regulares

```
1 const regexCleanSpaces = /\s\s+/g;
2 const regexEndOfLine = /([\s\S])(;)/gi;
3
4 const regexProgram = /program\s*(\S+)/gi;
5 const regexVariable = /(.*):(?:\s)?(char|integer|boolean|real|string)/gi;
6 const regexWrite = /((?:write|writeln)(?:\s)?\()(.*)(\))/i;
7 const regexReplaceWrite = /((?:write)(\s)*\()/i;
8 const regexRead = /((?:read|readln)(?:\s)?\()(.*)(\))/i;
9 const regexIf = /if(?:\s)?(.*)\sthen/gi;
10 const regexElse = /(else)(?:\s)if(?:\s)?(.*)\sthen/gi;
11 const regexAttr = /((\S+)(?:\s)?):=(.+)/gi;
```

```

12
13 const regexQuote = /(["']) (?: (?: (\\?)) \2.)*? \1 /gi;
14 const regexCleanFunctions = /\b([a-zA-Z])*\ /gi;
15 const regexCleanVar = /^[^A-Za-z0-9\*<>=+-\s\]/gi;
16 const regexCleanSignals = /[<>=+-/*]/gi;
17 const regexCleanBool = /(and|or|not|xor)/gi;
18 const regexCleanVarConst = /(var|const)\s/gi;

```

A terceira etapa listada na figura G.1 é onde o código observável será executado. Através da saída gerada por este código são obtidos os dados do que está contido dentro de cada variável. O código observável é desenvolvido de forma que o conteúdo das variáveis e o resultado das expressões de controle de fluxo entreguem como saída os dados nos momentos exatos em que são necessários. De forma que a animação seja atualizada da mesma forma que um depurador que age em tempo real no computador seria, ainda que o código tenha sido executado de forma completa previamente.

Para isto é criado um novo arquivo na linguagem Pascal que contém o Código Observável. A saída da execução deste código é armazenada para uso posterior, como demonstrado no listing G.6. Neste código o conteúdo da variável *scriptToRun* contém o Código Observável e o resultado final da execução é armazenado na variável *result*.

Listing G.6: Executando o Código Observável

```

1 fs.writeFileSync(filepath + filename + "_debug.pas", scriptToRun, function
  (err) {
2   console.log('success to create ' + filename + "_debug.pas");
3 });
4 fs.writeFileSync(filepath + filename + "_debug.txt", "", function (err) {
5   console.log('success to create ' + filename + "_debug.txt");
6 });
7
8 result = exec(execCompile + filepath + filename + "_debug.pas" + ' -o"' +
  filename + '_debug.exe').toString();
9 var absolutePathExe = path.resolve(filepath + filename + "_debug.exe");
10 var absolutePathTxt = path.resolve(filepath + filename + "_debug.txt");
11 result = exec(absolutePathExe + " > " + absolutePathTxt);
12 result = fs.readFileSync(absolutePathTxt, "utf8");

```

Após o Código Observável estar convertido e a saída de dados de sua execução armazenada, ambos são unidos para completar as entradas necessárias para o módulo de animação. A função responsável por esse processo é demonstrada abaixo no listing G.7, em que é possível demonstrar a complexidade do processo. Exige-se a identificação de vários parâmetros para que seja feita a união de ambos os conjuntos de dados de maneira correta, dado o processo diferente que cada comando possui.

Listing G.7: Montagem da Entrada de Dados da Animação

```

1 while (cont < parsedCode.length) {
2   if (parsedCode[cont] == null) {
3     index = programIfs.findIndex(element => element[0] == line);
4     if (index > -1) {
5       line = programIfs[index][2];
6       cont = programIfs[index][2] - 1;
7     }
8   } else {
9     if (parsedCode[cont][0] == "if") {
10      var variables = parsedCode[cont][1][4];
11      for (var pos = 0; pos < variables.length; pos = pos + 3) {

```

```

12         var aux = GetVariable(variables[pos]);
13         variables[pos+2] = aux[2];
14     }
15     parsedCode[cont][1][4] = variables;
16     index = programIfs.findIndex(element => element[0] == line);
17     var aux = programValues.shift();
18     var resultExpression = aux != undefined ? aux.split(" ") : aux;
19     if (resultExpression == undefined || (resultExpression[0] != "true"
20         " || (resultExpression[0] == "true" && parseInt(
21         resultExpression[1]) != line))) {
22         parsedCode[cont][1][3] = "Falso";
23         programValues.unshift(aux);
24         line = programIfs[index][2];
25         cont = programIfs[index][2] - 1;
26     } else {
27         parsedCode[cont][1][3] = "Verdadeiro";
28         if (programIfs[index][3] != null) {
29             var aux = programIfs[index];
30             programIfs[index] = null;
31             var aux2 = new Array(3);
32             aux2[0] = aux[2] + 1;
33             aux2[1] = aux[3];
34             aux2[2] = aux[4];
35             programIfs[index] = aux2;
36         } else {
37             programIfs[index][0] = 0;
38         }
39     }
40 } else if (parsedCode[cont][0] == "write(y + z)") {
41     var variables = parsedCode[cont][1][3];
42     for (var pos = 2; pos < variables.length; pos = pos + 3) {
43         variables[pos] = programValues.shift();
44     }
45     parsedCode[cont][1][3] = variables;
46     parsedCode[cont][1][2] = programValues.shift();
47 } else if (parsedCode[cont][0] == "write") {
48     var variable = parsedCode[cont][1][2];
49     if (variable[2] == null) {
50         variable[2] = programValues.shift();
51         variable[0] = variable[2];
52         parsedCode[cont][1][2] = variable;
53     }
54 } else if (parsedCode[cont][0] == "x = y + z") {
55     var variables = parsedCode[cont][1][4];
56     for (var pos = 2; pos < variables.length; pos = pos + 3) {
57         variables[pos] = programValues.shift();
58         UpdateVariable(variables[pos-2], variables[pos]);
59     }
60     parsedCode[cont][1][3] = programValues.shift();
61     UpdateVariable(parsedCode[cont][1][1], parsedCode[cont][1][3]);
62 }
63 cont++;
64 line++;
65 }

```

Após este processo o servidor retorna a estrutura necessária como entrada para o módulo de animação, através da requisição inicialmente realizada pelo sistema. Durante esta etapa, o

vetor de dados a serem animados é filtrado de acordo com as linhas selecionadas pelo aluno e o resultado dos controles de fluxo.

G.2 MÓDULO DE ANIMAÇÃO

As animações do Core Inc. são estruturadas em 3 momentos diferentes, sendo elas:

- **Controlador Geral:** Módulo responsável por receber a entrada de dados a serem animados e chamar as funções responsáveis por controlar cada animação.
- **Controladores de Animação:** Módulo com as funções responsáveis por controlar especificamente a execução de cada animação ao interagir com código CSS.
- **Clipes:** Módulo de códigos CSS responsáveis por sequenciar animações individuais e uni-las em clipes completos de animação.

O controlador geral é a parte do módulo de animação que entra em contato com a entrada recebida externamente e trata ela de maneira acionar as animações do Core Inc. Dado a entrada padrão de vetores que o módulo aceita ele seleciona de maneira simples quais funções devem ser chamadas, como demonstrado no exemplo do listing G.8.

Listing G.8: Controlador Geral

```

1  switch (string.toLowerCase()) {
2      case "program":
3          end = await Begin(args[0], args[1], args[2]);
4          console.log("terminou program");
5          break;
6      case "if":
7          FadeLoading();
8          end = await IfThenElse(args[0], args[1], args[2], args[3], args[4],
9                                  args[5]);
10         console.log("terminou if");
11         break;
12     case "var" || "CONST":
13         FadeLoading();
14         end = await Declare(args[0], args[1], args[2]);
15         console.log("terminou declare");
16         break;
17     case "read":
18         FadeLoading();
19         end = await Read(args[0], args[1], args[2]);
20         console.log("terminou read");
21         break;
22     case "write":
23         FadeLoading();
24         end = await Write(args[0], args[1], args[2]);
25         console.log("terminou write");
26         break;
27     case "x = y + z":
28         FadeLoading();
29         end = await CalcAndSave(args[0], args[1], args[2], args[3], args[4],
30                                 args[5]);
31         console.log("terminou calculate and save");
32         break;
33     case "write(y + z)":
34         FadeLoading();

```

```

33     end = await CalcAndWrite(args[0], args[1], args[2], args[3]);
34     console.log("terminou calculate and write");
35     break;
36 default:
37     console.log("Instrução Inválida: " + string.toLowerCase());

```

As funções de cada comando chamadas por este trecho de código tem diferenças de acordo com a necessidade de cada animação, mas compartilham do recurso de funções assíncronas. A programação assíncrona permite o módulo de animação trabalhar com o sequenciamento de chamadas de funções diferentes. Como visto no listing G.8 a chamada de funções assíncronas com *await* garante que trechos do código irão aguardar a conclusão de outros processos até retornar sua execução. Essa ferramenta é essencial, pois as animações precisam ser executadas de forma sequencial sem que duas estejam sendo executadas ao mesmo tempo.

O código do listing G.9, responsável pela animação de declaração de variáveis demonstra que estas funções necessitam ser declaradas com a palavra-chave *async*. Além disto este processo é repetido em maior escala ao chamar os Controladores de Animação, com uma sequência de funções assíncronas chamadas uma após a conclusão da outra.

Listing G.9: Funções Assíncronas

```

1  async function Declare(instructionName, nextInstructionName, variable) {
2      SetInstructionNames(instructionName, nextInstructionName);
3      end = await GetInstruction(instructionName, nextInstructionName);
4      end = await GetDeclare(variable[0], variable[1]);
5      end = await SetPackage(variable[0], variable[1], "");
6
7      return "done!";
8  }

```

Os Controladores de Animação possuem a função de interagir com as tags HTML e CSS que exibem as animações do Core Inc. de forma a acioná-las e desativa-las no momento correto. Elas funcionam de maneira assíncrona, assim como todo o sistema de animações do Core Inc. e possuem características semelhantes entre si. Para ilustrar o funcionamento geral destes controladores o listing G.10 demonstra a função responsável pela animação de declaração de variável.

Listing G.10: Animando Manipulando Classes

```

1  // Chama Animação CSS de Declaração de Variável
2  function GetDeclare(name, size) {
3      var contAnimationEnd = 0;
4      // DEFINE TOOLTIP
5      SetLucy("Pegando pacote (variável) declarado.");
6      SetPackageLucy(name, size, "");
7      SetPackageDeclare(name, size, "");
8
9      ("package - declare").addClass(sizePackage[size]); ("#lucy-package").addClass (
10         sizePackage[size] + "-hand");
11     ("lucy").toggleClass("get - declare"); ("#lucy-package").toggleClass("get-declare");

```

Neste trecho de código é possível visualizar a manipulação das animações através do ativamento de classes HTML, além do uso de eventos do JS para definir quando a animação foi concluída e o sistema pode avançar para a próxima animação.

Estas classes quando chamadas estão atreladas a clipes de animação da linguagem CSS. Estes Clipes são a última parte do módulo de animação do Core Inc. Eles são montados

primeiro com a construção de *keyframes* que fazem os personagens fazerem ações pequenas e específicas, como se mover do ponto A ao ponto B do cenário. O trecho de código do listing G.11 os demonstra.

Listing G.11: Keyframes de Animação

```

1  /* Animation Movement Keyframes Lucy */
2
3  @keyframes lucy-route0-1 {
4      from {}
5
6      to {
7          left: 19.3%;
8          top: 13%;
9      }
10 }
11
12 @keyframes lucy-route1-1 {
13     from {}
14
15     to {
16         left: 26.7%;
17         top: 47.8%;
18     }
19 }

```

Estes *keyframes* são sequenciados de diversas maneiras em um Clipe de animação, este que é acionado quando os objetos HTML possuem classes específicas, que são acionadas pelos Controladores de Animação. O listing G.12 demonstra como o acionamento da classe *ask-package* inicia uma animação que sequencia diversos *keyframes* diferentes, formando um clipe completo que é visto em tela pelo aluno.

Listing G.12: Keyframes de Animação

```

1  /* Animation Ask Package - sync with animation-screen.css */
2  #lucy.ask-package {
3      animation-name: lucy-route1-1, lucy-sprite-front, lucy-route2B-1, lucy-
4      sprite-right, lucy-route4-1, lucy-sprite-front;
5      animation-duration: 1s, 0.5s, 1s, 0.5s, 0.5s, 0.5s;
6      animation-timing-function: linear, steps(4), linear, steps(4), linear,
7      steps(4);
8      animation-fill-mode: forwards, forwards, forwards, forwards, forwards,
9      forwards;
10     animation-iteration-count: 1, 2, 1, 2, 1, 0.9;
11     animation-delay: 0.5s, 0.5s, 1.5s, 1.5s, 2.5s, 2.5s;
12 }

```

O sequenciamento de diversos destes Clipes completa o ciclo de processamento do Código Submetido até o momento que ele se torna uma animação completa para visualização do aluno.