



Universidade de Brasília

Instituto de Ciências Exatas
Departamento de Ciência da Computação

**Ensino do Pensamento Computacional Utilizando
Scratch: Um relato de Experiências com Estudantes
do Ensino Médio Público no Distrito Federal**

Vinícius Aguiar Monteiro

Monografia apresentada como requisito parcial
para conclusão do Curso de Computação — Licenciatura

Orientadora
Prof.a Dr.a Maristela Terto de Holanda

Brasília
2022



Universidade de Brasília

Instituto de Ciências Exatas
Departamento de Ciência da Computação

**Ensino do Pensamento Computacional Utilizando
Scratch: Um relato de Experiências com Estudantes
do Ensino Médio Público no Distrito Federal**

Vinícius Aguiar Monteiro

Monografia apresentada como requisito parcial
para conclusão do Curso de Computação — Licenciatura

Prof.a Dr.a Maristela Tertó de Holanda (Orientadora)
CIC/UnB

Prof.a Dr.a Roberta Barbosa Oliveira Prof. Dr. Edison Ishikawa
CIC/UnB CIC/UnB

Prof. Dr. Jorge Henrique Cabral Fernandes
Coordenador do Curso de Computação — Licenciatura

Brasília, 30 de setembro de 2022

Dedicatória

Dedico este trabalho e toda minha trajetória para uma **MULHER**, nordestina raiz, a minha Mãe, **Maria Aguiar Rocha**, que apesar de ter partido dessa vida ainda com 57 anos, quando eu tinha apenas 16 anos, e mesmo tento estudado até a quarta série, me formou um homem de caráter, perseverança, garra, resiliência e vontade de conquistar tudo que almejo. Ela, Mulher forte e Guerreira que criou a mim e meu irmão sozinha, infelizmente, não pôde ver em vida a minha trajetória acadêmica e a realização do seu sonho que era me ver formar. Ao realizar esse sonho honro todo o orgulho e reconhecimento que ela sempre teve por mim e a sua escolha do meu nome na lista de aprovados da Universidade de Brasília à época.

Por fim, dedico este trabalho a mim que, mesmo com todas as dificuldades, consegui concluir esse ciclo em minha vida, tendo contribuído da minha melhor forma para uma sociedade melhor, participando ativamente da Pesquisa, do Ensino e da Extensão universitária na Universidade de Brasília.

Agradecimentos

A Minha Mãe por todos os ensinamentos, os conselhos e as broncas que me formaram quem sou.

Aos meus familiares que me deram apoio e conselhos em vários momentos da minha trajetória acadêmica e pessoal, em especial a minha tia Maria das Dores, meu primo Robson, minha prima e madrinha Alécia, meu irmão Gustavo e minha esposa Isabela. Ao Eduardo que me motivou e me acompanhou no registro de matrícula na Universidade de Brasília.

Aos meus colegas de curso que me ajudaram nas disciplinas e trabalhos acadêmicos, em especial: Emerson, Andreia, Lucas, Rodrigo, Maria Zanchet, Larissa, Caio, Maria Eduarda e Pablo. Aos colegas de outros cursos que convivi e também me ajudaram, especialmente aos colegas da Engenharia da Computação que conheci no ensino remoto. Para todas as pessoas que me davam caronas para a Rodoviária ou próximo de casa ao sair das aulas à noite. Aos professores das disciplinas que contribuíram incrementalmente nos meus conhecimentos e formação acadêmica.

Agradeço imensamente a todos os pesquisadores do Parque de Inovação e Sustentabilidade do Ambiente Construído (PISAC/UnB), em especial as professoras Raquel Naves Blumenschein e Maria Vitória Duarte Ferrari, que desde 2016 me formou a pessoa e pesquisador que sou e me possibilitou chegar no detalhamento de pesquisa e trabalho que realizei.

Minha orientadora, Maristela Terto de Holanda, que, primeiramente, acreditou na minha capacidade e aceitou me orientar e que me orientou em todas as etapas do trabalho com muita vontade, força, conhecimento, didática, experiência, me possibilitando aprofundar no tema de pesquisa e trabalho que fiz.

Ao Decanato de Extensão da Universidade de Brasília (DEX/UnB) pelos apoios financeiros concedidos em dois editais e que foram essenciais para todo o desenvolvimento deste trabalho.

E, por fim, agradeço a todas as pessoas que, direta ou indiretamente, acreditaram e me auxiliaram para que esta etapa de formação pudesse ser concluída.

Resumo

Ensinar os conceitos do Pensamento Computacional na Educação Básica Brasileira é de extrema importância para o pleno atendimento das competências e habilidades recomendadas pela Base Nacional Comum Curricular para o Ensino Médio. A metodologia proposta neste trabalho tem o objetivo de inserir o Pensamento Computacional nas escolas públicas de Ensino Médio no Distrito Federal, utilizando a Linguagem de Programação Visual Scratch. Foram detalhados neste trabalho o relato das experiências de aplicação do curso, realizado em duas etapas e em três escolas, o que possibilitou aprimorar a proposta e avaliar sua efetividade. Os principais resultados observados foram que a metodologia atingiu seu objetivo, teve boa recepção entre os estudantes, que relataram que gostaram do curso em seus diferentes aspectos e perceberam o aumento em seus conhecimentos gerais de informática, bem como a necessidade de investimentos nos laboratórios de informática das escolas públicas, possibilitando a ampliação das atividades desenvolvidas pelos professores com apoio tecnológico. A partir deste trabalho, as seguintes propostas poderão ser realizadas para avanços na área da pesquisa: aplicar a metodologia em outras escolas públicas; aprimorar o curso e seus métodos avaliativos com base nas próximas aplicações, coleta e análise de dados futuros; e elaborar uma versão do curso para aplicação em escolas sem infraestrutura computacional adequada.

Palavras-chave: Pensamento Computacional, Computação na Educação, Scratch, Base Nacional Comum Curricular

Abstract

Teaching Computational Thinking concepts in Brazilian Basic Education is extremely important to fully meet the competencies and skills recommended by the Common National Curricular Base (BNCC) for high school. The methodology proposed in this paper aims to introduce Computational Thinking in public high schools in the state in the Distrito Federal, Brazil, using Scratch, a visual programming language. This paper contains the detailed reports of experiences of course application, which was carried out in two stages and in three schools, enabling improvement and evaluation of its effectiveness. The main findings were that the methodology reached its objectives; it was well received by students, who reported their appreciation for its different aspects and noticed a boost in their general computer knowledge, as well as the need of investments in current computer labs in public schools, enabling the expansion of technological-aided learning activities developed by teachers. Based on this work, the following actions may be carried out to advance this field: Implement the course in other public schools; Improve the course and its evaluation methods based on future implementations, data collection, and data analysis; and develop a course version for schools without adequate computing infrastructure.

Keywords: Computational Thinking, Informatics in Education, Scratch, Common National Curricular Base

Sumário

1	Introdução	1
1.1	Objetivos	2
1.2	Metodologia	3
1.3	Resultados	3
1.4	Organização do Trabalho	3
2	Fundamentação Teórica	4
2.1	Base Nacional Comum Curricular (BNCC)	4
2.1.1	Base Nacional Comum Curricular — Ensino Médio	6
2.2	Pensamento Computacional	7
2.2.1	Pilares do Pensamento Computacional	9
2.3	Pensamento Computacional na Educação Básica Brasileira	12
2.3.1	Pensamento Computacional no Ensino Médio - BNCC	14
2.4	Scratch	16
2.4.1	Diferença entre Versões do Scratch	17
2.4.1.1	Scratch Versão 3.0	19
2.4.2	Avaliação de Projetos — Dr. Scratch	21
3	Metodologia Proposta	24
3.1	Objetivos do Curso	24
3.2	Conteúdo Programático e Carga Horária do Curso	25
3.3	Metodologia de Ensino do Curso	26
3.4	Avaliação do Curso	28
3.5	Planos de Aula do Curso	28
4	Relato de Experiências - Aplicações do Curso	32
4.1	Primeira Aplicação do Curso	33
4.1.1	Introdução e Contextualização	33
4.1.2	Relato de Experiência	33
4.1.3	Resultados e Discussões	40

4.1.4	Lições Aprendidas	44
4.2	Segunda Aplicação do Curso	44
4.2.1	Centro de Ensino Médio da Asa Norte - CEAN	44
4.2.1.1	Introdução e Contextualização	44
4.2.1.2	Relato de Experiência	45
4.2.1.3	Resultados e Discussões	49
4.2.1.4	Lições Aprendidas	58
4.2.2	Centro de Ensino Médio Paulo Freire - CEMPF	58
4.2.2.1	Introdução e Contextualização	58
4.2.2.2	Relato de Experiência	58
4.2.2.3	Resultados e Discussões	62
4.2.2.4	Lições Aprendidas	67
4.3	Resultados, Discussões e Lições Aprendidas das Experiências	68
5	Conclusões e Trabalhos futuros	72
	Referências	73
	Apêndice	76
A	Planos de aula: Aprenda Pensamento Computacional e Scratch	77
B	Formulário de Inscrição do Curso no Centro de Ensino Médio 01 do Paranoá	93
C	Cartaz de Divulgação do Curso no Centro de Ensino Médio 01 do Paranoá	96
D	Formulário de Inscrição do Curso no Centro de Ensino Médio da Asa Norte	98
E	Cartaz de divulgação do Curso no Centro de Ensino Médio da Asa Norte	101
F	Cartaz de divulgação do Curso no Centro de Ensino Médio Paulo Freire	103
G	Formulário de Caracterização do Perfil dos Estudantes	105
H	Formulário de Avaliação Final do Curso	112

Lista de Figuras

2.1	Competência gerais da BNCC para o Ensino médio. BNCC [1].	7
2.2	Pilares do Pensamento Computacional. Baseado em “ <i>Computational Thinking for Problem Solving</i> ” [2].	9
2.3	Estrutura da redação dissertativa-argumentativa do ENEM. Adaptado de “ <i>A redação do ENEM 2020 - Cartilha do Participante</i> ”, INEP [3].	10
2.4	Exemplo do reconhecimento de padrões. Adaptado do Curso “ <i>Computational thinking for problem solving</i> ” [2].	11
2.5	Exemplo de abstração e representação de dados. Elaborado pelo Autor, 2022.	12
2.6	Exemplo de algoritmo. Elaborado pelo Autor, 2022.	12
2.7	Nuvem de palavras da BNCC [1]. Elaborado pelo Autor, 2022.	14
2.8	Tela do Scratch 1.4. Scratch Wiki [4].	18
2.9	Tela do Scratch 2.0. Scratch Wiki [4].	18
2.10	Tela do Scratch 3.0. Scratch Wiki [4].	19
2.11	Tela do Editor Scratch 3.0. Scratch, 2022.	21
3.1	Exemplo da estrutura do plano de aula do curso intitulado “Aprenda Pensamento Computacional e Scratch” - Parte 1. Elaborado pelo Autor, 2022.	29
3.2	Exemplo da estrutura do plano de aula do curso intitulado “Aprenda Pensamento Computacional e Scratch” - Parte 2. Elaborado pelo Autor, 2022.	30
3.3	Exemplo da estrutura do plano de aula do curso intitulado “Aprenda Pensamento Computacional e Scratch” - Parte 3. Elaborado pelo Autor, 2022.	31
4.1	Histórico de presença dos estudantes no curso - CEM 01 do Paranoá. Elaborado pelo Autor, 2022.	40
4.2	Telas dos projetos desenvolvidos pelos estudantes do CEM 01 do Paranoá. Elaborado pelo Autor, 2022.	42
4.2.1	Projeto “Labirinto”. Estudante A.	42
4.2.2	Projeto “Meteorite Destroyer”. Estudante B.	42
4.2.3	Projeto “Gula sem fim”. Estudante C.	42

4.3	Tela das avaliações dos projetos desenvolvidos pelos estudantes do CEM 01 do Paranoá. Elaborado pelo Autor, 2022.	43
4.3.1	Tela de avaliação do projeto “Labirinto”. Estudante A.	43
4.3.2	Tela de avaliação do projeto “Meteorite Destroyer”. Estudante B.	43
4.3.3	Tela de avaliação do projeto “Gula sem fim”. Estudante C.	43
4.4	Histórico de presença dos estudantes no curso - CEAN. Elaborado pelo Autor, 2022.	50
4.5	Perfil dos estudantes do CEAN. Elaborado pelo Autor, 2022.	51
4.5.1	Possui computador?	51
4.5.2	Possui telefone celular?	51
4.5.3	Possui <i>internet</i> móvel?	51
4.5.4	Possui <i>tablet</i> ?	51
4.5	Continuação: Perfil dos estudantes do CEAN. Elaborado pelo Autor, 2022.	52
4.5.5	Possui <i>internet</i> em casa?	52
4.5.6	Possui conhecimento em programação?	52
4.5.7	Se considera bom em raciocínio lógico?	52
4.5.8	Qual disciplina mais gosta ou tem facilidade de aprender?	52
4.5.9	Qual disciplina menos gosta ou tem dificuldade de aprender?	52
4.6	Telas dos projetos dos estudantes do CEAN. Elaborado pelo Autor, 2022.	54
4.6.1	Projeto “Chaotic Space”.	54
4.6.2	Projeto “Rocket and Rocks War”.	54
4.6.3	Projeto “Dinossauro pula cacto”.	54
4.6.4	Projeto “Pega fruta”	54
4.6.5	Projeto “Sereia e tubarão”	54
4.7	Avaliação dos projetos dos estudantes do CEAN no Dr. Scratch. Elaborado pelo Autor, 2022.	55
4.7.1	Avaliação Dr. Scratch - Projeto “Chaotic Space”.	55
4.7.2	Avaliação Dr. Scratch - Projeto “Rocket and Rocks War”.	55
4.7.3	Avaliação Dr. Scratch - Projeto “Dinossauro pula cacto”.	55
4.7.4	Avaliação Dr. Scratch - Projeto “Pega fruta”.	55
4.7.5	Avaliação Dr. Scratch - Projeto “Sereia e tubarão”.	55

4.8	Avaliação final dos estudantes - CEAN. Elaborado pelo Autor, 2022.	56
4.8.1	Você gostou do curso?	56
4.8.2	Você considera que a aplicação dos conceitos/temas desse curso em sua vida acadêmica/profissional e pessoal é:	56
4.8.3	Como você avalia o formato e condução (metodologia) das aulas do curso?	56
4.8.4	Como você avalia a forma do professor comunicar e aplicar o conteúdo (didática) nas aulas do curso?	56
4.8.5	Como você avalia o conhecimento que o professor tem sobre o assunto do curso?	56
4.8.6	Como você avalia a sua aprendizagem ao longo do curso?	56
4.9	Comparativo da percepção dos estudantes quanto ao nível de conhecimento em informática antes e depois do curso - CEAN. Elaborado pelo Autor, 2022.	57
4.10	Histórico de presença dos estudantes no curso - CEMPF. Elaborado pelo Autor, 2022.	62
4.11	Perfil dos estudantes do CEMPF. Elaborado pelo Autor, 2022.	63
4.11.1	Possui computador?	63
4.11.2	Possui telefone celular?	63
4.11.3	Possui <i>internet</i> móvel?	63
4.11.4	Possui <i>tablet</i> ?	63
4.11.5	Possui <i>internet</i> em casa?	63
4.11	Continuação: Perfil dos estudantes do CEMPF. Elaborado pelo Autor, 2022.	64
4.11.6	Possui conhecimento em programação?	64
4.11.7	Se considera bom em raciocínio lógico?	64
4.11.8	Qual disciplina mais gosta ou tem facilidade de aprender?	64
4.11.9	Qual disciplina menos gosta ou tem dificuldade de aprender?	64
4.12	Telas dos projetos dos estudantes do CEAN. Elaborado pelo Autor, 2022. .	65
4.12.1	Projeto “Caixa Eletrônico”.	65
4.12.2	Projeto “Coruja e Varinha”.	65
4.13	Avaliação dos projetos dos estudantes do CEAN no Dr. Scratch. Elaborado pelo Autor, 2022.	65
4.13.1	Avaliação Dr. Scratch - Projeto “Caixa Eletrônico”.	65
4.13.2	Avaliação Dr. Scratch - Projeto “Coruja e Varinha”.	65

4.14	Avaliação final dos estudantes - CEAN. Elaborado pelo Autor, 2022.	66
4.14.1	Você gostou do curso?	66
4.14.2	Você considera que a aplicação dos conceitos/temas desse curso em sua vida acadêmica/profissional e pessoal é:	66
4.14.3	Como você avalia o formato e condução (metodologia) das aulas do curso?	66
4.14.4	Como você avalia a forma do professor comunicar e aplicar o conteúdo (didática) nas aulas do curso?	66
4.14.5	Como você avalia o conhecimento que o professor tem sobre o assunto do curso?	66
4.14.6	Como você avalia a sua aprendizagem ao longo do curso?	66
4.15	Comparativo da percepção dos estudantes quanto ao nível de conhecimento em informática antes e depois do curso - CEMPF. Elaborado pelo Autor, 2022.	67
4.16	Pontuação por projeto por escola - CEAN e CEMPF. Elaborado pelo Autor, 2022.	70
4.17	Comparação dos projetos por habilidade analisada no Dr. Scratch. Elaborado pelo Autor, 2022.	70

Lista de Tabelas

2.1	Conceitos e definições Pensamento Computacional. Adaptado e traduzido de “ <i>Computational Thinking in K-12 Education: teacher resources</i> ” de CSTA e ISTE [5].	8
2.2	Interface e funcionalidades do Scratch 3.0. Adaptado de Scratch Wiki [4]. .	20
2.3	Conceitos e nível de competência Dr. Scratch. Traduzido de “ <i>Dr. Scratch: Automatic Analysis of Scratch Projects to Assess and Foster Computational Thinking</i> ”, Moreno-León, Robles e Román-González [6].	23
3.1	Conteúdo programático e carga horária do curso “Aprenda Pensamento Computacional e Scratch”. Adaptado de “ <i>Computational Thinking for Problem Solving</i> ”, Davidson e Murphy [2] e “Utilização do Scratch como Ferramenta no Ensino de Pensamento Computacional para Crianças”, Aono <i>et al.</i> [7].	27
4.1	Diferenças entre a primeira e segunda aplicação do curso. Elaborado pelo Autor, 2022.	33
4.2	Percepções entre as diferenças das modalidades de ensino das experiências. Elaborado pelo Autor, 2022.	69

Lista de Abreviaturas e Siglas

BNCC Base Nacional Comum Curricular.

CIC Departamento de Ciência da Computação.

DEX Decanato de Extensão.

EBB Educação Básica Brasileira.

GDF Governo do Distrito Federal.

LDB Lei de Diretrizes e Bases da Educação.

LPV Linguagem de Programação Visual.

MIT *Massachusetts Institute of Technology*.

PC Pensamento Computacional.

PIBEX Programa Institucional de Bolsas de Extensão.

PNE Plano Nacional de Educação.

SBC Sociedade Brasileira de Computação.

TDIC Tecnologias Digitais de Informação e Comunicação.

UnB Universidade de Brasília.

Capítulo 1

Introdução

Wing [8] afirmou que o Pensamento Computacional (PC) é “um conjunto de habilidades, universalmente aplicável, que todos, não apenas cientistas da computação, podem aprender e devem usar, pois é um processo para resolver problemas, projetar sistemas e compreender comportamentos a partir de ferramentas e técnicas sólidas, baseadas nos conceitos que fundamentam a Ciência da Computação”. Wing [8] enfatiza que “pensar como um cientista da computação significa mais do que ser capaz de programar um computador. Requer pensar em vários níveis de abstração”.

A Base Nacional Comum Curricular (BNCC) [1] faz parte da política nacional da Educação Básica Brasileira (EBB), sendo fundamental, pois contribui para a elaboração de políticas e ações em todos os níveis (Federal, Estadual e Municipal), além de contribuir para a formação e avaliação de professores, para a elaboração de conteúdos educacionais e para os critérios de oferta de infraestrutura adequada para o pleno desenvolvimento da educação.

A informática e a computação são mencionadas como áreas do conhecimento em diversas competências e habilidades da BNCC para o Ensino Médio, reforçando a necessidade de esse tema ser tratado por todos e para todos durante essa etapa de formação.

Diante da necessidade de inserir a informática educacional nas escolas, Vieira e Sabbatini [9] afirmam que:

A informática educacional nas escolas é essencial para o fortalecimento da autonomia dos alunos, dando espaço para a inovação e criação do conhecimento. Ter domínio do uso das tecnologias é fundamental para que se tenha um protagonismo na sociedade contemporânea, tanto como mero usuário receptor das informações, como sujeito transformador [e] produtor de conhecimento.

Lye e Koh [10] concluem que as Linguagens de Programação Visual (LPVs) são mais adequadas para inserir o Pensamento Computacional na educação básica, visto que diminuem a carga cognitiva exigida pela sintaxe das linguagens de programação tradicionais, permitindo aos estudantes se concentrar na lógica necessária para a resolução do problema.

O Scratch é uma Linguagem de Programação Visual (LPV) cuja função é tornar o aprendizado de programação mais fácil, interativo e divertido. Foi desenvolvido pelo grupo de pesquisa “*Lifelong Kindergarten*” no *Media Lab* do *Massachusetts Institute of Technology* (MIT). Apesar de ter sido projetado especialmente para jovens entre 8 e 16 anos, ele também tem sido utilizado por pessoas de todas as idades. No Scratch basta, conectar blocos de maneira lógica e interativa, formando a programação do ambiente para criar histórias, animações, jogos, simuladores, músicas e arte [11].

Avila *et al.* [12] afirma que:

Experiências de ensino que trabalham conceitos fundamentais da computação para desenvolver o PC, em geral, são efetivados por meio de cursos e/ou oficinas que envolvem programação de computadores (algoritmos), animações e jogos.

Este trabalho cria uma metodologia (curso) para ensino do Pensamento Computacional nas escolas públicas de Ensino Médio no Distrito Federal. Para o trabalho proposto as questões norteadoras são:

- O que é o Pensamento Computacional?
- Como o Pensamento Computacional está inserido na Base Nacional Comum Curricular para o Ensino Médio?
- Qual a melhor metodologia e ferramenta para ensinar o Pensamento Computacional no Ensino Médio?
- Como avaliar o Pensamento Computacional no Scratch?

1.1 Objetivos

O objetivo geral deste trabalho consiste em propor uma metodologia para inserir o Pensamento Computacional nas escolas públicas de Ensino Médio no Distrito Federal.

Os objetivos específicos são os seguintes:

- Aplicar a metodologia proposta em escolas públicas de Ensino Médio no Distrito Federal;
- Relatar as experiências das aplicações da metodologia;
- Analisar, discutir e avaliar os dados coletados e os resultados obtidos;
- Avaliar a metodologia e propor melhorias e trabalhos futuros.

1.2 Metodologia

A metodologia deste trabalho envolve: um levantamento bibliográfico exploratório para embasar este trabalho com fundamentação teórica, estudos de casos e relatos de experiências; a proposição de uma metodologia fundamentada para o ensino do Pensamento Computacional em escolas públicas de Ensino Médio no Distrito Federal, e a aplicação da metodologia e a coleta e análise dos dados para discussões e avaliações dos resultados obtidos.

1.3 Resultados

Os seguintes resultados foram alcançados após o desenvolvimento deste trabalho:

- Uma metodologia que insere o Pensamento Computacional no Ensino Médio público do Distrito Federal;
- Uma metodologia (curso) replicável;
- A aplicação da metodologia;
- O relato das três experiências das aplicações da metodologia;
- A análise dos dados coletados e resultados das aplicações da metodologia.

1.4 Organização do Trabalho

O este trabalho está organizado da seguinte forma:

- No Capítulo 2 é apresentada a fundamentação teórica;
- No Capítulo 3 são apresentados a fundamentação, a estruturação e o detalhamento da metodologia em formato de curso;
- No Capítulo 4 são detalhadas as experiências das aplicações do curso, bem como os resultados, discussões e lições aprendidas;
- No Capítulo 5 é feita a conclusão deste trabalho e as ideias para trabalhos futuros são apresentadas.

Capítulo 2

Fundamentação Teórica

Neste capítulo são apresentados os conceitos que fundamentam este trabalho. Na Seção 2.1 é apresentada a Base Nacional Comum Curricular (BNCC). Na Seção 2.2 é apresentado o Pensamento Computacional (PC), seus conceitos e pilares. Na Seção 2.3 é apresentada a inserção do PC na Educação Básica Brasileira (EBB), segundo a BNCC e, por fim, na Seção 2.4 é abordado o Scratch como ferramenta para o ensino do PC.

2.1 Base Nacional Comum Curricular (BNCC)

A BNCC [1] é:

Um documento de caráter normativo que define o conjunto orgânico e progressivo de aprendizagens essenciais que todos os alunos devem desenvolver ao longo das etapas e modalidades da Educação Básica, de modo que tenham assegurados seus direitos de aprendizagem e desenvolvimento, em conformidade com o que preceitua o Plano Nacional de Educação (PNE).

As aprendizagens essenciais devem assegurar aos estudantes o desenvolvimento de dez competências gerais. Competência é definida na BNCC [1] como “mobilização de conhecimentos (conceitos e procedimentos), habilidades (práticas, cognitivas e socioemocionais), atitudes e valores para resolver demandas complexas da vida cotidiana, do pleno exercício da cidadania e do mundo do trabalho”.

A BNCC [1], tendo seu foco no desenvolvimento das competências, indica que as decisões pedagógicas precisam considerar o “saber” e o “saber fazer”; respectivamente, eles constituem-se pelos conhecimentos, habilidades, atitudes e valores, além da mobilização deles para resolver demandas complexas da vida cotidiana, do pleno exercício da cidadania e do mundo do trabalho.

As dez competências a seguir estão descritas na BNCC [1] e fundamentam todo o documento. Elas devem ser alcançadas ao longo da EBB em cada etapa da escolaridade:

1. Valorizar e utilizar os conhecimentos historicamente construídos sobre o mundo físico, social, cultural e digital para entender e explicar a realidade, continuar aprendendo e colaborar para a construção de uma sociedade justa, democrática e inclusiva.
2. Exercitar a curiosidade intelectual e recorrer à abordagem própria das ciências, incluindo a investigação, a reflexão, a análise crítica, a imaginação e a criatividade, para investigar causas, elaborar e testar hipóteses, formular e resolver problemas e criar soluções (inclusive tecnológicas) com base nos conhecimentos das diferentes áreas.
3. Valorizar e fruir as diversas manifestações artísticas e culturais, das locais às mundiais, e também participar de práticas diversificadas da produção artístico-cultural.
4. Utilizar diferentes linguagens — verbal (oral ou visual-motora, como Libras, e escrita), corporal, visual, sonora e digital —, bem como conhecimentos das linguagens artística, matemática e científica, para se expressar e partilhar informações, experiências, ideias e sentimentos em diferentes contextos e produzir sentidos que levem ao entendimento mútuo.
5. Compreender, utilizar e criar Tecnologias Digitais de Informação e Comunicação (TDIC) de forma crítica, significativa, reflexiva e ética nas diversas práticas sociais (incluindo as escolares) para se comunicar, acessar e disseminar informações, produzir conhecimentos, resolver problemas e exercer protagonismo e autoria na vida pessoal e coletiva.
6. Valorizar a diversidade de saberes e vivências culturais e apropriar-se de conhecimentos e experiências que lhe possibilitem entender as relações próprias do mundo do trabalho e fazer escolhas alinhadas ao exercício da cidadania e ao seu projeto de vida, com liberdade, autonomia, consciência crítica e responsabilidade.
7. Argumentar com base em fatos, dados e informações confiáveis, para formular, negociar e defender ideias, pontos de vista e decisões comuns que respeitem e promovam os direitos humanos, a consciência socioambiental e o consumo responsável em âmbito local, regional e global, com posicionamento ético em relação ao cuidado de si mesmo, dos outros e do planeta.
8. Conhecer-se, apreciar-se e cuidar de sua saúde física e emocional, compreendendo-se na diversidade humana e reconhecendo suas emoções e as dos outros, com auto-crítica e capacidade para lidar com elas.
9. Exercitar a empatia, o diálogo, a resolução de conflitos e a cooperação, fazendo-se respeitar e promovendo o respeito ao outro e aos direitos humanos, com acolhimento e valorização da diversidade de indivíduos e de grupos sociais, seus saberes, identidades, culturas e potencialidades, sem preconceitos de qualquer natureza.
10. Agir pessoal e coletivamente com autonomia, responsabilidade, flexibilidade, resiliência e determinação, tomando decisões com base em princípios éticos, democráticos, inclusivos, sustentáveis e solidários.

Observa-se que em diversos itens são mencionados tanto a questão “digital”, quanto a necessidade de se “formular e resolver problemas e criar soluções (inclusive tecnológicas)” e “Compreender, utilizar e criar Tecnologias Digitais de Informação e Comunicação”. Logo, percebe-se a necessidade do letramento e da inclusão digital dos estudantes para que eles

consigam alcançar as competências em sua plenitude, fazendo com que a dimensão digital também precise ser incluída nas práticas educacionais em todo território nacional.

2.1.1 Base Nacional Comum Curricular — Ensino Médio

A Lei de Diretrizes e Bases da Educação (LDB) [13] define que o Ensino Médio possui as seguintes finalidades:

- I — a consolidação e o aprofundamento dos conhecimentos adquiridos no ensino fundamental, possibilitando o prosseguimento de estudos;
- II — a preparação básica para o trabalho e a cidadania do educando, para continuar aprendendo, de modo a ser capaz de se adaptar com flexibilidade a novas condições de ocupação ou aperfeiçoamento posteriores;
- III — o aprimoramento do educando como pessoa humana, incluindo a formação ética e o desenvolvimento da autonomia intelectual e do pensamento crítico;
- IV — a compreensão dos fundamentos científico-tecnológicos dos processos produtivos, relacionando a teoria com a prática, no ensino de cada disciplina.

A BNCC [1] define a etapa do Ensino Médio como:

A etapa final da Educação Básica, direito público subjetivo de todo cidadão brasileiro. Todavia, a realidade educacional do País tem mostrado que essa etapa representa um gargalo na garantia do direito à educação. Para além da necessidade de universalizar o atendimento, tem-se mostrado crucial garantir a permanência e as aprendizagens dos estudantes, respondendo às suas demandas e aspirações presentes e futuras. Com o objetivo de consolidar, aprofundar e ampliar a formação integral, atende às finalidades dessa etapa e contribui para que os estudantes possam construir e realizar seu projeto de vida, em consonância com os princípios da justiça, da ética e da cidadania.

Para o pleno atendimento das finalidades do Ensino Médio, a BNCC [1] estrutura o currículo dessa etapa em áreas do conhecimento (Linguagens e suas Tecnologias, Matemática e suas Tecnologias, Ciências da Natureza e suas Tecnologias, Ciências Humanas e Sociais Aplicadas e formação técnica e profissional), conforme esquema da Figura 2.1 e de acordo com o que determina a LDB. Ela reforça a necessidade do relacionamento entre as disciplinas e da contextualização para compreensão e intervenção na realidade, requerendo trabalho conjunto dos professores no planejamento e na execução dos planos de ensino das suas respectivas áreas [1].

As competências e habilidades da BNCC [1] para cada área do conhecimento constituem a formação geral básica. Os currículos do Ensino Médio são compostos por ela e articulados aos itinerários formativos na totalidade indissociável.



Figura 2.1: Competência gerais da BNCC para o Ensino médio. BNCC [1].

2.2 Pensamento Computacional

PC é definido por Wing [14] como “processos de pensamento envolvidos na formulação de problemas e suas soluções, sendo que as soluções devem ser representadas de forma que possa ser realizada por agentes de processamento de informações”.

CTA e ISTE [5] afirmam que o PC é um processo de resolução de problemas que inclui (mas não se limita) as seguintes características:

- Formular problemas de uma forma que nos permita usar um computador e outras ferramentas para ajudar a resolvê-los;
- Organizar e analisar lógica de dados;
- Representar dados por meio de abstrações, com modelos e simulações;
- Automatizar soluções por meio de pensamento algorítmico (por uma série de etapas ordenadas);
- Identificar, analisar e implementar possíveis soluções com o objetivo de alcançar a combinação mais eficiente e eficaz de etapas e recursos;
- Generalizar e transferir esse processo de resolução de problemas para uma ampla variedade de problemas.

E que essas habilidades são apoiadas e aprimoradas pelas seguintes disposições ou atitudes, dimensões essenciais para o PC:

- Confiança em lidar com a complexidade;
- Persistência em trabalhar com problemas difíceis;
- Tolerância em relação a ambiguidades;

Tabela 2.1: Conceitos e definições Pensamento Computacional. Adaptado e traduzido de “*Computational Thinking in K-12 Education: teacher resources*” de CSTA e ISTE [5].

Conceito	Definição
Coleta de dados	Coletar informações apropriadas
Análise de dados	Dar sentido aos dados, encontrar padrões e tirar conclusões
Representação de dados	Descrever e organizar dados em gráficos, tabelas, palavras ou imagens apropriadas
Decomposição de problema	Dividir as tarefas em partes menores e gerenciáveis
Abstração	Reduzir a complexidade para definir a ideia principal
Algoritmos e procedimentos	Série de etapas ordenadas tomadas para resolver um problema ou alcançar algum fim
Automação	Ter computadores ou máquinas fazendo tarefas repetitivas, ou tediosas
Simulação	Representação ou modelo de um processo. A simulação também envolve a execução de experimentos usando modelos
Paralelismo	Organizar recursos para realizar tarefas simultaneamente e alcançar um objetivo comum

- Capacidade de lidar com problemas em aberto;
- Capacidade de se comunicar e trabalhar com outras pessoas para alcançar um objetivo ou solução comum. [5].

Vários conceitos e definições das habilidades do PC são essenciais para seu desenvolvimento, como: coleta de dados, análise de dados, representação de dados, decomposição de problemas, abstração, algoritmo e procedimentos, automação, simulação e paralelismo. Esses conceitos foram definidos por CSTA e ISTE [5] e estão detalhados na Tabela 2.1.

Apesar de os seguintes conceitos já terem sido definidos: decomposição, reconhecimento de padrões, abstração e representação de dados e algoritmo, eles são considerados os quatro pilares que fundamentam o PC, ou seja, eles não são somente aprendidos durante a construção do PC como o fundamentam.

2.2.1 Pilares do Pensamento Computacional

O PC é dividido em quatro estágios/passos: decomposição, reconhecimento de padrões, abstração e representação de dados e algoritmo. Esses pilares são considerados essenciais para que um problema, computacional ou não, seja resolvido por qualquer pessoa e em qualquer área utilizando o PC. Os quatro pilares, conforme ilustrado na Figura 2.2, são conceituados a seguir.

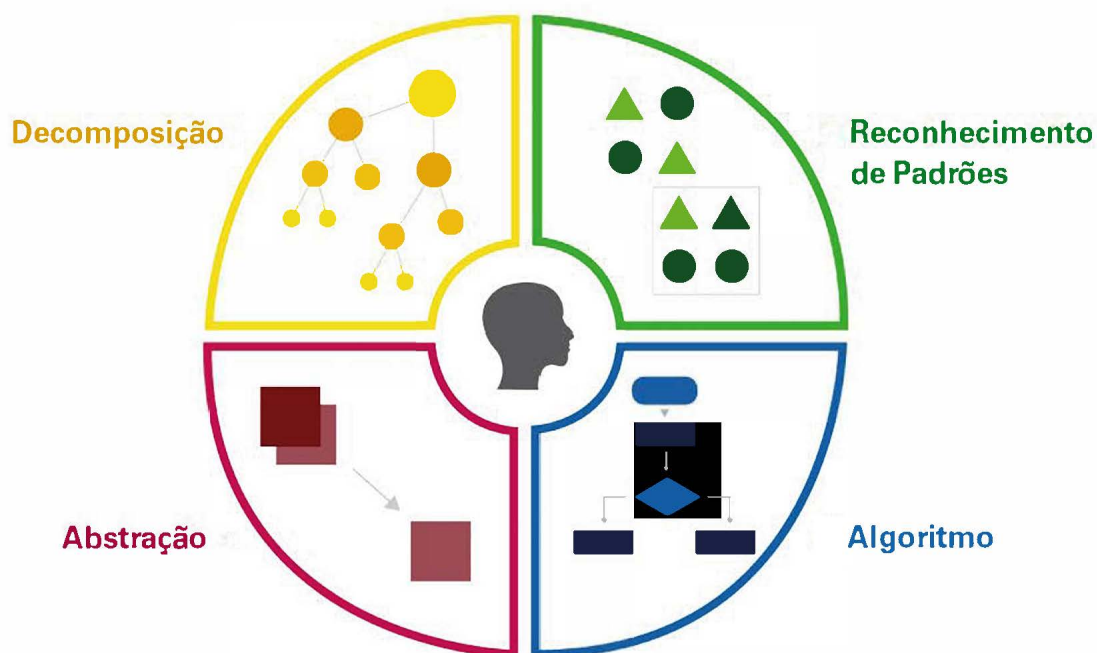


Figura 2.2: Pilares do Pensamento Computacional. Baseado em “*Computational Thinking for Problem Solving*” [2].

Decomposição: identificar um problema complexo e dividi-lo em problemas menores para ficar mais fácil resolvê-lo, ou seja, as soluções para os subproblemas fornecem uma solução para o problema principal e complexo [15] [2] [16].

Um exemplo de decomposição de um problema maior e complexo em problemas menores e mais simples é a escrita de um artigo científico [2] ou de uma redação do Exame Nacional do Ensino Médio — ENEM. A prova de redação exige a produção de um texto em prosa, do tipo dissertativo-argumentativo, sobre um tema dado pela banca organizadora. O problema maior consiste na construção de todo o texto; porém, primeiro deve se escrever a ideia geral (introdução); depois, defendê-la com argumentos consistentes (desenvolvimento) e, por fim, propor uma intervenção social (conclusão) para o problema apresentado no desenvolvimento do texto [3], conforme ilustrado na Figura 2.3.

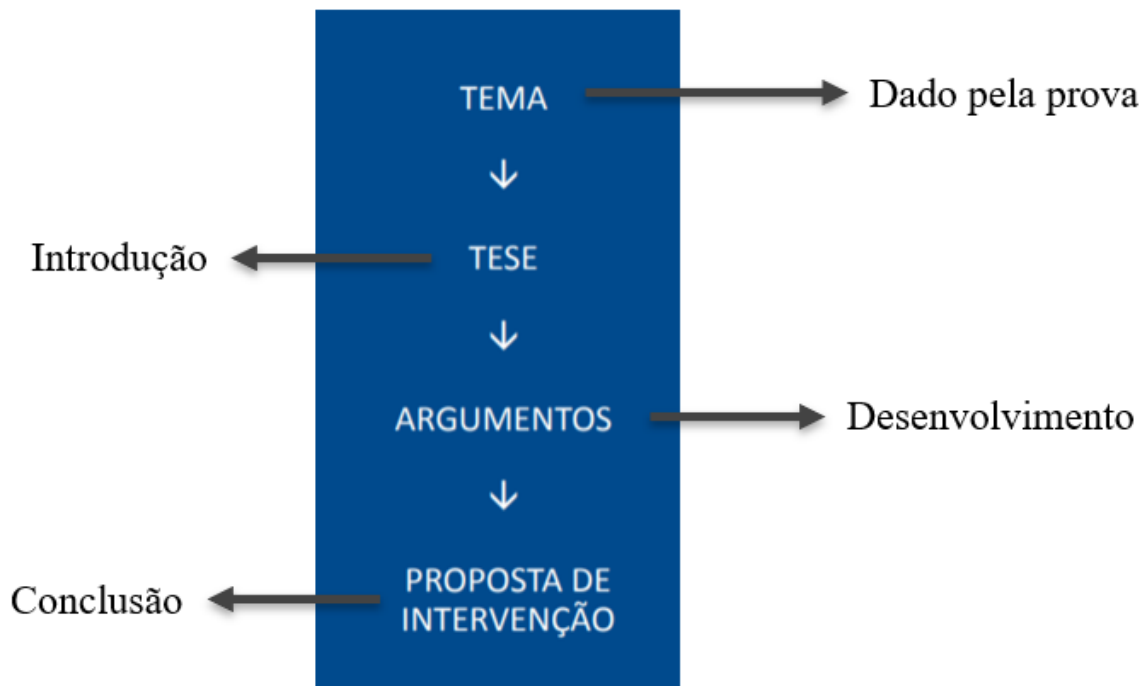


Figura 2.3: Estrutura da redação dissertativa-argumentativa do ENEM. Adaptado de “A redação do ENEM 2020 - Cartilha do Participante”, INEP [3].

Reconhecimento de padrões: consiste em encontrar as semelhanças ou características compartilhadas entre os problemas; ou seja, cada um desses problemas menores pode ser analisado individualmente e com maior profundidade a partir da identificação de problemas semelhantes. Assim, a mesma solução pode ser usada para cada ocorrência do padrão [15] [2] [16].

Um exemplo do reconhecimento de padrões se dá quando se desenha um cachorro. Todos os cães, independentemente da raça, compartilham características em comum, como o fato de terem: uma cabeça, duas orelhas, um corpo, uma cauda, quatro pernas e quatro patas. O que irá diferenciar uma raça de outra são as características específicas, como o corpo ou pernas mais alongados em relação aos das outras raças, ou a cabeça em formato um pouco diferente, conforme ilustrado na Figura 2.4.

Abstração e representação de dados: consiste em identificar os princípios básicos que geram padrões perceptíveis; ou seja, em determinar quais características do problema são importantes e em filtrar aquelas que não o são. Ou, ainda, em usá-los para criar uma representação do que se está tentando resolver apenas conforme os detalhes que são importantes, enquanto as informações irrelevantes são ignoradas [15] [2] [16].

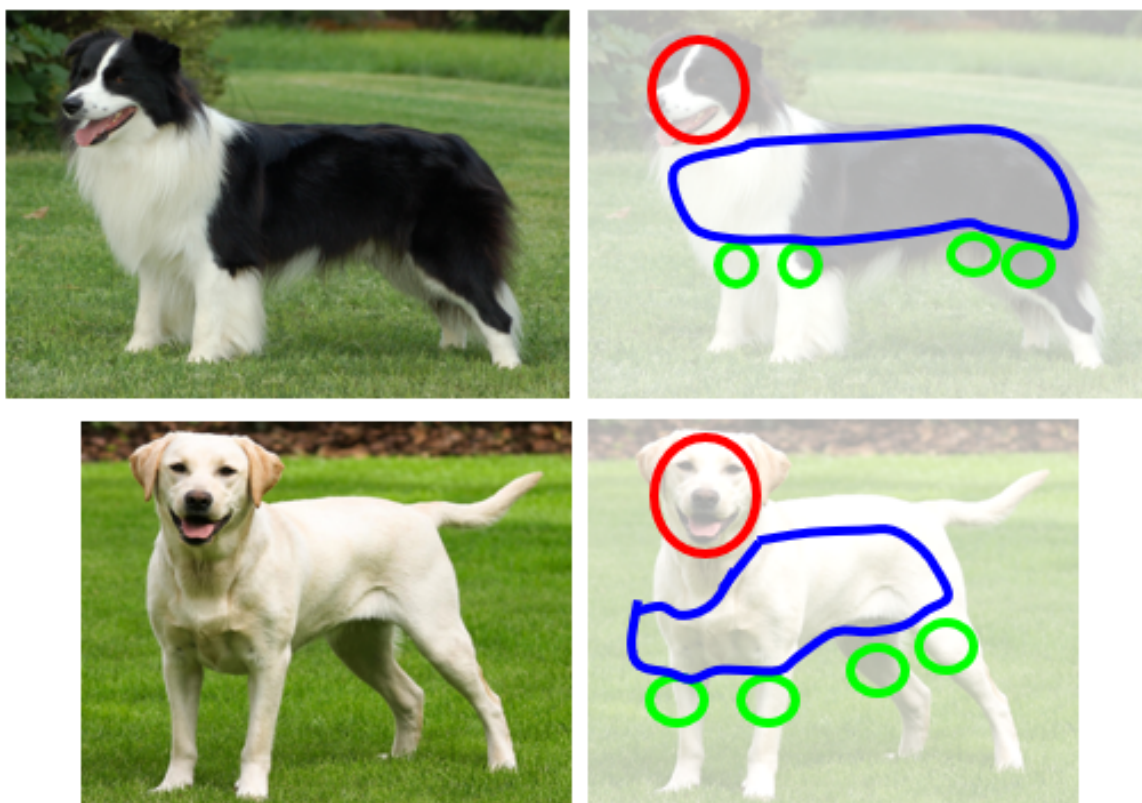


Figura 2.4: Exemplo do reconhecimento de padrões. Adaptado do Curso “*Computational thinking for problem solving*” [2].

Um exemplo de abstração e representação de dados consiste na criação de um quadro com as informações das aulas sobre cada turma. Na Figura 2.5, há duas cores que representam os dados considerados importantes ou desimportantes para o problema que foi colocado. Se o problema for o de criar um quadro para os estudantes se organizarem e saberem quais aulas eles terão, apenas os dados da esquerda (azuis) serão necessários; se o problema for construir um quadro das aulas para planejamento e controle da coordenação da escola, os dados da direita (amarelos) também são importantes. Essa análise traz sempre a reflexão sobre o que se cogita resolver em relação a cada problema.

Algoritmo: desenvolver instruções, regras ou passo a passo para resolver os subproblemas e, conseqüentemente, o problema; ou seja, identificar o que deve ser feito e a ordem onde as instruções devem ser feitas [15] [2] [16].

Para representar o algoritmo e para descrever as instruções que devem ser realizadas a fim de se resolver o problema, podem ser utilizadas diversas linguagens, computacionais ou não, como: descrição narrativa, fluxogramas, pseudocódigos e linguagens de programação para implementar o algoritmo e resolver o problema [17].

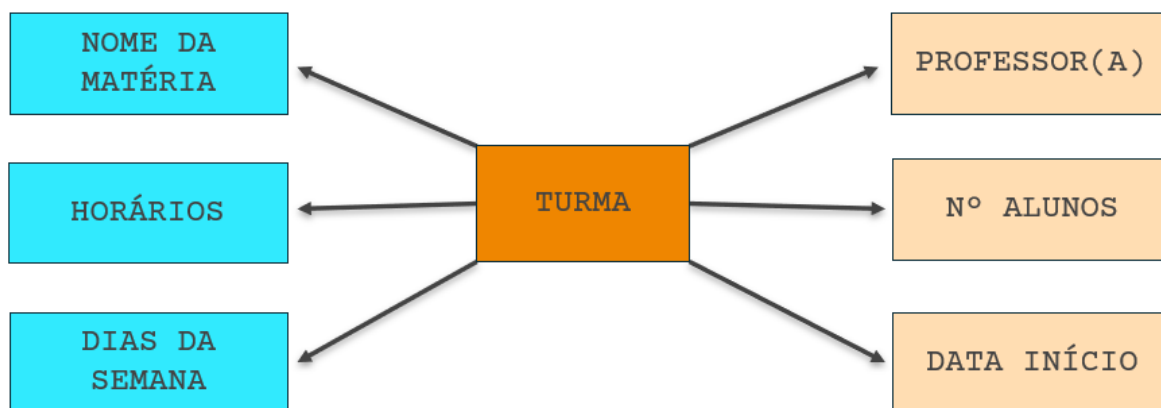


Figura 2.5: Exemplo de abstração e representação de dados. Elaborado pelo Autor, 2022.

Um exemplo de representação do algoritmo, conforme ilustrado na Figura 2.6, está na pretensão de se resolver problemas e de se identificar números primos. Para que um número seja primo, ele deve ser divisível por um ou por ele mesmo, exclusivamente.

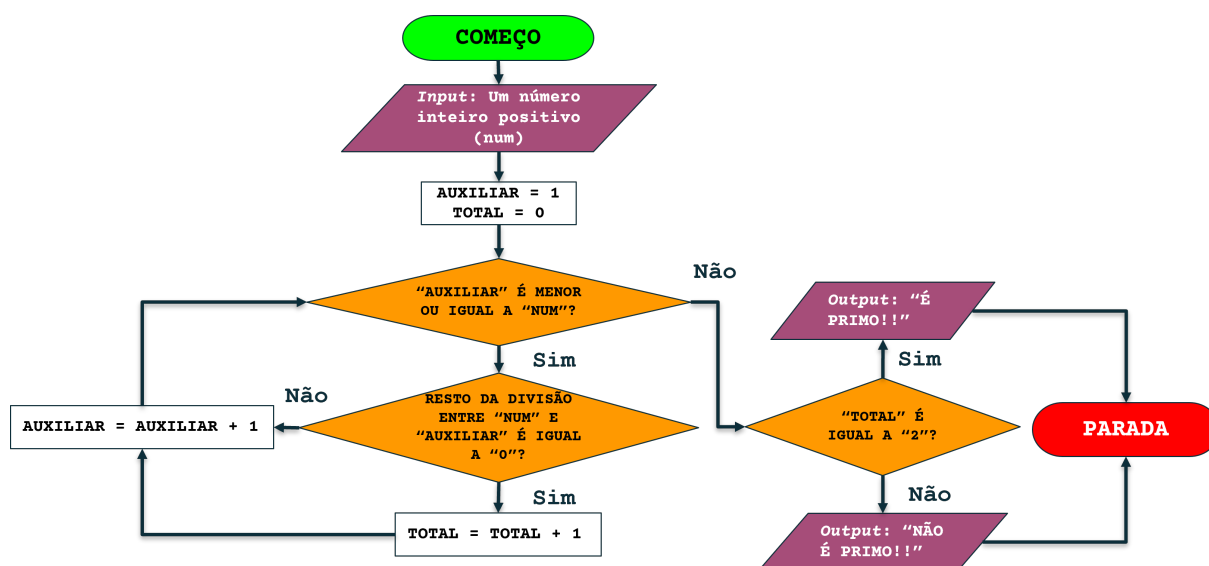


Figura 2.6: Exemplo de algoritmo. Elaborado pelo Autor, 2022.

2.3 Pensamento Computacional na Educação Básica Brasileira

André [18] afirma que “o Pensamento Computacional nas escolas permite que os alunos resolvam problemas, dividam-nos em problemas menores e criem algoritmos para

solucioná-los”.

A BNCC [1] enfatiza que:

É preciso garantir aos jovens aprendizagens para [que eles atuem] em uma sociedade [que está] em constante mudança; prepará-los para profissões que ainda não existem [e para] tecnologias que ainda não foram inventadas e[, ainda,] para resolver problemas que ainda não conhecemos. Certamente, grande parte das futuras profissões envolverá, direta ou indiretamente, computação e tecnologias digitais.

No Brasil, até 2015, as políticas educacionais relacionadas à tecnologia estavam restritas à abordagem de alfabetização e inclusão digital [15]. A BNCC trouxe uma grande mudança no contexto da inclusão digital e do PC na EBB.

Apesar de a BNCC mencionar o “Pensamento Computacional” apenas nove vezes ao longo de suas 595 páginas e, exclusivamente, na área de Matemática, tanto no Ensino Fundamental quanto no Médio, há a menção de palavras e derivações relacionadas à Inclusão Digital e ao PC, como: “identificar”(456 vezes), “desenvolver” (395 vezes), “tecnologia” (333 vezes), “analisar” (323 vezes), “problema” (285 vezes), “informação” (234 vezes), “processos” (226 vezes), “digital” (158 vezes) e outros termos relacionados, conforme ilustrado na Figura 2.7.

Várias competências e habilidades transversais aos conceitos de PC são mencionadas na BNCC, porém apenas na área de Matemática, trazendo a preocupação sobre seu uso como ferramenta mental para resolução de problemas em outras áreas.



Figura 2.7: Nuvem de palavras da BNCC [1]. Elaborado pelo Autor, 2022.

2.3.1 Pensamento Computacional no Ensino Médio - BNCC

O PC está inserido nas competências e habilidades da BNCC [1] para o Ensino Médio, trazendo um tópico específico sobre “As tecnologias digitais e a computação” e reforçando ainda mais a necessidade de abordar tanto a computação quanto as TDIC nessa etapa educacional. Diante disso, a BNCC traz a preocupação com os impactos da constante transformação ocasionada pelas tecnologias na sociedade.

Wing [8] reforça a necessidade de:

Atingir especialmente o público pré-universitário, incluindo professores, pais e alunos, enviando-lhes duas mensagens principais:

- Problemas científicos intelectualmente desafiadores e envolventes ainda precisam ser compreendidos e resolvidos. O domínio do problema e o domínio da solução são limitados apenas por nossa própria curiosidade e criatividade; e
- Eles podem se formar em Ciência da Computação e fazer qualquer coisa. Podem se especializar em Inglês ou Matemática e seguir uma infinidade de carreiras diferentes. Podem, por exemplo, seguir a carreira de Medicina, Direito, Negócios, Política, qualquer tipo de ciência ou engenharia e até artes.

Os jovens estão fortemente, e cada vez mais, inseridos na cultura digital, como consumidores e produtores de novas tecnologias e soluções. Dessa forma, a BNCC [1] enfatiza que as propostas de trabalho precisam priorizar o acesso a saberes do mundo digital e a práticas da cultura digital. Eles impactam o cotidiano dos estudantes dos vários campos de atuação social e despertam tanto o interesse quanto a identificação com as TDIC. O uso das TDIC nas escolas “não só possibilita maior apropriação técnica e crítica desses recursos, como também é determinante para uma aprendizagem significativa e autônoma por parte dos estudantes” [1].

A BNCC [1] tematiza as diferentes dimensões da computação e das TDIC, tanto no que diz respeito a conhecimentos e habilidades quanto no que diz respeito a atitudes e valores, conforme o seguinte:

- **Pensamento computacional:** envolve as capacidades de compreender, analisar, definir, modelar, resolver, comparar e automatizar problemas e suas soluções, de forma metódica e sistemática, por meio do desenvolvimento de algoritmos;
- **Mundo digital:** envolve as aprendizagens relativas às formas de processar, transmitir e distribuir a informação de maneira segura e confiável em diferentes artefatos digitais — tanto físicos (computadores, celulares, tablets, etc.) como virtuais (internet, redes sociais e nuvens de dados, entre outros) —, compreendendo a importância contemporânea de codificar, armazenar e proteger a informação;
- **Cultura digital:** envolve aprendizagens voltadas a uma participação mais consciente e democrática por meio das tecnologias digitais, o que supõe a compreensão dos impactos da revolução digital e dos avanços do mundo digital na sociedade contemporânea, a construção de uma atitude crítica, ética e responsável em relação à multiplicidade de ofertas midiáticas e digitais, aos usos possíveis das diferentes tecnologias e aos conteúdos por elas veiculados, e, também, à fluência no uso da tecnologia digital para expressão de soluções e manifestações culturais de forma contextualizada e crítica.

Na BNCC [1] são definidas as competências e habilidades das diferentes áreas que permitem aos estudantes adquirirem as seguintes capacidades:

- buscar dados e informações de forma crítica nas diferentes mídias, inclusive as sociais, analisando as vantagens do uso e da evolução da tecnologia na sociedade atual, como também seus riscos potenciais;
- apropriar-se das linguagens da cultura digital, dos novos letramentos e dos multiletramentos para explorar e produzir conteúdos em diversas mídias, ampliando as possibilidades de acesso à ciência, à tecnologia, à cultura e ao trabalho;
- usar diversas ferramentas de *software* e aplicativos para compreender e produzir conteúdos em diversas mídias, simular fenômenos e processos [de] diferentes áreas do conhecimento, elaborar e explorar diversos registros de representação matemática; e
- utilizar, propor e/ou implementar soluções (processos e produtos) envolvendo diferentes tecnologias para identificar, analisar, modelar e solucionar problemas complexos em diversas áreas da vida cotidiana, explorando de forma efetiva o raciocínio lógico, o pensamento computacional, o espírito de investigação e a criatividade.

Diante de toda a contextualização sobre a importância do uso e do ensino das tecnologias e da computação para o Ensino Médio, é possível perceber não haver uma definição por área do conhecimento das competências e habilidades para uso e ensino das tecnologias e da computação de maneira geral, visto que a BNCC considera o letramento e a inclusão digital como objetivos transversais do Ensino Médio.

2.4 Scratch

As LPV levam os estudantes a uma forma de pensar computacionalmente baseada em estruturas, como laços de repetição, condicionais e outras bases algorítmicas herdadas das linguagens de programação tradicionais [19].

Bombsar *et al.* [19] mostra que os dados estatísticos entre 2006 e o início de 2015 confirmam o grande interesse da comunidade científica pelo ensino do PC na educação básica (K-12), e mostram que o uso das LPVs foi o mais utilizado nesse período.

O estudo de Avila *et al.* [12], que respondeu algumas questões sobre o PC no Brasil, chegou à conclusão de que o Scratch é a ferramenta mais utilizada em projetos para o desenvolvimento de habilidades e conceitos do PC, seguido do Alice, AppInventor, Lego Mindstorm, CS Unplugged, AgentSheet/AgentCubes, MATLAB, Scalable Game Design, Python e Kodu. Dos trabalhos analisados e que utilizaram o Scratch, concluiu-se que ele pode ser aproveitado de forma satisfatória por estudantes da EBB no desenvolvimento de habilidades e conceitos do PC, que ele viabiliza a aprendizagem significativa, e que os estudantes gostaram e avaliaram positivamente as atividades realizadas nele.

O mapeamento sistemático em metodologias de ensino-aprendizagem de programação para o Ensino Médio de Khouri, Santos e Barbosa [20] chegou à conclusão de que 85% (12)

dos estudos realizam atividades com o uso de computadores, sendo o Scratch a ferramenta mais utilizada.

Considerando a complexidade das linguagens de programação tradicionais para o ensino do PC, Scratch é uma alternativa para os estudantes que nunca tiveram contato com a programação de computadores; assim, seu primeiro contato com a programação será amigável, desmistificando o pensamento de que programar é uma tarefa difícil [21].

Vieira e Sabbatini [9] afirmam que “com Scratch os alunos são envolvidos no uso de pensamento computacional como uma ferramenta para resolver problemas, ensiná-los a usar os conceitos e métodos de programação criando artefatos digitais e manter seu interesse na área de computação”.

O Scratch foi lançado em sua primeira versão estável, 1.0, em 15 de maio de 2007; a versão final da série 1.X, a 1.4, foi lançada em 2 de julho de 2009, sendo sucedida quatro anos depois pelo Scratch 2.0. Atualmente o Scratch está na versão 3.0, lançada em 2 de janeiro de 2019, que veio para modificá-lo significativamente, pois, nela, foi efetuada uma reestruturação por completo, agora em HTML 5 e JavaScript, possibilitando assim o acesso do Scratch a diferentes tipos de dispositivos.

2.4.1 Diferença entre Versões do Scratch

Apesar de as versões 1.X terem sido desenvolvidas e atualizadas gradualmente, a versão 1.4 foi a que ficou mais conhecida por ser a última e mais completa versão [4].

Por se tratar da primeira versão oficialmente disponibilizada, houve mudanças significativas na implementação de novos blocos e funcionalidades, além de grandes mudanças na interface do usuário [4].

A versão 2.0 foi a segunda versão principal do Scratch. Foi a primeira versão que incluiu um editor *online* no *site* Scratch, bem como um editor *offline*. O editor *online* Scratch 2.0 não pode mais ser usado, pois requer o Flash Player, mas o editor *offline* ainda pode ser baixado e utilizado. As mudanças realizadas estavam relacionadas à interface do usuário, aos procedimentos, às clonagens, aos dados na nuvem, aos gráficos vetoriais, à “mochila” para coletar *scripts* e às mídias para o reuso ou para mostrar e ocultar listas e o editor de som [4].

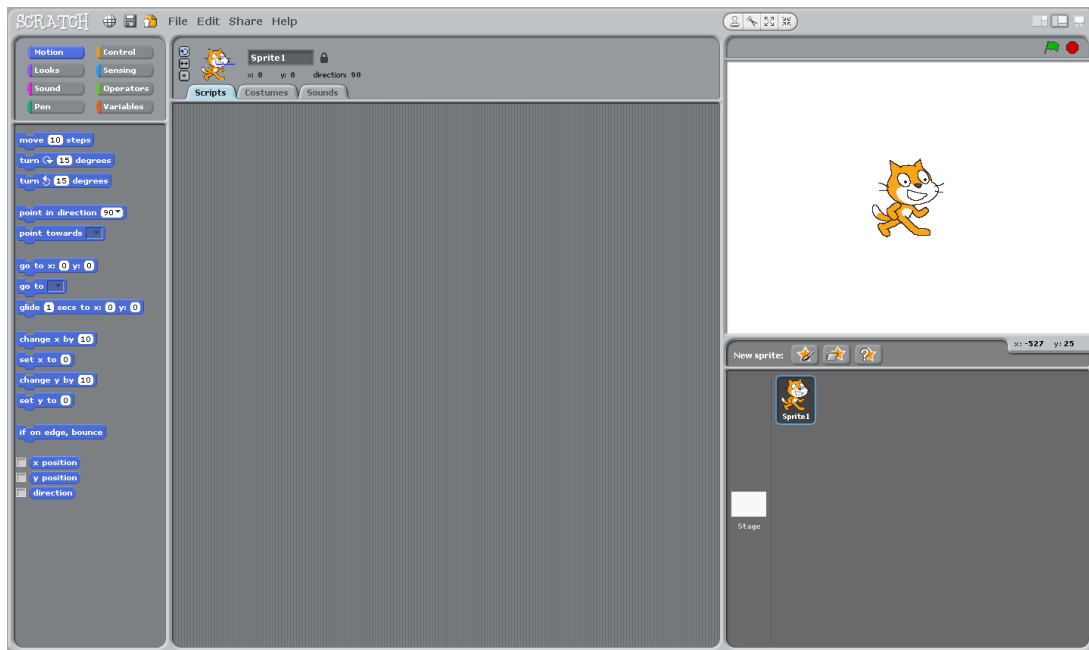


Figura 2.8: Tela do Scratch 1.4. Scratch Wiki [4].

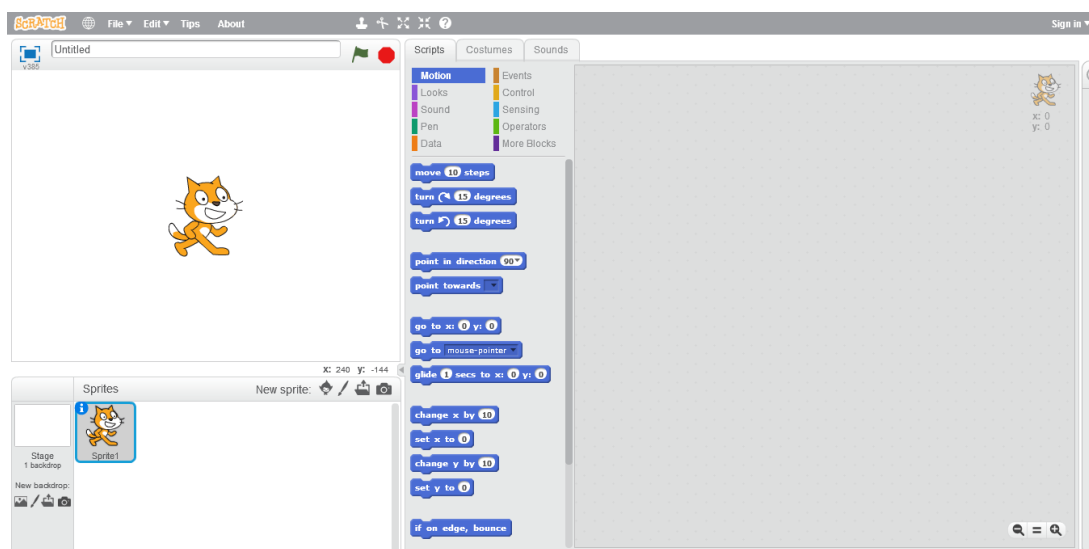


Figura 2.9: Tela do Scratch 2.0. Scratch Wiki [4].

A versão 3.0 foi a versão que mais se destacou e que é utilizada atualmente. As principais mudanças foram relativas ao palco, aos nomes e às funcionalidades de blocos, o editor de pintura e som, o novo site, os blocos de efeitos sonoros, o tratamento de *strings* e listas, a posição do objeto, camadas, trajés, fundo e extensões de música, caneta, vídeo, tradução e outros [4].

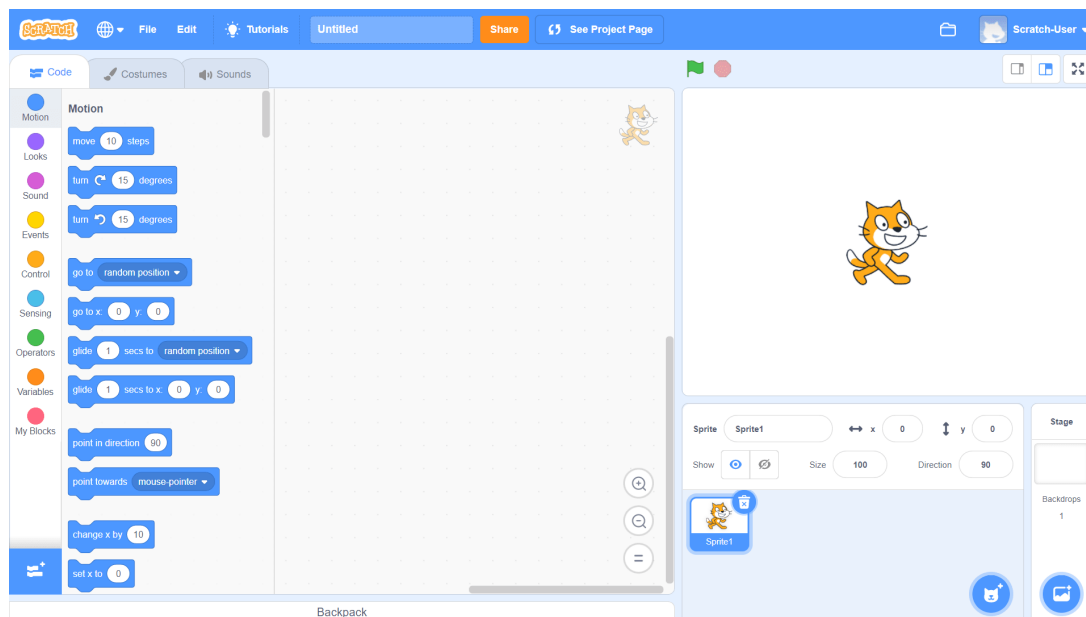


Figura 2.10: Tela do Scratch 3.0. Scratch Wiki [4].

Em geral, observa-se que as principais diferenças entre as versões 1.4, 2.0 e 3.0 do Scratch estão relacionadas à interface, à disponibilidade em plataformas, às funcionalidades e às implementações de blocos, além da customização e da edição de cenários e de atores [4].

2.4.1.1 Scratch Versão 3.0

O Scratch 3.0 apresenta um visual e um *design* novos e modernos, além de ter corrigido erros do Scratch 2.0. Ele não requer o Flash Player, permitindo que os usuários visualizem e editem projetos na *Web* e em dispositivos móveis [4].

Com o objetivo de ilustrar quanto às funcionalidades, os botões e a interface do Scratch 3.0, a seguir é apresentada a *interface* na Figura 2.11 do *software*, bem como a Tabela 2.2, contendo a descrição das áreas e das funcionalidades marcadas na Figura 2.11, conforme a numeração [4].

Tabela 2.2: Interface e funcionalidades do Scratch 3.0. Adaptado de Scratch Wiki [4].

Número	Definição
1	Cabeçalho do Scratch, onde se concentram as opções de configurações e <i>login</i> .
2	Selecionar o idioma.
3	Importar (do computador) ou exportar (para o computador) um projeto.
4	Restaurar um projeto e ativar/desativar o modo turbo para um processamento mais rápido.
5	Acessar os tutoriais básicos e práticos para aprender de forma guiada.
6	Navegar nas abas de “Código”, “Fantasias” e “Sons”. A “Área de Código” permite construir <i>scripts</i> que executam o projeto, “Fantasias” permite que o usuário altere a aparência do ator usando o “Editor de figuras” e “Sons” usa o “Editor de Sons” para editar sons.
7	Selecionar entre as categorias de comandos. Contém as seguintes divisões: “Movimento”, “Aparência”, “Som”, “Eventos”, “Controle”, “Sensores”, “Operadores”, “Variáveis” e “Meus blocos”.
8	Escolher na lista todos os comandos que podem ser arrastados para a “Área de Código” para fazer <i>scripts</i> .
9	Adicionar extensões
10	A área de edição é onde se concentram as edições mencionadas no item 6.
11	Iniciar ou parar a execução do programa.
12	Alterar o tamanho de visualização do palco e deixá-lo no modo de exibição em tela cheia.
13	O “Palco” é onde os atores são exibidos e realizam suas ações programadas no <i>script</i> .
14	O “Painel do Ator” permite a alternância entre os atores e as suas configurações (tamanho, direção e a visualização) dentro do “Palco”.
15	Adicionar ou desenhar um “Ator”, ou “Cenário”.

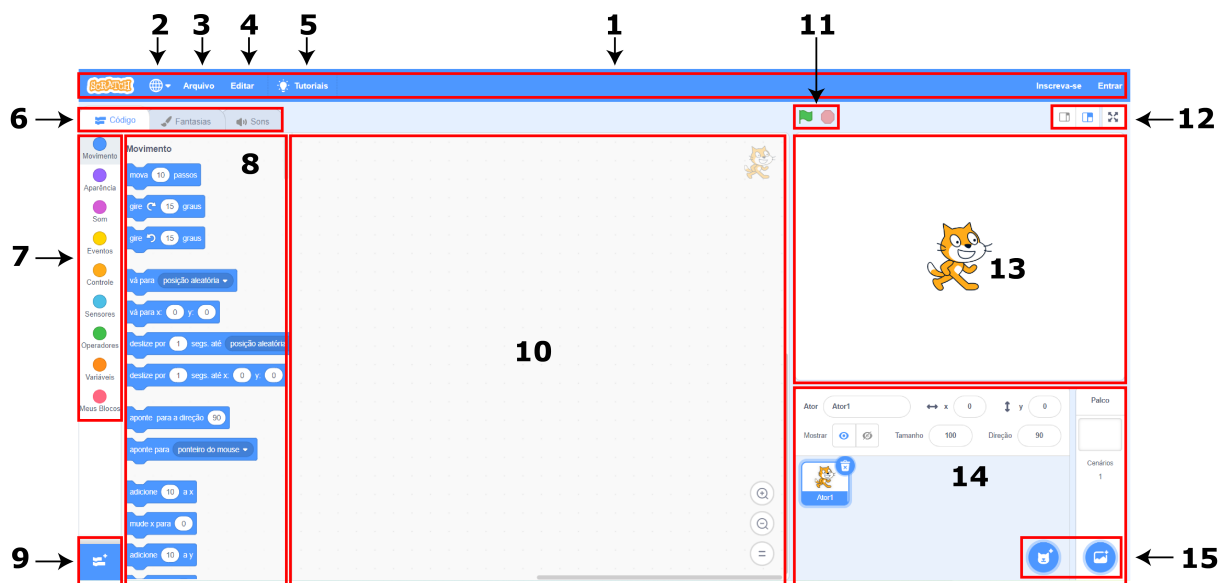


Figura 2.11: Tela do Editor Scratch 3.0. Scratch, 2022.

O Scratch 3.0 é organizado por atores. Cada ator tem características únicas, como trajes, tamanho, posição, efeitos e variáveis, além de um código específico que pode ser executado paralelamente e a partir do qual a interação com outros atores do cenário é possibilitada. O editor Scratch foi projetado para facilitar a codificação. A área para edição do projeto está à esquerda, enquanto o Palco (para visualização de projetos) e o painel de atores estão à direita [4].

Os blocos de cada ator são exibidos na área de código. Eles são estruturados em *scripts*, cadeias de blocos conectados e que são executados juntos. O bloco de comandos contém uma lista de todos os blocos que podem ser arrastados e utilizados. Além disso, os blocos são organizados em dez categorias diferentes: Movimento, Aparência, Som, Eventos, Controle, Sensores, Operadores, Variáveis e meus blocos. Alguns podem mover o ator ou alterar sua aparência; alguns interagem com outros atores ou com o Palco, enquanto outros realizam operações internas, como as operações lógicas e matemáticas ou a alteração de variáveis e de listas [4].

2.4.2 Avaliação de Projetos — Dr. Scratch

Apesar de ser um grande desafio mensurar o nível do PC de uma pessoa, alguns autores defendem que isso é possível através da análise da complexidade dos projetos de *softwares* criados em determinadas tecnologias [22].

O mapeamento sistemático de Araújo, Andrade e Guerrero [23] sobre avaliação do PC no Brasil apontou que a categoria de testes (questionários, pré e pós-testes elaborados pe-

los próprios autores, bem como provas) foi o instrumento mais utilizado pelos pesquisados para avaliar o PC (9/22 artigos), seguido por códigos e projetos, considerando a análise das estruturas de programação presentes/ausentes ou o processo de desenvolvimentos. Encontram também a avaliação observacional e qualitativa (com discussão das atividades e análise do que foi produzido).

O estudo de mapeamento sistemático sobre as abordagens avaliativas das competências do PC, com base na análise de código de Alves, Wangenheim e Hauck [24], encontrou 23 artigos relevantes descrevendo 14 abordagens diferentes. A maioria delas foram desenvolvidas para avaliar o código criado com o Scratch, o que demonstra que ele é uma ferramenta amplamente utilizada entre as práticas de Programação e ensino do PC.

Para mensurar o PC nos projetos Scratch, os artigos utilizavam avaliação manual dos códigos e até os que usavam *frameworks* de avaliação, o principal critério era a ausência ou presença de estruturas de programação para determinar o nível de competência do estudante em determinado conceito ou habilidade [23].

Para o Scratch, foi desenvolvida uma ferramenta intitulada Dr. Scratch, que avalia os códigos com base no envio do arquivo da extensão do Scratch ou no *link* do projeto. Dr. Scratch infere a competência demonstrada pelo desenvolvedor nos seguintes sete conceitos: abstração e decomposição de problemas, raciocínio lógico, sincronização, paralelismo, noções algorítmicas de controle de fluxo, interatividade do usuário e representação de dados. A cada categoria é atribuída uma pontuação de 1, 2 ou 3 em função da complexidade e frequência das estruturas e estratégias de programação utilizadas; porém, se o projeto não atender ao requisito em nenhum nível, ele não receberá nenhuma pontuação (nulo).

Após a análise do projeto, o Dr. Scratch calcula a pontuação a partir da soma simples das pontuações parciais de cada categoria e classifica o desenvolvedor com a pontuação menor ou igual a sete (≤ 7) como “básico”, entre oito e quatorze (≥ 8 e ≤ 14) como “em desenvolvimento” e como “proficientes” aqueles que têm pontuação igual ou superior a quinze (> 15).

Para dar transparência aos critérios das pontuações das métricas utilizadas no Dr. Scratch, os criadores Moreno-León, Robles e Román-González [6] elaboraram a Tabela 2.3, que relaciona a métrica pontuada (1 a 3) aos comandos do Scratch ou aos estados do sistema associados ao conceito avaliado. A partir desta tabela, é possível entender os requisitos para a pontuação de cada categoria.

Tabela 2.3: Conceitos e nível de competência Dr. Scratch. Traduzido de “*Dr. Scratch: Automatic Analysis of Scratch Projects to Assess and Foster Computational Thinking*”, Moreno-León, Robles e Román-González [6].

Conceitos PC	Nível de competência			
	Nulo	Básico (1 ponto)	Em desenvolvimento (2 pontos)	Proficiente (3 pontos)
Abstração e decomposição de problemas	-	Mais de um comando e mais de um ator	Definição de blocos	Uso de clones
Paralelismo	-	Dois comandos na bandeira verde	Dois comandos quando uma tecla for pressionada ou dois comandos a partir de um <i>click</i> em um mesmo ator	Dois comandos “Quando recebo a mensagem”, criar clone ou dois comandos “quando o cenário muda para”, em paralelo
Lógica	-	Comando “Se”	Comando “Se então”	Operadores lógicos
Sincronização	-	Comando “Espere”	Comandos: “transmissão”, “quando recebe mensagem”, “para tudo”, “para o comando”	Comandos: “espere até”, “quando o fundo mudar para”, “transmita e aguarde”
Controle de fluxo	-	Sequência de blocos	Comando “Repita sempre”	Comando “Repita até”
Interatividade com o usuário	-	Bandeira verde	Blocos: “tecla pressionada”, “perguntar e esperar”, “blocos de <i>mouse</i> ”	Interação a partir da entrada de áudio ou vídeo
Representação de dados	-	Modificadores de propriedades do ator	Operação com variáveis	Operação com listas

Capítulo 3

Metodologia Proposta

Conforme fundamentado no Capítulo 2, é necessário incluir o PC na EBB, especialmente no Ensino Médio, visto que, conforme relatado na Seção 2.3, a BNCC reserva parte do texto do Ensino Médio para discutir sobre a importância das tecnologias digitais e da computação na vida dos estudantes nessa etapa educacional.

Conforme enfatiza a BNCC [1], para o Ensino Médio:

Os jovens estão dinamicamente inseridos na cultura digital, não somente como consumidores, mas se engajando cada vez mais como protagonistas. Portanto, na BNCC dessa etapa (Ensino Médio), o foco passa a estar no reconhecimento das potencialidades das tecnologias digitais para a realização de uma série de atividades relacionadas a todas as áreas do conhecimento, a diversas práticas sociais e ao mundo do trabalho.

Como metodologia de inserção do PC na EBB, e conforme a BNCC, foi proposto o curso intitulado “Aprenda Pensamento Computacional e Scratch”, elaborado com base nos objetivos, habilidades e competências da BNCC para o Ensino Médio, nas Diretrizes para Ensino de Computação, na Educação Básica da Sociedade Brasileira de Computação (SBC) e nos relatos de experiências de cursos com objetivos, ferramentas, metodologias e público-alvo semelhantes aos propostos por Oliveira *et al.* [25], Schoeffel *et al.* [26], Marinho *et al.* [27] e Costa *et al.* [28].

3.1 Objetivos do Curso

O objetivo geral do curso consiste no ensino dos conceitos do PC (coleta de dados, análise de dados, representação de dados, decomposição de problemas, abstração, algoritmo e procedimentos, automação, simulação e paralelismo) através da LPV Scratch.

Os objetivos específicos são:

- Ensinar os conceitos do PC e de seus pilares;

- Contextualizar os pilares do PC à realidade dos estudantes;
- Ensinar sobre as diferentes formas de representação de algoritmo (pseudocódigo, fluxograma e Scratch);
- Ensinar sobre a utilização da ferramenta Scratch;
- Ensinar os conceitos e ferramentas do Scratch;
- Estimular o aprendizado de linguagens de programação no Ensino Médio;
- Expandir as competências e habilidades em lógica de programação;
- Estimular o desenvolvimento criativo.

3.2 Conteúdo Programático e Carga Horária do Curso

Para alcançar os objetivos do curso, os conteúdos programáticos foram estruturados em dois módulos:

- O primeiro módulo aborda os conceitos do PC e de seus pilares (2 horas). O conteúdo proposto e o material didático foram adaptados a partir do curso “*Computational Thinking for Problem Solving*” da Universidade da Pensilvânia [2]. O curso utilizado como referência para esse módulo, apesar de ter sido produzido no âmbito da Universidade da Pensilvânia, não possui nenhum pré-requisito e destina-se a qualquer pessoa que tenha interesse em abordar problemas de forma sistemática, desenvolver soluções mais eficientes e compreender como o PC pode ser utilizado no processo de resolução de problemas, ou seja, nenhuma experiência anterior em ciência da computação ou programação é necessária por parte do usuário do curso para que ele seja realizado;
- O segundo módulo aborda os conceitos, a utilização da plataforma Scratch e o desenvolvimento do projeto (8 horas). O conteúdo proposto foi baseado na proposta do artigo intitulado “*Utilização do Scratch como Ferramenta no Ensino de Pensamento Computacional para Crianças*” de Aono *et al.* [7], e o material didático foi baseado no “*Creative Computing Curriculum*”, desenvolvido por pesquisadores do Laboratório de Computação Criativa da Escola de Graduação em Educação de Harvard [29].

Para que não houvesse tantas desistências devido à quantidade de encontros, para que as idas dos estudantes à escola fossem otimizadas e considerando que os alunos estudam no turno contrário ao da aplicação do curso, foi proposta a carga horária de aula de dez

horas, tendo cada encontro a duração de duas horas, o que totalizam cinco encontros. A divisão dos conteúdos e da carga horária foi pensada para que estes se adaptassem ao cenário educacional dos estudantes do Ensino Médio público do Distrito Federal e estão detalhados na Tabela 3.1.

3.3 Metodologia de Ensino do Curso

Para desenvolver o PC, é necessário incorporar estratégias eficientes ao processo de ensino-aprendizagem, fazendo com que o estudante participe da construção do conhecimento de forma flexível, ativa, criativa e usufruir dos benefícios da utilização da ferramenta educacional proposta. Logo, o embasamento das estratégias de aprendizagem precisa estar bem consolidado para que se consiga atingir os objetivos sugeridos.

Estudos têm registrado evidências de que o uso de metodologias ativas de aprendizagem são mais eficientes e produzem resultados superiores aos da abordagem tradicional, como o maior interesse e engajamento dos estudantes e a promoção de um pensamento crítico e profundo. Isto por meio de um processo de ensino contextualizado, que estabelece condições para que os estudantes, ainda que orientados, também sejam responsáveis por sua própria aprendizagem [30] [31].

A implementação de metodologias ativas de aprendizagem no contexto do ensino de programação pode oportunizar diversos benefícios, como a maior aceitação, satisfação, interesse e motivação em programar, além da melhora no desempenho em trabalhos colaborativos e na habilidade de comunicação por parte dos estudantes.

Tabela 3.1: Conteúdo programático e carga horária do curso “Aprenda Pensamento Computacional e Scratch”. Adaptado de “*Computational Thinking for Problem Solving*”, Davidson e Murphy [2] e “Utilização do Scratch como Ferramenta no Ensino de Pensamento Computacional para Crianças”, Aono *et al.* [7].

APRENDA PENSAMENTO COMPUTACIONAL E SCRATCH		
Conteúdo	Descrição	CH (h)
Módulo I		2
Introdução do curso e ambientação	Apresentar o curso e conhecer os estudantes	
Pensamento Computacional	Apresentar o significado de Pensamento Computacional	
Pilares do Pensamento Computacional	Descrever os quatro pilares do pensamento computacional	2
Fluxograma	Apresentar conceitos de fluxograma e exemplos	
Pseudocódigo	Apresentar conceitos de pseudocódigo e exemplos	
Módulo II		8
Introdução à programação e utilização do Scratch	Apresentar a programação para resolução de tarefas cotidianas para os alunos, o ambiente de programação Scratch e os diferentes tipos de comandos para criação de cenários	
Conceitos de algoritmos e criação de figuras, efeitos visuais e sonoros com Scratch	Apresentar os conceitos de algoritmos, descrição narrativas de algoritmos, efeitos sonoros e visuais disponíveis no Scratch, utilização de camadas e criação de cenas com animações	2
Procedimentos	Apresentar e implementar procedimentos em Scratch e técnicas para construção de comandos	2
Variáveis	Apresentar e implementar tipos de dados no Scratch, criar e manipular dados	
Estruturas de decisão	Apresentar e implementar técnicas para resolução de problemas, uso dos comandos de seleção e fluxo de controle	2
Estruturas de repetição	Apresentar e implementar estruturas de repetição para execução de comandos eficientes e variáveis de controle	
Processamento de <i>strings</i>	Apresentar e implementar técnicas para processamento de <i>strings</i>	2
Finalização do projeto e apresentação	Finalizar os últimos ajustes do projeto e apresentar o funcionamento e código	
CARGA HORÁRIA TOTAL:		10

A proposta do curso é a de que os estudantes desenvolvam as habilidades e competências do PC através do Scratch em uma proposta de aprendizagem ativa, através da qual são consideradas e aplicadas as seguintes abordagens e premissas do curso:

- **Aprendizagem Ativa:** Ao decorrer do curso são feitos exercícios, perguntas, tarefas, e projetos que motivem os estudantes a procurarem o conhecimento necessário para atingir o objetivo proposto, trazendo o estudante para uma participação ativa na aquisição do conhecimento.
- **Aprendizagem Significativa:** Os exemplos demonstrados são pensados para que se identifique a realidade da maioria dos estudantes, considerando o contexto tanto social quanto escolar de cada um deles e buscando assim seu interesse pelo conteúdo apresentado através da aprendizagem significativa.
- **Aprendizagem Colaborativa:** Na proposta do curso, foi previsto que os estudantes realizariam o projeto e as atividades em grupos de no mínimo duas pessoas. Assim, as trocas de ideias refinariam tanto o pensamento lógico quanto as soluções para os problemas. Dessa maneira, a construção do conhecimento também se tornaria muito mais efetiva, colaborativa e compartilhada.
- **Aprendizagem Baseada em Projetos:** Ao longo das aulas, os estudantes desenvolveram um projeto. O desenvolvimento do projeto é incremental em relação aos conteúdos repassados nas partes expositivas e práticas de cada aula.

3.4 Avaliação do Curso

A avaliação de aprendizagem dos estudantes é realizada através da análise observacional do desenvolvimento e do desempenho na elaboração das atividades e do projeto, bem como na análise automática dos códigos dos projetos desenvolvidos ao longo do curso, utilizando a ferramenta Dr. Scratch (veja mais sobre a ferramenta e critérios de avaliação na Seção 2.4.2). Além disso, são aplicados questionários antes e depois do curso para entender o perfil e a avaliação dos estudantes.

3.5 Planos de Aula do Curso

Considerando que os artigos científicos de Oliveira *et al.* [25], Schoeffel *et al.* [26], Marinho *et al.* [27] e Costa *et al.* [28] não disponibilizam os planos de aula disponibilizados, foi elaborada uma sequência didática autoral para o curso proposto. Apesar de os planos das cinco aulas que compõem o curso intitulado “Aprenda Pensamento Computacional

e Scratch” terem sido desenvolvidos para os estudantes de escolas públicas do Ensino Médio no Distrito Federal, é possível que qualquer pessoa que tenha domínio do conteúdo consiga adaptar e aplicar o curso para estudantes.

As Figuras 3.1 a 3.3 exemplificam a estrutura do plano de aula da “Aula 01 — Pensamento Computacional”. Todos os planos de aula do curso estão disponíveis no Apêndice A. Importante enfatizar que o professor que quiser aplicar o curso proposto poderá utilizar e adaptar todos os materiais.

Na Figura 3.1 são detalhadas as informações do curso como: o nome do curso, o tema, a carga horária, os objetivos e os conteúdos previstos para a aula.

Na Figura 3.2 são detalhados os conteúdos para cada parte da aula com: o tempo previsto, o material didático e o detalhamento do que deve ocorrer, de forma planejada e sequencial, se possível. As apresentações mencionadas no “Material didático” estão disponíveis para consulta e utilização.¹

Na Figura 3.3 estão detalhados o método de avaliação e as referências bibliográficas da aula.

2.1. AULA 01 - Pensamento Computacional	
Informações do curso:	
Curso:	Aprenda Pensamento Computacional e Scratch
Tema:	Pensamento Computacional
Carga horária:	2 horas
Objetivos:	
Apresentar o curso e conhecer os estudantes Apresentar o significado de Pensamento Computacional Descrever os quatro pilares para o pensamento computacional Apresentar Conceitos de fluxograma e exemplos Apresentar Conceitos de pseudocódigo e exemplos	
Conteúdos:	
<ul style="list-style-type: none"> • Introdução do curso e ambientação • Conhecendo os estudantes • Pensamento computacional • Pilares do pensamento computacional • Fluxograma • Pseudocódigo 	

Figura 3.1: Exemplo da estrutura do plano de aula do curso intitulado “Aprenda Pensamento Computacional e Scratch” - Parte 1. Elaborado pelo Autor, 2022.

¹Material didático disponível em:
https://github.com/viniciusaguiarmonteiro/TCC_UnB_Vinicius_Aguiar_Monteiro.git.

Procedimentos:			
Conteúdo	Tempo previsto	Detalhamento	Material utilizado
Introdução do curso e ambientação	0:20	O professor irá se apresentar, apresentar o projeto e o curso que irá ministrar e seu detalhamento. Pedirá aos estudantes que se apresentem com o Nome, a Idade, a Turma e o que os motivou a participar do curso. Este momento tem o objetivo de aproximar os estudantes e conhecê-los melhor para conduzir as aulas.	Slide "Aula 1 - Pensamento Computacional"
Conhecendo os estudantes	0:05	Os estudantes irão preencher o formulário de caracterização do perfil da turma para subsidiar o conteúdo e atividades das próximas aulas.	Questionário "Perfil dos Estudantes"
Pensamento computacional	0:20	Exposição teórica e apresentação do conceito do Pensamento Computacional.	Slide "Aula 1 - Pensamento Computacional"
Pilares do pensamento computacional	0:25	Exposição teórica e apresentação dos pilares do Pensamento Computacional, além de dar exemplos de como aplicá-los na resolução de problemas. A cada explicação de um dos quatro pilares do Pensamento Computacional, é solicitado aos estudantes que deem outros exemplos dos pilares aplicados à vida.	
Fluxograma	0:25	Exposição teórica e apresentação do conceito de Fluxograma como ferramenta de construção do pilar "Algoritmo" do Pensamento Computacional. São apresentados os símbolos e significados das estruturas que formam os fluxogramas. São demonstrados os fluxogramas com exemplos de atividades do cotidiano. Ao final, é passada uma atividade extraclasse sobre construção de fluxogramas.	
Pseudocódigo	0:25	Exposição teórica e apresentação do conceito de Pseudocódigo como ferramenta mais avançada e estruturada de construção do pilar "Algoritmo" do Pensamento Computacional. É apresentada a estrutura do pseudocódigo. Exemplificar problemas do cotidiano, resolvidos em fluxogramas e com pseudocódigos, trazendo assim uma comparação direta entre as duas formas de resolver um problema. Ao final, é passada uma atividade extraclasse sobre construção de fluxogramas e pseudocódigos.	

Figura 3.2: Exemplo da estrutura do plano de aula do curso intitulado “Aprenda Pensamento Computacional e Scratch” - Parte 2. Elaborado pelo Autor, 2022.

<p>Avaliação:</p> <p>A avaliação ocorrerá durante a apresentação dos exemplos dos estudantes. Deve-se observar não só os estudantes que tenham conhecimentos prévios do assunto, mas também o comportamento dos estudantes, como respeito pelo professor, interesse no conteúdo e interação quando solicitado. É extremamente importante que nesse momento o professor observe o contexto social dos estudantes e seus conhecimentos no uso de tecnologias, para que nas próximas aulas futuras possa citar exemplos de conceitos de programação aplicados às situações conhecidas dos estudantes, estimulando a aprendizagem significativa.</p>
<p>Referências:</p> <p>AONO, Alexandre Hild; RODY, Hugo Vianna Silva; MUSA, Daniela Leal; PEREIRA, Vanessa Andrade; ALMEIDA, Jurandy. A Utilização do Scratch como Ferramenta no Ensino de Pensamento Computacional para Crianças. In: WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO (WEI), 25. , 2017, São Paulo. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2017 . ISSN 2595-6175. DOI: https://doi.org/10.5753/wei.2017.3556.</p> <p>Davidson, Susan e Chris Murphy: Computational thinking for problem solving. University of Pennsylvania, 2022. https://www.coursera.org/learn/computational-thinking-problem-solving/, acesso em 2022-07-24.</p> <p>Lab, Creative Computing: Creative computing curriculum. Desenvolvido por Laboratório de Computação Criativa da Escola de Graduação em Educação de Harvard e Traduzido por Rede Brasileira de Aprendizagem Criativa (aprendizagemcriativa.org) e Universidade Estadual de Feira de Santana (UEFS), Brasil, 2019. https://lcl.media.mit.edu/resources/readings/creative-computing-guide.pt.pdf, acesso em 2021-10-06.</p>

Figura 3.3: Exemplo da estrutura do plano de aula do curso intitulado “Aprenda Pensamento Computacional e Scratch” - Parte 3. Elaborado pelo Autor, 2022.

Capítulo 4

Relato de Experiências - Aplicações do Curso

As experiências de aplicação do curso intitulado “Aprenda Pensamento Computacional e Scratch” (Capítulo 3) foram feitas em duas etapas ao longo de um ano, em três turmas e escolas diferentes. A primeira aplicação foi realizada em caráter experimental com uma turma no Centro de Ensino Médio 01 do Paranoá (CEM 01 - Paranoá), na modalidade de ensino remoto, devido à pandemia da COVID-19, e a outra aplicação foi feita em duas turmas, na modalidade presencial e já incorporadas as melhorias observadas na primeira aplicação do curso, nas seguintes escolas: uma turma no Centro de Ensino Médio da Asa Norte (CEAN) e a outra no Centro de Ensino Médio Paulo Freire (CEMPF).

Todas as aplicações do curso foram ofertadas através do projeto de extensão intitulado “CODIFICO”, coordenado pela Profa. Dra. Maristela Terto de Holanda, do Departamento de Ciência da Computação (CIC) da Universidade de Brasília (UnB). No período de aplicação do curso o projeto recebeu apoio financeiro nos editais “7/2021 - Polo de Extensão do Paranoá” e “1/2022 - Programa Institucional de Bolsas de Extensão (PIBEX)”, do Decanato de Extensão (DEX) da UnB.

Algumas questões de organização e planejamento da primeira aplicação difere da utilizada na segunda, visto que a primeira foi ministrada em caráter de teste para melhorias do curso e a segunda é a proposta final, detalhada no Capítulo 3, refinada a partir do que foi observado e aprendido. As diferenças entre as aplicações estão detalhadas na Tabela 4.1.

Tabela 4.1: Diferenças entre a primeira e segunda aplicação do curso. Elaborado pelo Autor, 2022.

Item	Primeira aplicação	Segunda aplicação
Carga horária total	Quatorze horas	Dez horas
Carga horária por aula	Uma hora	Duas horas
Divisão do conteúdo	Três módulos (Pensamento Computacional, Scratch e desenvolvimento do projeto final)	Dois módulos (Pensamento Computacional e Scratch)
Projeto final	Desenvolvido individualmente	Desenvolvido em grupo
Modalidade de ensino	Remoto	Presencial
Quantidade de professores	Dois professores (Marina Pinho e Vinícius Aguiar)	Um professor (Vinícius Aguiar)

4.1 Primeira Aplicação do Curso

A primeira aplicação do curso teve o objetivo de coletar dados e percepções para poderem ser feitas melhorias para a versão final e as próximas aplicações do curso.

O curso foi ministrado para estudantes do Centro de Ensino Médio 01 do Paranoá (CEM 01 do Paranoá) na modalidade de ensino remoto, visto restrições impostas pelo Governo do Distrito Federal (GDF), devido à pandemia da COVID-19, o que impossibilitou a oferta na modalidade presencial.

4.1.1 Introdução e Contextualização

A escola escolhida para a primeira aplicação do curso foi o CEM 01 do Paranoá, visto que em anos anteriores já tinha feito outras atividades com outros projetos da UnB, logo a parceria já estava estabelecida e possuía suporte da coordenação pedagógica e direção da escola para realizar as atividades.

O CEM 01 do Paranoá é uma escola urbana, fica localizada na região administrativa VII – Paranoá do Distrito Federal e atende anualmente uma média de 1.900 estudantes, do 1º ao 3º ano do Ensino Médio.

4.1.2 Relato de Experiência

Organização e planejamento do curso

A coordenação do projeto CODIFICO com a direção da escola fez a divulgação do curso nos meios de comunicações com os estudantes. Foi encaminhado um texto explicativo, o

link do formulário de inscrições (Apêndice B) e o cartaz de divulgação do curso (Apêndice C) para que os estudantes fizessem o registro de interesse e participação no curso.

Após a finalização do período de inscrições, 120 estudantes fizeram o preenchimento do formulário. Em seguida, os professores organizaram um repositório no Google Sala de Aula para armazenar os materiais relevantes para o curso. Após isso os estudantes foram comunicados do local virtual das aulas.

A seguir será relatado as experiências de aplicação por aula ministrada nessa escola.

Aula 01 - Introdução do curso e Pensamento Computacional e seus pilares

- **Dia e Horário:** 26/10/2021 – 12h às 13h.
- **Presentes:** Vinte e seis estudantes.
- **Conteúdo programático:** Introdução do curso e ambientação; Pensamento computacional; Pilares do Pensamento Computacional.
- **Atividade da aula:** Elaborar um fluxograma (algoritmo) para resolver algum problema. Utilizar os símbolos apresentados na aula. Utilizar condição e entrada de dados, igual o último exemplo apresentado em aula. Preferencialmente feito à mão. Enviar foto da resolução no repositório da turma.
- **Observações:** A aula iniciou 12:05 onde foram apresentados o curso, as regras, as orientações, o projeto CODIFICO e os professores. Em seguida foi solicitado aos estudantes que se apresentassem e dissessem o porquê se interessaram e fizeram a inscrição no curso. Após a apresentação de quem quis falar/digitar, iniciou-se a aula expositiva e prática do conteúdo programado. Ao longo da aula foi solicitado aos estudantes que dessem exemplos de aplicações dos quatro pilares do Pensamento Computacional no cotidiano. Aproximadamente dez estudantes interagiram. Após finalizada a aula, foram disponibilizados na plataforma Google Sala de Aula a apresentação utilizada, o formulário de presença e a atividade para entregar.

Aula 02 - Fluxograma e Pseudocódigo

- **Dia e Horário:** 28/10/2021 – 12h às 13h.
- **Presentes:** Vinte estudantes.
- **Conteúdo programático:** Fluxogramas; Pseudocódigos.
- **Atividade da aula:** Fazer um fluxograma e um Pseudocódigo para resolver um problema de sua escolha. Fluxograma: Observar o que significa cada figura geométrica. As setas indicam a ordem de execução e sequência. Pseudocódigo: Observar a “identação” das linhas. A ordem das atividades importa.

- **Observações:** A aula iniciou 12:05 com a apresentação dos objetivos de aprendizagem e em seguida começou a aula expositiva e prática do conteúdo programado. Ao longo da aula os estudantes foram questionados se haviam dúvidas. As dúvidas foram sanadas. Aproximadamente cinco estudantes interagiram no bate-papo. Após finalizada a aula, foram disponibilizados na plataforma Google Sala de Aula a apresentação utilizada, o formulário de presença e a atividade para entregar.

Aula 03 - Introdução ao Scratch

- **Dia e Horário:** 04/11/2021 – 12h às 13h.
- **Presentes:** Dez estudantes.
- **Conteúdo programático:** Introdução à programação; utilização do Scratch.
- **Atividade da aula:** Criar um projeto no Scratch com apenas os seguintes blocos: Vá para; Deslize; diga; mostre; esconda; defina o tamanho como; Toque o som até o fim; quando esse ator for clicado; espere; repita; cada bloco deve ser usado pelo menos uma vez.
- **Observações:** A aula iniciou 12:05 com a apresentação dos objetivos de aprendizagem e em seguida começou a aula expositiva e prática do conteúdo programado. Foi reforçada a necessidade de entrega dos exercícios. Ao longo da aula os estudantes foram questionados se haviam dúvidas. As dúvidas foram sanadas. Aproximadamente cinco estudantes interagiram no bate-papo. Após finalizada a aula, foram disponibilizados na plataforma Google Sala de Aula a apresentação utilizada, o formulário de presença e a atividade para entregar.

Aula 04 - Algoritmos e Criação de figuras e efeitos visuais e sonoros

- **Dia e Horário:** 09/11/2022 - 19h30min - 20h30min.
- **Presentes:** Dez estudantes.
- **Conteúdo programático:** Apresentar os conceitos de algoritmos, descrição narrativas de algoritmos, efeitos sonoros e visuais disponíveis no Scratch, utilização de camadas e criação de cenas com animações.
- **Atividade da aula:** Criar um projeto no Scratch, de acordo com o projeto padrão, e realizar modificações nos códigos.
- **Observações:** Após a terceira aula, houve mudança no horário da aula, visto que os estudantes voltaram a ter aula presencial, impossibilitando a participação no curso. Essa foi a primeira aula depois da mudança do horário do curso. A aula

começou 19:35 com a apresentação dos objetivos de aprendizagem. Foi reforçada a necessidade de entrega dos exercícios. Deu-se início a aula expositiva e prática do conteúdo programado. Ao longo da aula os estudantes foram questionados se haviam dúvidas. As dúvidas foram sanadas. Aproximadamente cinco estudantes interagiram no bate-papo. Após finalizada a aula, foram disponibilizados na plataforma Google Sala de Aula a apresentação utilizada, o formulário de presença e a atividade para entregar. Alguns estudantes entraram em contato após a aula para tirar dúvidas sobre os exercícios já passados.

Aula 05 - Procedimentos

- **Dia e Horário:** 11/11/2022 - 19h30min - 20h30min.
- **Presentes:** Onze estudantes.
- **Conteúdo programático:** Apresentar e Implementar procedimentos em Scratch e técnicas para construção de comandos.
- **Atividade da aula:** Crie um projeto no Scratch com pelo menos duas funções.
- **Observações:** A aula começou 19:35 com a apresentação dos objetivos de aprendizagem. Deu-se início a aula expositiva e prática do conteúdo programado. Ao longo da aula os estudantes foram questionados se haviam dúvidas. As dúvidas foram sanadas. Aproximadamente cinco estudantes interagiram no bate-papo. Após finalizada a aula, foram disponibilizados na plataforma Google Sala de Aula a apresentação utilizada, o formulário de presença e a atividade para entregar.

Aula 06 - Variáveis

- **Dia e Horário:** 16/11/2022 - 19h30min - 20h30min.
- **Presentes:** Oito estudantes.
- **Conteúdo programático:** Apresentar e Implementar tipos de dados no Scratch, criar e manipular dados.
- **Atividade da aula:** Criar um projeto no Scratch que utilize variáveis de diferentes tipos. Você deverá: Criar as variáveis; Utilizar pelo menos quatro variáveis; Solicitar dados do usuário e armazená-los; Realizar operações com as variáveis; Mostrar o resultado das operações.
- **Observações:** A aula começou 19:35 com a apresentação dos objetivos de aprendizagem. Deu-se início a aula expositiva e prática do conteúdo programado. Ao

longo da aula os estudantes foram questionados se haviam dúvidas. As dúvidas foram sanadas. Aproximadamente cinco estudantes interagiram no bate-papo. Após finalizada a aula, foram disponibilizados na plataforma Google Sala de Aula a apresentação utilizada, o formulário de presença e a atividade para entregar.

Aula 07 - Estruturas de Decisão

- **Dia e Horário:** 18/11/2022 - 19h30min - 20h30min.
- **Presentes:** Sete estudantes.
- **Conteúdo programático:** Apresentar e Implementar técnicas para resolução de problemas, uso dos comandos de seleção e fluxo de controle.
- **Atividade da aula:** Criar um projeto no Scratch com pelo menos duas estruturas condicionais.
- **Observações:** A aula começou 19:35 com a apresentação dos objetivos de aprendizagem. Deu-se início a aula expositiva e prática do conteúdo programado. Ao longo da aula os estudantes foram questionados se haviam dúvidas. As dúvidas foram sanadas. Aproximadamente cinco estudantes interagiram no bate-papo. Após finalizada a aula, foram disponibilizados na plataforma Google Sala de Aula a apresentação utilizada, o formulário de presença e a atividade para entregar.

Aula 08 - Estruturas de Repetição

- **Dia e Horário:** 23/11/2022 - 19h30min - 20h30min.
- **Presentes:** Sete estudantes.
- **Conteúdo programático:** Apresentar e Implementar estruturas de repetição para execução de comandos eficientes e variáveis de controle.
- **Atividade da aula:** Criar um projeto no Scratch com pelo menos duas estruturas de repetição.
- **Observações:** A aula começou 19:35 com a apresentação dos objetivos de aprendizagem. Deu-se início a aula expositiva e prática do conteúdo programado. Ao longo da aula os estudantes foram questionados se haviam dúvidas. As dúvidas foram sanadas. Aproximadamente cinco estudantes interagiram no bate-papo. Após finalizada a aula, foram disponibilizados na plataforma Google Sala de Aula a apresentação utilizada, o formulário de presença e a atividade para entregar.

Aula 09 - Processamento de Strings

- **Dia e Horário:** 25/11/2022 - 19h30min - 20h30min.
- **Presentes:** Cinco estudantes.
- **Conteúdo programático:** Apresentar e Implementar técnicas para processamento de *strings*.
- **Atividade da aula:** Criar um projeto no Scratch que utilize processamento de *strings* e trazer na próxima aula uma ideia de projeto que queira desenvolver.
- **Observações:** A aula começou 19:35 com a apresentação dos objetivos de aprendizagem. Deu-se início a aula expositiva e prática do conteúdo programado. Ao longo da aula os estudantes foram questionados se haviam dúvidas. As dúvidas foram sanadas. Aproximadamente cinco estudantes interagiram no bate-papo. Após finalizada a aula, foram disponibilizados na plataforma Google Sala de Aula a apresentação utilizada, o formulário de presença e a atividade para entregar.

Aula 10 - Desenvolvimento do projeto final

- **Dia e Horário:** 02/12/2022 - 19h30min - 20h30min.
- **Presentes:** Cinco estudantes.
- **Conteúdo programático:** Projeto Final.
- **Atividade da aula:** Depositar a ideia de projeto no Google Sala de Aula.
- **Observações:** A aula começou 19:35 com a apresentação dos objetivos de aprendizagem. Deu-se início à discussão com os estudantes sobre as ideias de desenvolvimento do projeto final deles. Os estudantes tiraram dúvidas sobre o projeto e as possibilidades de desenvolvimento até que todos chegaram à conclusão do projeto que iria desenvolver até o final do curso.

Aula 11 - Desenvolvimento do projeto final

- **Dia e Horário:** 07/12/2022 - 19h30min - 20h30min.
- **Presentes:** Três estudantes.
- **Conteúdo programático:** Projeto Final.
- **Atividade da aula:** Desenvolvimento do projeto final.
- **Observações:** A aula começou 19:35 com a apresentação dos objetivos de aprendizagem. Deu-se início a aula tirando dúvidas dos estudantes em relação ao desenvolvimento do projeto final. Ao longo da aula os estudantes foram questionados

se havia dúvidas. As dúvidas foram sanadas. Os três estudantes interagiram no bate-papo.

Aula 12 - Desenvolvimento do projeto final

- **Dia e Horário:** 09/12/2022 - 19h30min - 20h30min.
- **Presentes:** Três estudantes.
- **Conteúdo programático:** Projeto Final.
- **Atividade da aula:** Desenvolvimento do projeto final.
- **Observações:** A aula começou 19:35 com a apresentação dos objetivos de aprendizagem. Deu-se início a aula tirando dúvidas dos estudantes em relação ao desenvolvimento do projeto final. Ao longo da aula os estudantes foram questionados se havia dúvidas. As dúvidas foram sanadas. Os três estudantes interagiram no bate-papo.

Aula 13 - Desenvolvimento do projeto final

- **Dia e Horário:** 14/12/2022 - 19h30min - 20h30min.
- **Presentes:** Três estudantes.
- **Conteúdo programático:** Projeto Final.
- **Atividade da aula:** Desenvolvimento do projeto final.
- **Observações:** A aula começou 19:35 com a apresentação dos objetivos de aprendizagem. Foi reforçada a necessidade de entrega dos exercícios. Deu-se início a aula tirando dúvidas dos estudantes em relação ao desenvolvimento do projeto final. Ao longo da aula os estudantes foram questionados se havia dúvidas. As dúvidas foram sanadas. Os três estudantes interagiram no bate-papo.

Aula 14 - Apresentação do projeto final

- **Dia e Horário:** 16/12/2022 - 19h30min - 20h30min.
- **Presentes:** Três estudantes.
- **Conteúdo programático:** Projeto Final.
- **Atividade da aula:** Apresentação do projeto final.

- **Observações:** A aula começou 19:35 com a apresentação dos objetivos de aprendizagem. Foi reforçada a necessidade de entrega dos exercícios. Deu-se início a aula com a apresentação dos projetos finais dos estudantes. A dinâmica de apresentação foi que primeiro os estudantes apresentaram o uso e funcionamento do projeto e depois passaram os códigos na tela explicando o que cada parte do projeto fazia. Após finalizadas as apresentações, houve uma conversa com os estudantes que finalizaram o curso, onde os estudantes fizeram colocações sobre as possíveis melhorias do curso para ser aplicado em outras oportunidades.

4.1.3 Resultados e Discussões

Como a divulgação e as inscrições foram realizadas em ambiente virtual e sem controle de vínculo com a instituição de ensino, provavelmente alguns dos inscritos não estavam formalmente matriculados na escola alvo do curso, porém isso não foi considerado um limitante para a primeira aplicação do curso, visto que estavam na faixa etária alvo do curso. Apesar de ter recebido 120 inscrições, apenas 26 estudantes compareceram na primeira aula.

Ao longo da aplicação do curso houve duas intercorrências que impossibilitaram a participação de muitos estudantes que fizeram inscrição e estavam acompanhando o curso até a segunda semana de aula (26/10/2022 a 09/11/2022), conforme ilustrado na Figura 4.1.

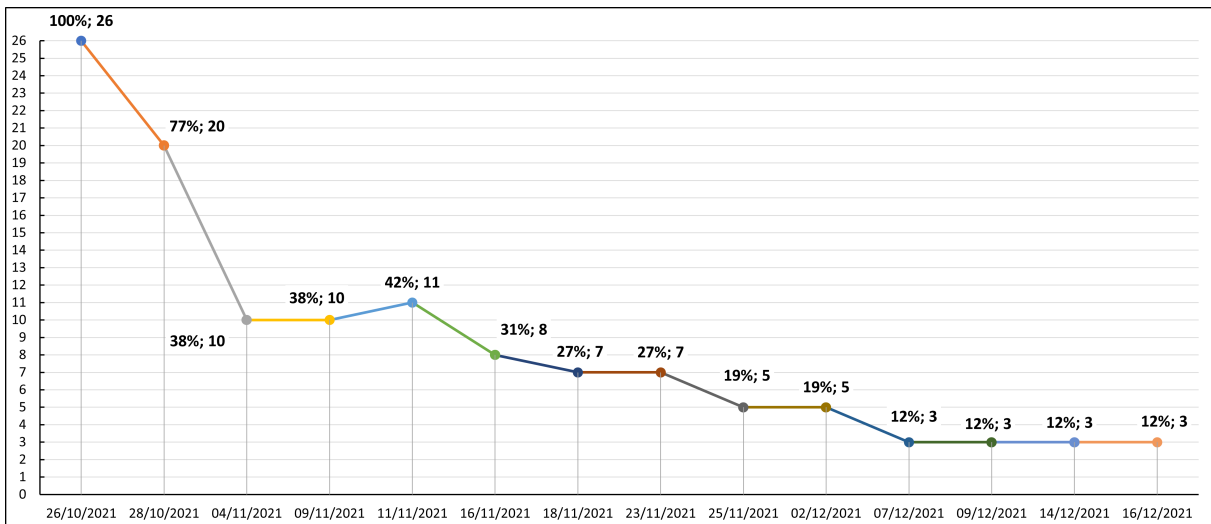


Figura 4.1: Histórico de presença dos estudantes no curso - CEM 01 do Paranoá. Elaborado pelo Autor, 2022.

A primeira ocorrência foi que os estudantes da escola estavam cursando o ensino regular na modalidade de ensino remoto, devido à pandemia da COVID-19, porém o GDF decretou o retorno híbrido (presencial e remoto) obrigatório de todos os estudantes no dia 02/11/2021. Conforme relatado pelos estudantes, metade dos estudantes das turmas regulares compareciam presencialmente em uma semana e a outra metade na outra semana, causando assim um retorno confuso, visto que os estudantes nunca haviam vivenciado esse tipo de dinâmica escolar. Considerando que o horário do curso era das 12h às 13h, o retorno às aulas ocasionou muitas desistências no curso, visto que na semana que o estudante precisava ir presencialmente na escola, o impossibilitava a participação no curso devido ao horário. Aproximadamente seis estudantes relataram por mensagens no WhatsApp que desistiram por esse motivo nessa etapa do curso, antes da primeira semana de curso, desconsiderando alguns estudantes que desistiram por motivos pessoais e que não foram relatados.

A segunda ocorrência foi quando o GDF decretou o retorno 100% presencial, aproximadamente na terceira aula do curso, o que causou grande parte das desistências no curso. No dia 06/11/2022 foi feita uma consulta aos estudantes, por mensagem no WhatsApp, sobre o motivo das faltas nas aulas. Os estudantes relataram que com o retorno das aulas presenciais estava muito difícil acompanhar o curso, visto que quem estudava pela manhã ainda não havia chegado em casa e quem estudava à tarde estava a caminho da escola. Considerando esse novo cenário, foi sugerido um novo horário para as aulas e solicitada a confirmação de continuidade dos estudantes no curso. A partir da quarta aula o horário do curso foi de 19h30min até 20h30min, porém mesmo com a mudança de horário a participação caiu de 20 para 10 estudantes da segunda para a terceira aula. Da quarta aula em diante as desistências (de 10 para 3 estudantes) foram gradativas.

As ocorrências podem ter causado a maioria das desistências dos estudantes, outra hipótese que pode ter inviabilizado a participação dos estudantes é que apenas 11 (44%) dos 26 estudantes que participaram da primeira aula possuíam computador para utilizar durante o curso. Além disso, a interação dos estudantes durante as aulas foi baixa, o que pode ter sido uma barreira para continuidade e finalização do curso.

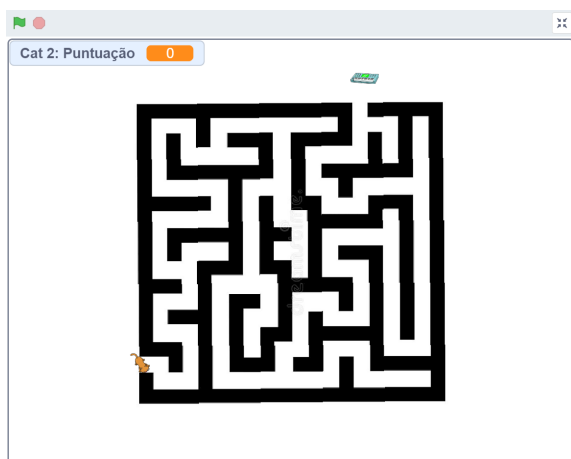
Ao final do curso, três estudantes efetivamente finalizaram o curso e entregam o projeto final proposto. As telas dos projetos podem ser visualizadas na Figura 4.2. Os projetos foram analisados na ferramenta Dr. Scratch e os estudantes “A”, “B” e “C”, obtiveram, respectivamente, as notas 12, 18 e 18 do total de 21 pontos. Os resultados detalhados das análises dos projetos podem ser visualizados na Figura 4.3 e os critérios e pontuações estão detalhados na Seção 2.4.2.

O desempenho dos estudantes foi bastante satisfatório, tanto pela análise automática

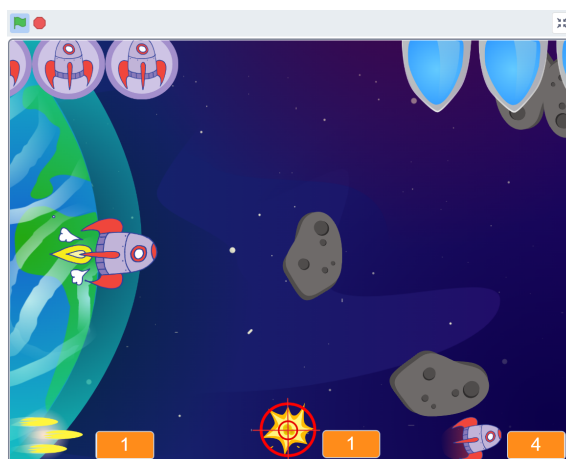
dos projetos no Dr. Scratch quanto pelo produto apresentado, pelo domínio na apresentação dos códigos e o desempenho durante as aulas.

Na última aula foi relatado pelos três estudantes que o curso foi muito longo e que não gostaram de o desenvolvimento do projeto ter ocorrido somente a partir da décima aula, o que pode ter sido o motivo principal, ou complementar, da desistência de muitos estudantes, além das intercorrências causadas pela COVID-19 que impactaram diretamente o andamento e participação dos estudantes no curso. Relataram também que aprenderam o conteúdo do curso, o que pode ser confirmado com o processo avaliativo do curso.

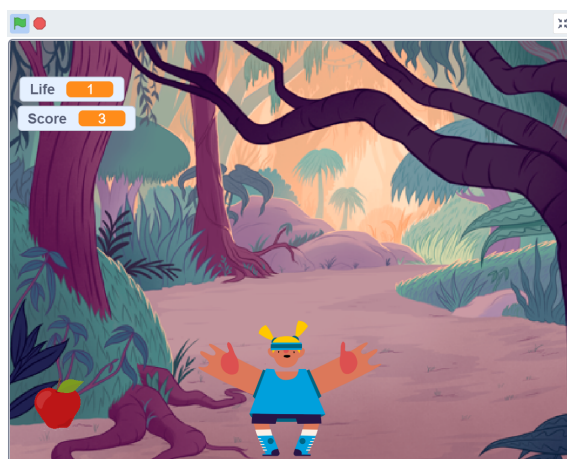
O curso obteve êxito e atingiu seus objetivos de aprendizagem, mesmo com o baixo quantitativo de concluintes e sendo a primeira aplicação para melhorias nas aplicações futuras.



4.2.1: Projeto “Labirinto”. Estudante A.



4.2.2: Projeto “Meteorite Destroyer”. Estudante B.

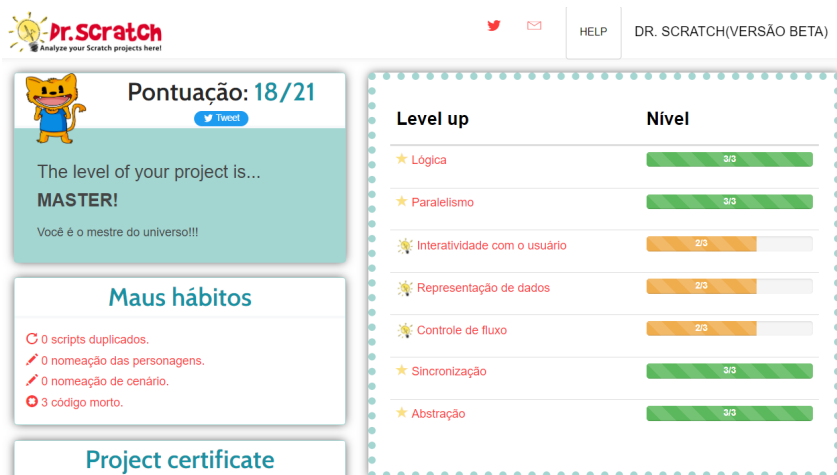


4.2.3: Projeto “Gula sem fim”. Estudante C.

Figura 4.2: Telas dos projetos desenvolvidos pelos estudantes do CEM 01 do Paranoá. Elaborado pelo Autor, 2022.



4.3.1: Tela de avaliação do projeto “Labirinto”. Estudante A.



4.3.2: Tela de avaliação do projeto “Meteorite Destroyer”. Estudante B.



4.3.3: Tela de avaliação do projeto “Gula sem fim”. Estudante C.

Figura 4.3: Tela das avaliações dos projetos desenvolvidos pelos estudantes do CEM 01 do Paranoá. Elaborado pelo Autor, 2022.

4.1.4 Lições Aprendidas

A partir dessa experiência de aplicação do curso no CEM 01 do Paranoá, têm-se as seguintes lições aprendidas:

- Um dos grandes desafios da modalidade de ensino remoto é a dificuldade de dar assistência e suporte na realização dos exercícios e desenvolvimento do projeto, visto que os estudantes não tiram muitas dúvidas no decorrer do curso, causando assim uma dificuldade para entender as dúvidas e dificuldades de cada estudante;
- A desistência é uma das questões mais difíceis de entender, visto que os estudantes apenas param de assistir às aulas, sem expor os motivos, logo requer uma atenção e percepção mais minuciosa sobre o que está ocorrendo, além de uma ação rápida;
- O professor deve estar preparado para adaptar sua proposta conforme a realidade e contexto social na qual está inserido.
- Um curso muito longo, principalmente virtual, pode desmotivar os estudantes a participarem até o final.

4.2 Segunda Aplicação do Curso

A segunda aplicação do curso teve o objetivo de consolidar uma versão final e avaliar o curso, considerando as alterações feitas após a primeira aplicação.

A segunda aplicação do curso foi realizada no Centro de Ensino Médio Paulo Freire (CEMPF) e no Centro de Ensino Médio da Asa Norte (CEAN), cada escola com uma turma, na modalidade presencial, visto que as restrições impostas pelo GDF, devido à pandemia da COVID-19, não estavam mais vigentes, possibilitando a aplicação do curso nos laboratórios das escolas escolhidas. As escolas foram escolhidas, por já terem feito outros projetos com a UnB, logo as parcerias já estavam estabelecidas, facilitando o suporte das coordenações pedagógicas e direções das escolas nas realizações das atividades.

4.2.1 Centro de Ensino Médio da Asa Norte - CEAN

4.2.1.1 Introdução e Contextualização

O CEAN é uma escola urbana, que está localizada na região administrativa I – Plano Piloto do Distrito Federal e atende anualmente uma média de 900 estudantes, do 1º ao 3º ano do Ensino Médio.

Considerando que o curso foi presencial e no turno contrário que os estudantes estudavam, foi preciso escolher um turno alvo para ministrar o curso. O turno escolhido nessa

escola foi o matutino, ou seja, os estudantes ficavam após a aula (matutino) para participar do curso no turno contrário (vespertino), uma vez por semana, às segundas-feiras, das 14h às 16h. Considerando a organização das disponibilidades de turmas por turnos da escola, o curso atendeu, exclusivamente, os estudantes do 2º e 3º ano do Ensino Médio nessa escola.

4.2.1.2 Relato de Experiência

Organização e planejamento do curso

A coordenação do projeto CODIFICO entrou em contato com a coordenação da escola, que prontamente atendeu a demanda e solicitou a elaboração de um cartaz para divulgação do curso aos estudantes nos meios de comunicações digitais da escola.

Considerando que o laboratório da escola estava desabilitado e não estava sendo utilizado para nenhuma atividade, foi necessário ir à escola antes do início das aulas para realizar um diagnóstico da quantidade de computadores em funcionamento, assim como verificar a conexão com a *internet*. Após o diagnóstico no laboratório, foi verificado estar sem *internet* e que 20 computadores estavam em funcionamento, logo as aulas precisaram ser adiadas para que a coordenação da escola corrigisse o problema de conexão.

Foi encaminhado à coordenação da escola um texto explicativo, o *link* do formulário de inscrições (Apêndice D) e o cartaz de divulgação do curso (Apêndice E) para que os estudantes fizessem o registro de interesse e participação no curso.

Após a finalização do período de inscrições, 23 estudantes (19 (82,6%) do 3º ano e quatro (17,4%) do 2º ano) fizeram o preenchimento do formulário. Em seguida foi organizado um repositório no Google Drive contendo todo o material das aulas. Os estudantes foram comunicados sobre o local, dias e horários das aulas presenciais.

A seguir será relatado as experiências de aplicação por aula ministrada nessa escola.

Aula 01 - Pensamento Computacional

- **Dia e Horário:** 02/05/2022 – 14h às 16h.
- **Presentes:** Quinze estudantes.
- **Conteúdo programático:** Introdução do curso e ambientação; Pensamento computacional; Pilares do Pensamento Computacional; Fluxograma; Pseudocódigo.
- **Atividade da aula:** Atividade 1 - Elaborar um fluxograma (algoritmo) para resolver algum problema. Utilizar os símbolos apresentados na aula. Utilizar condição e entrada de dados, igual o último exemplo apresentado em aula. Preferencialmente

feito à mão. Enviar foto da resolução no repositório da turma. Atividade 2 - Fazer um fluxograma e um pseudocódigo para resolver um problema de sua escolha. Fluxograma: Observar o que significa cada figura geométrica. As setas indicam a ordem de execução e sequência. Pseudocódigo: Observar a “identação” das linhas. A ordem das atividades importa.

- **Observações:** A aula iniciou 14:05 onde foram apresentados o curso, as regras, as orientações, o projeto CODIFICO e o professor. Em seguida foi solicitado aos estudantes que se apresentassem e dissessem o porquê se interessaram e fizeram a inscrição no curso. Após a apresentação de quem quis falar, iniciou-se a aula expositiva e prática do conteúdo programado. Ao longo da aula foi solicitado aos estudantes que dessem exemplos de aplicações dos quatro pilares do Pensamento Computacional no cotidiano. Aproximadamente dez estudantes interagiram. Após finalizada a aula, a apresentação utilizada, o formulário de caracterização do perfil da turma (Apêndice G) e as orientações das atividades foram disponibilizados no Google Drive. Apesar de o início da turma ter sido adiado devido à conexão com a *internet* do laboratório, os computadores ainda estavam sem conexão, o que impossibilitou o preenchimento do formulário em sala de aula, diante disso, foi solicitado que os estudantes fizessem o preenchimento em casa.

Aula 02 - Introdução à Programação e ao Scratch

- **Dia e Horário:** 09/05/2022 – 14h às 16h.
- **Presentes:** Treze estudantes.
- **Conteúdo programático:** Revisão do conteúdo da aula anterior; Introdução à programação; Criando conta no Scratch; Explorando o Scratch; Atividade dez blocos; Apresentação do Scratch; Conceitos de algoritmos; Criação de figuras, efeitos visuais e sonoros no Scratch; Discussão sobre o projeto.
- **Atividade da aula:** Atividade 1 - Explorar o Scratch por dez minutos, sem orientação. Atividade 2 - Criar um projeto no Scratch com apenas os seguintes blocos: Vá para; Deslize; diga; mostre; esconda; defina o tamanho como; Toque o som até o fim; quando esse ator for clicado; espere; repita; cada bloco deve ser usado pelo menos uma vez.
- **Observações:** A aula iniciou 14:05 com a revisão do conteúdo da aula anterior, principalmente os conceitos do Pensamento Computacional. Em seguida ocorreu a apresentação dos objetivos de aprendizagem. Deu-se início a aula expositiva e prática do conteúdo programado. Ao longo da aula os estudantes foram questionados

se haviam dúvidas. As dúvidas foram sanadas. No final da aula foi solicitado que os estudantes formassem duplas ou trios e discutissem as ideias de projetos que queriam desenvolver. Aproximadamente dez estudantes interagiram. Estava previsto que os estudantes criassem a conta no Scratch 3.0, porém os computadores ainda estavam sem acesso à *internet*. As atividades práticas no Scratch não foram afetadas, visto que os computadores possuíam a versão 1.4 do Scratch. Após finalizada a aula, a apresentação utilizada e as orientações das atividades foram disponibilizados no Google Drive.

Aula 03 - Procedimentos e Variáveis no Scratch

- **Dia e Horário:** 16/05/2022 – 14h às 16h.
- **Presentes:** Dez estudantes.
- **Conteúdo programático:** Revisão do conteúdo da aula anterior; Definição do projeto; Procedimentos; Variáveis; Desenvolvimento do projeto.
- **Atividade da aula:** Criar um projeto no Scratch que utilize variáveis de diferentes tipos. Criar as variáveis; Utilizar pelo menos quatro variáveis; Solicitar dados do usuário e armazená-los; Realizar operações com as variáveis; Mostrar o resultado das operações.
- **Observações:** A aula iniciou 14:30 com a revisão do conteúdo da aula anterior. Em seguida ocorreu a apresentação dos objetivos de aprendizagem. Deu-se início a aula expositiva e prática do conteúdo programado. Ao longo da aula os estudantes foram questionados se haviam dúvidas. As dúvidas foram sanadas. Aproximadamente cinco estudantes interagiram. Na metade da aula os estudantes se organizaram perto de suas duplas ou trios e solicitado que eles falassem a ideia de projeto deles. Após a validação das ideias, os estudantes iniciaram o desenvolvimento dos projetos. Os estudantes tiraram dúvidas e pediram ideias ao professor ao longo do desenvolvimento, porém também interagiram e discutiram as soluções dos problemas com seu grupo. A aula atrasou 30 minutos porque o professor precisou fazer alguns testes e adequações no laboratório para conseguir conectar alguns computadores à *internet*, visto que a escola ainda não havia conseguido resolver o problema. Após isso o professor conseguiu conectar cinco dispositivos, ou seja, todos os grupos conseguiram utilizar um computador com conexão para utilizar o Scratch 3.0 e desenvolver o projeto. Após finalizada a aula, a apresentação utilizada e as orientações das atividades foram disponibilizados no Google Drive.

Aula 03 - Procedimentos e Variáveis no Scratch

- **Dia e Horário:** 16/05/2022 – 14h às 16h.
- **Presentes:** Dez estudantes.
- **Conteúdo programático:** Revisão do conteúdo da aula anterior; Definição do projeto; Procedimentos; Variáveis; Desenvolvimento do projeto.
- **Atividade da aula:** Criar um projeto no Scratch que utilize variáveis de diferentes tipos. Criar as variáveis; Utilizar pelo menos quatro variáveis; Solicitar dados do usuário e armazená-los; Realizar operações com as variáveis; Mostrar o resultado das operações.
- **Observações:** A aula iniciou 14:30 com a revisão do conteúdo da aula anterior. Em seguida ocorreu a apresentação dos objetivos de aprendizagem. Deu-se início a aula expositiva e prática do conteúdo programado. Ao longo da aula os estudantes foram questionados se haviam dúvidas. As dúvidas foram sanadas. Aproximadamente cinco estudantes interagiram. Na metade da aula os estudantes se organizaram perto de suas duplas ou trios e solicitado que eles falassem a ideia de projeto deles. Após a validação das ideias, os estudantes iniciaram o desenvolvimento dos projetos. Os estudantes tiraram dúvidas e pediram ideias ao professor ao longo do desenvolvimento, porém também interagiram e discutiram as soluções dos problemas com seu grupo. A aula atrasou 30 minutos porque o professor precisou fazer alguns testes e adequações no laboratório para conseguir conectar alguns computadores à *internet*, visto que a escola ainda não havia conseguido resolver o problema. Após isso o professor conseguiu conectar cinco dispositivos, ou seja, todos os grupos conseguiram utilizar um computador com conexão para utilizar o Scratch 3.0 e desenvolver o projeto. Após finalizada a aula, a apresentação utilizada e as orientações das atividades foram disponibilizados no Google Drive.

Aula 04 - Estruturas de Decisão e Repetição no Scratch

- **Dia e Horário:** 23/05/2022 – 14h às 16h.
- **Presentes:** Dez estudantes.
- **Conteúdo programático:** Revisão do conteúdo da aula anterior; Definição do projeto; Procedimentos; Variáveis; Desenvolvimento do projeto.
- **Atividade da aula:** Atividade 1 - Criar um projeto no Scratch com pelo menos duas estruturas condicionais. Lembrando que seu programa deve ser diferente dos que foram apresentados em sala de aula. Atividade 2 - Criar um projeto no Scratch com pelo menos duas estruturas de repetição diferentes. Lembrando que seu programa deve ser diferente dos que foram apresentados em sala de aula.

- **Observações:** A aula iniciou 14:05 com a revisão do conteúdo da aula anterior. Em seguida ocorreu a apresentação dos objetivos de aprendizagem. Deu-se início a aula expositiva e prática do conteúdo programado. Ao longo da aula os estudantes foram questionados se haviam dúvidas. As dúvidas foram sanadas. Aproximadamente cinco estudantes interagiram. Após a exposição do conteúdo, os estudantes se organizaram em seus grupos e continuaram o desenvolvimento dos projetos. Os estudantes tiraram dúvidas e pediram ideias ao professor ao longo do desenvolvimento, porém também interagiram e discutiram as soluções dos problemas com seu grupo. Após finalizada a aula, a apresentação utilizada e as orientações das atividades foram disponibilizados no Google Drive.

Aula 05 - Processamento de *strings* e Finalização e Apresentação do Projeto

- **Dia e Horário:** 30/05/2022 – 14h às 16h.
- **Presentes:** Nove estudantes.
- **Conteúdo programático:** Revisão do conteúdo da aula anterior; Processamento de *strings*; Desenvolvimento do projeto; Apresentação do projeto; Avaliação final.
- **Atividade da aula:** Criar um projeto no Scratch que utilize processamento de *strings*.
- **Observações:** A aula iniciou 14:05 com a revisão do conteúdo da aula anterior. Em seguida ocorreu a apresentação dos objetivos de aprendizagem. Deu-se início a aula expositiva e prática do conteúdo programado. Ao longo da aula os estudantes foram questionados se haviam dúvidas. As dúvidas foram sanadas. Aproximadamente cinco estudantes interagiram. Após a exposição do conteúdo, os estudantes se organizaram em seus grupos e fizeram os últimos ajustes no projeto. Após finalizados os ajustes, foi solicitado que cada grupo fizesse uma apresentação do seu projeto. Primeiro apresentaram o uso e funcionamento e depois os códigos. Após as apresentações, os estudantes puderam socializar e utilizar/jogar o projeto dos demais. Os estudantes preencheram o formulário final de avaliação do curso. Após finalizada a aula, a apresentação utilizada e as orientações da atividade foram disponibilizados no Google Drive.

4.2.1.3 Resultados e Discussões

Ao longo da aplicação do curso tiveram algumas desistências consideradas esperadas, visto que muitas vezes os estudantes se matriculam e não sabem exatamente do que se trata. Analisando o percentual e quantidade de presentes por aula, ilustrado na Figura 4.4, pode-se concluir que o percentual (40%) de desistências foi baixo.

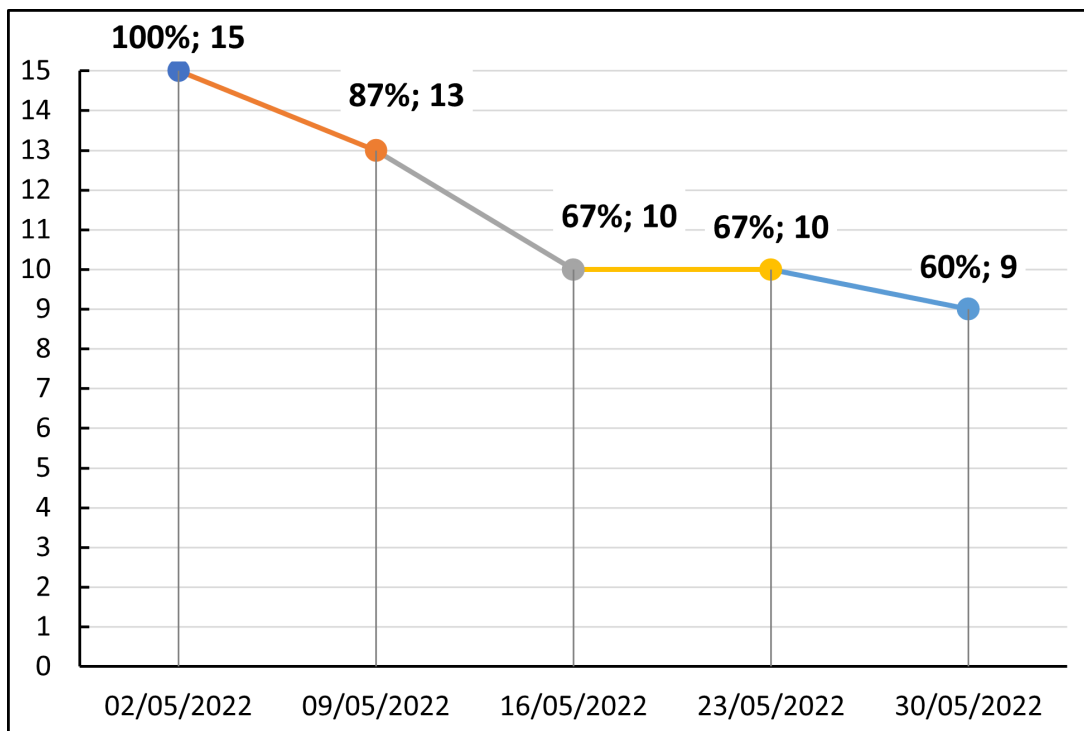
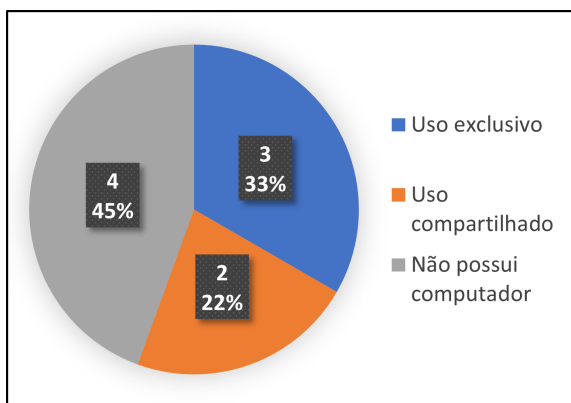
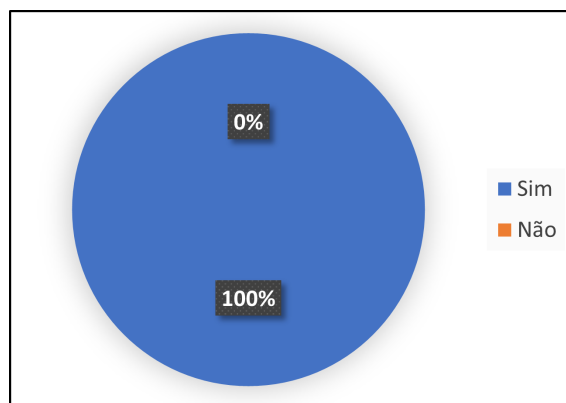


Figura 4.4: Histórico de presença dos estudantes no curso - CEAN. Elaborado pelo Autor, 2022.

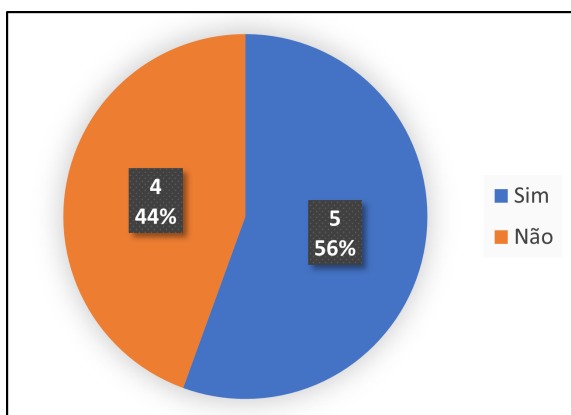
Conforme relatado na primeira aula, nove estudantes preencheram o “Formulário de Caracterização do Perfil dos Estudantes” (Apêndice G), o que possibilitou avaliar o perfil da turma para melhor condução das aulas. Também foi possível fazer análise socioeconômica e comparativa do perfil dos estudantes. Os resultados dos dados quantitativos do perfil dos nove estudantes que finalizaram o curso, estão ilustrados na Figura 4.5.



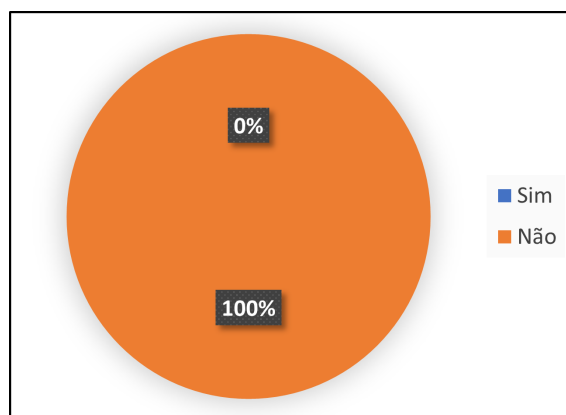
4.5.1: Possui computador?



4.5.2: Possui telefone celular?

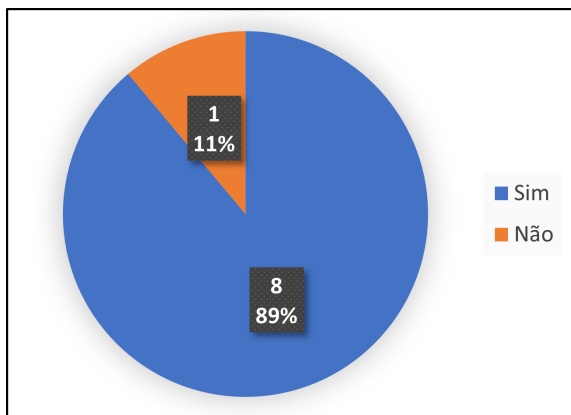


4.5.3: Possui *internet* móvel?

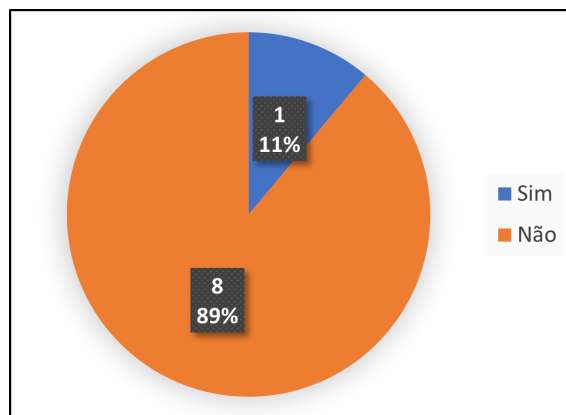


4.5.4: Possui *tablet*?

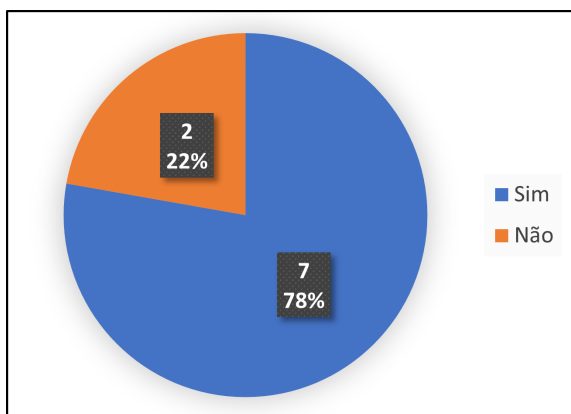
Figura 4.5: Perfil dos estudantes do CEAN. Elaborado pelo Autor, 2022.



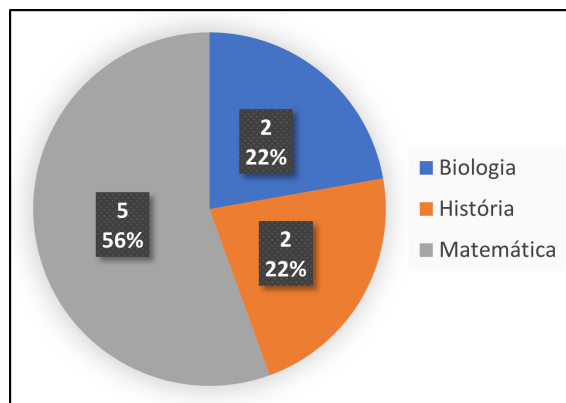
4.5.5: Possui *internet* em casa?



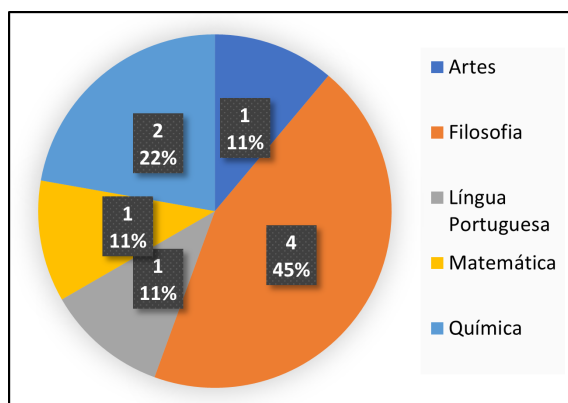
4.5.6: Possui conhecimento em programação?



4.5.7: Se considera bom em raciocínio lógico?



4.5.8: Qual disciplina mais gosta ou tem facilidade de aprender?



4.5.9: Qual disciplina menos gosta ou tem dificuldade de aprender?

Figura 4.5: Continuação: Perfil dos estudantes do CEAN. Elaborado pelo Autor, 2022.

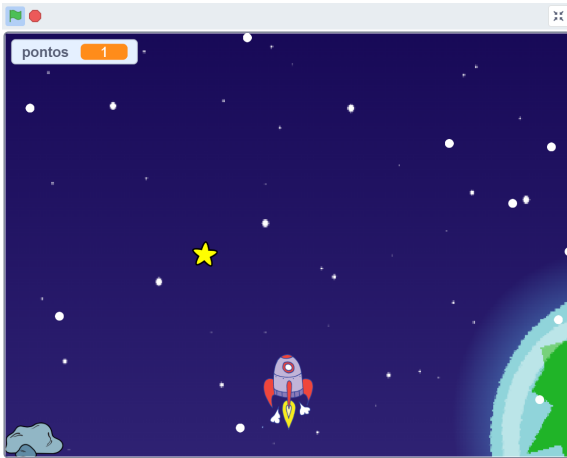
Alguns resultados do perfil dos estudantes, ilustrados na Figura 4.5 merecem destaques. Apenas 22% (2) estudantes possuíam computador de uso exclusivo, ou seja, 78% (7) dos estudantes realizavam atividades apenas na escola no horário do curso. Percebeu-se que 89% (8) dos estudantes não possuíam nenhum conhecimento em programação, o que reforça que o Scratch foi uma ótima escolha para introdução à programação nessa turma. Outro dado interessante é que dos sete (78%) estudantes que declararam serem bons em raciocínio lógico, cinco (71%), dos sete (100%), informaram que Matemática é a disciplina que eles mais gostam/tem facilidade de aprender, o que pode ter relação com afinidade com a área de Ciências Exatas.

Na aplicação do curso pôde-se perceber que os estudantes estavam bastantes engajados na elaboração do projeto, tirando dúvidas e discutindo as soluções nos grupos. Após o curso foi realizada a análise dos códigos produzidos pelos nove estudantes (5 grupos) que efetivamente finalizaram o curso e entregam o projeto final, no Dr. Scratch. A pontuação máxima do Dr. Scratch é de 21 pontos, conforme critérios e pontuações detalhados na Seção 2.4.2, logo observa-se que os estudantes, em geral, obtiveram notas satisfatórias. As telas dos projetos podem ser visualizadas na Figura 4.6 e o detalhamento da avaliação dos projetos na Figura 4.7. Os projetos dos cinco grupos obtiveram as seguintes notas:

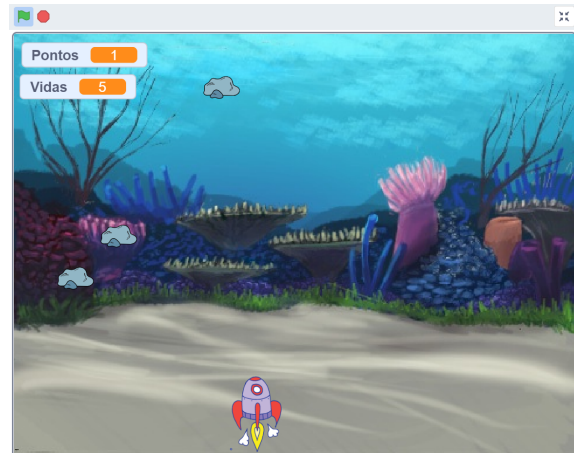
- Grupo 1 - “Chaotic Space” (estudantes “A” e “B”) 18 pontos;
- Grupo 2 - “Rocket and Rocks War” (estudantes “C” e “D”) 16 pontos;
- Grupo 3 - “Dinossauro pula cacto” (estudantes “E” e “F”) 16 pontos;
- Grupo 4 - “Pega fruta” (estudante “G”) 14 pontos;
- Grupo 5 - “Sereia e tubarão” (estudantes “H” e “I”) 11 pontos.

Em relação ao desempenho dos grupos, é importante ressaltar que o Grupo 1 (estudantes “A” e “B”) obteve a maior pontuação dentre os projetos apresentados. O maior desempenho pode estar relacionado pelo fato de que os estudantes possuíam computador para uso exclusivo e *internet* em casa, possibilitando a continuidade do projeto em casa, ou seja, mais tempo de estudo e desenvolvimento do projeto. O estudante do Grupo 4 (estudante “G”) finalizou o projeto sozinho, visto que sua dupla desistiu no meio do desenvolvimento do projeto e, mesmo com uma carga de desenvolvimento maior, teve desempenho e nota satisfatórios.

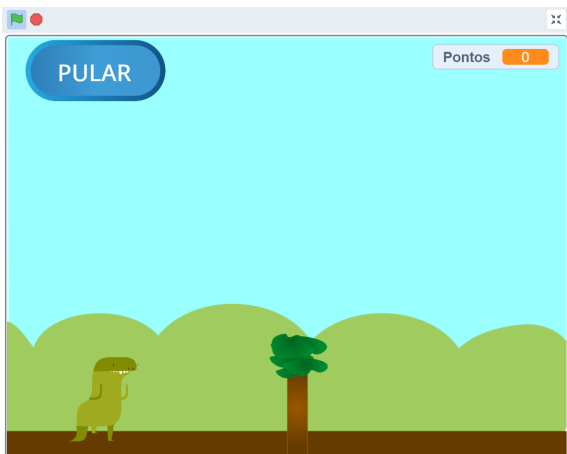
Após a finalização do curso foi aplicado o questionário (Apêndice H) que teve o objetivo de coletar a percepção dos estudantes em relação ao curso. Os resultados demonstram que os estudantes gostaram do curso, em seus diversos aspectos, conforme ilustrado na Figura 4.8.



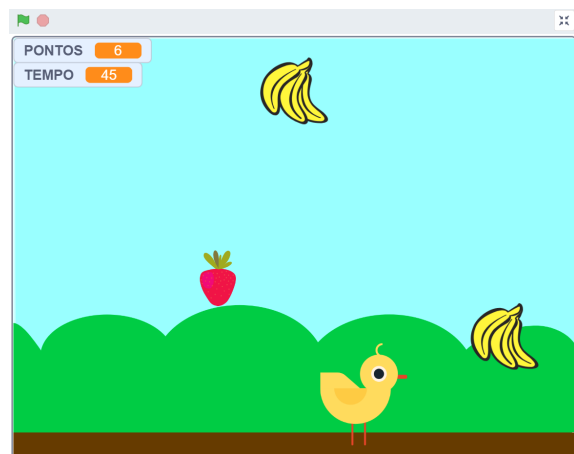
4.6.1: Projeto "Chaotic Space".



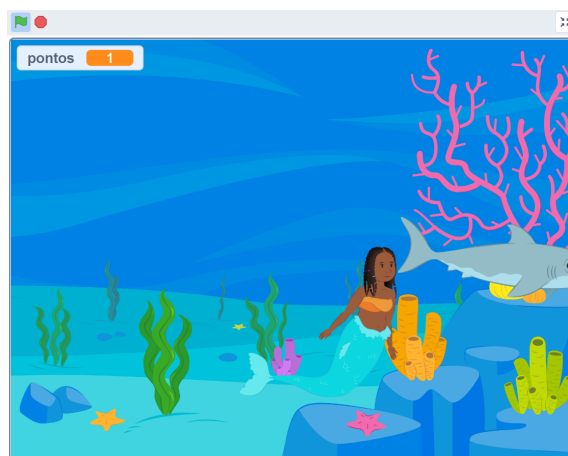
4.6.2: Projeto "Rocket and Rocks War".



4.6.3: Projeto "Dinossauro pula cacto".



4.6.4: Projeto "Pega fruta"

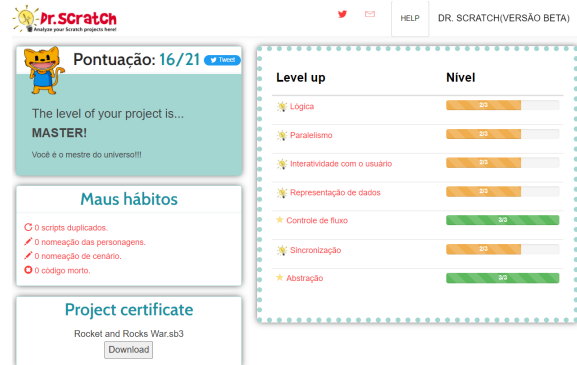


4.6.5: Projeto "Sereia e tubarão"

Figura 4.6: Telas dos projetos dos estudantes do CEAN. Elaborado pelo Autor, 2022.



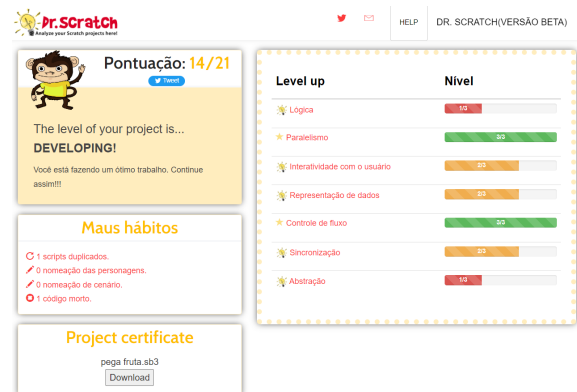
4.7.1: Avaliação Dr. Scratch - Projeto “Chaotic Space”.



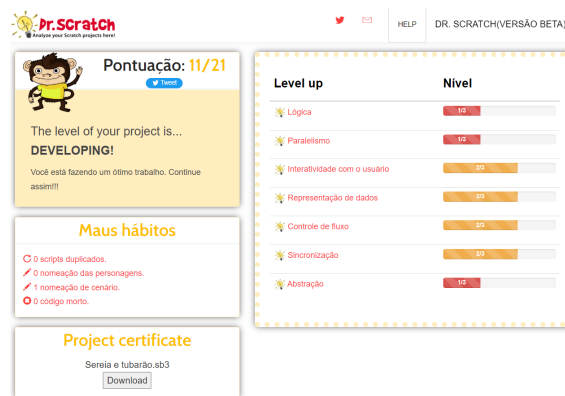
4.7.2: Avaliação Dr. Scratch - Projeto “Rocket and Rocks War”.



4.7.3: Avaliação Dr. Scratch - Projeto “Dinosauro pula cacto”.

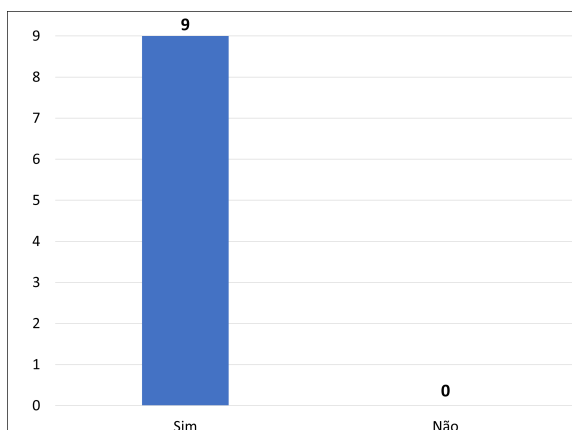


4.7.4: Avaliação Dr. Scratch - Projeto “Pega fruta”.

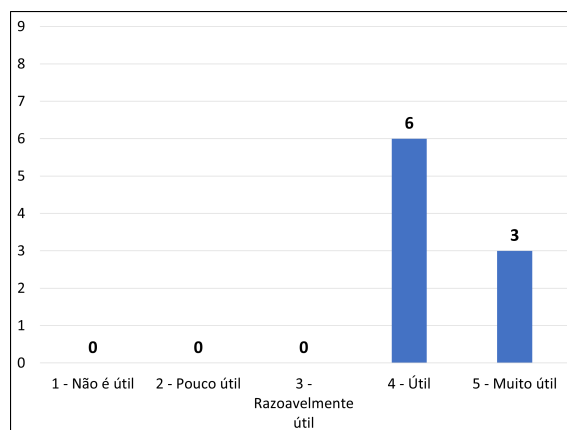


4.7.5: Avaliação Dr. Scratch - Projeto “Sereia e tubarão”.

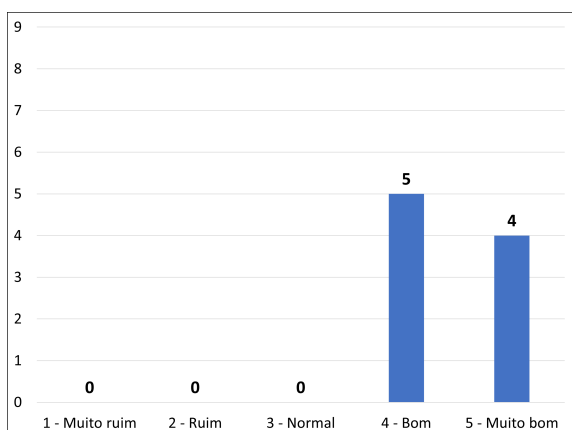
Figura 4.7: Avaliação dos projetos dos estudantes do CEAN no Dr. Scratch. Elaborado pelo Autor, 2022.



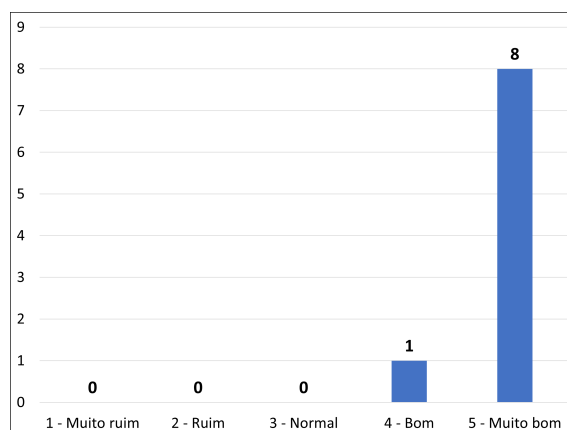
4.8.1: Você gostou do curso?



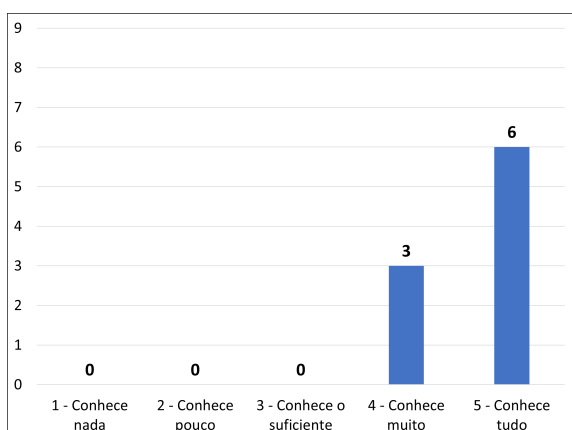
4.8.2: Você considera que a aplicação dos conceitos/temas desse curso em sua vida acadêmica/profissional e pessoal é:



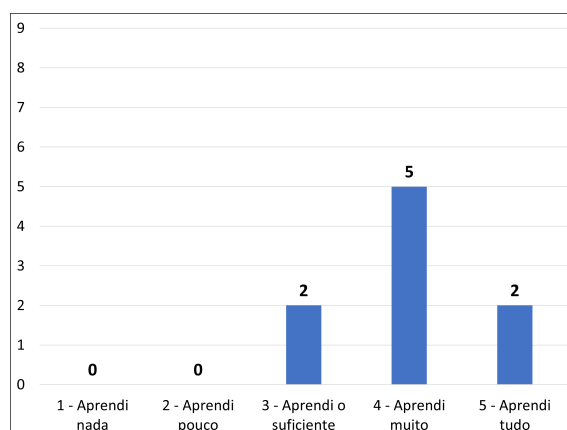
4.8.3: Como você avalia o formato e condução (metodologia) das aulas do curso?



4.8.4: Como você avalia a forma do professor comunicar e aplicar o conteúdo (didática) nas aulas do curso?



4.8.5: Como você avalia o conhecimento que o professor tem sobre o assunto do curso?



4.8.6: Como você avalia a sua aprendizagem ao longo do curso?

Figura 4.8: Avaliação final dos estudantes - CEAN. Elaborado pelo Autor, 2022.

Os dados do formulário após o curso demonstram que os estudantes gostaram do curso, em seus diversos aspectos, e consideram que aprenderam, pelo menos, o suficiente. Os estudantes sugeriram como melhorias o aumento na quantidade de aulas, continuidade do curso com outras linguagens de programação e melhoria na infraestrutura do laboratório, como conexão com a *internet* e computadores.

A pergunta sobre o nível de conhecimento em informática dos estudantes foi aplicada tanto no questionário anterior quanto no posterior ao curso, possibilitando uma comparação da percepção do conhecimento geral em informática pelos estudantes. Observa-se que 78% (7) dos estudantes consideraram que, após o curso, seu nível de conhecimento em informática aumentou em média dois pontos, conforme ilustrado na Figura 4.9.

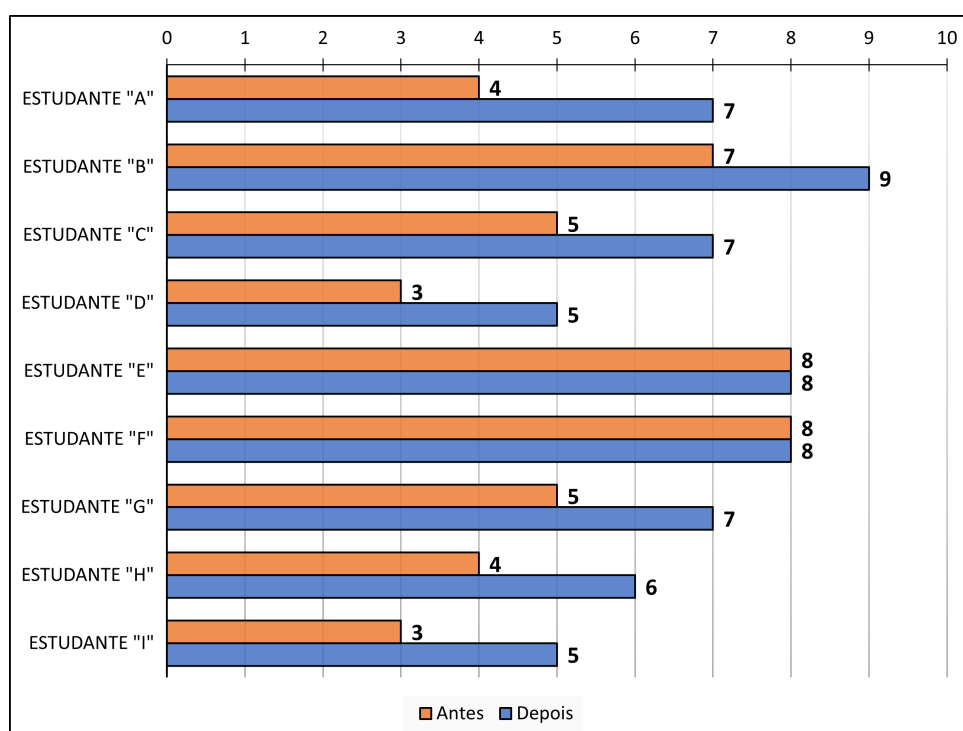


Figura 4.9: Comparativo da percepção dos estudantes quanto ao nível de conhecimento em informática antes e depois do curso - CEAN. Elaborado pelo Autor, 2022.

As maiores dificuldades encontradas na aplicação dessa turma foram em relação ao funcionamento dos computadores, visto que alguns não estavam funcionando e em relação à conexão com a *internet* que atrasou o desenvolvimento do projeto.

O curso, em geral, cumpriu com seu objetivo de aprendizagem, conforme demonstra os dados analisados dos questionários, bem como as pontuações obtidas nos projetos.

4.2.1.4 Lições Aprendidas

A partir dessa experiência de aplicação do curso no CEAN, têm-se as seguintes lições aprendidas:

- O professor precisa criar um ambiente propício a tirar dúvidas para que os estudantes se sintam à vontade para perguntar;
- Os estudantes se sentem mais próximos e acolhidos pelo professor ao ter um diálogo aberto e amigável, favorecendo a aprendizagem e diminuindo as desistências;
- O professor deve estar preparado para adaptar sua proposta conforme a realidade e contexto social na qual está inserido.

4.2.2 Centro de Ensino Médio Paulo Freire - CEMPF

4.2.2.1 Introdução e Contextualização

O CEMPF é uma escola urbana, que está localizada na região administrativa I – Plano Piloto do Distrito Federal e atende anualmente uma média de 1.000 estudantes, do 1º ao 3º ano do Ensino Médio.

Considerando que o curso foi presencial e no turno contrário que os estudantes estudavam, foi preciso escolher um turno alvo para ministrar o curso. O turno escolhido nessa escola foi o vespertino, ou seja, os estudantes precisavam ir ao turno contrário (matutino) de suas aulas regulares (vespertino) para participar do curso, uma vez por semana, às quartas-feiras, das 9h às 11h. Considerando a organização das disponibilidades de turmas por turnos da escola, o curso atendeu, exclusivamente, os estudantes do 1º ano do Ensino Médio nessa escola.

4.2.2.2 Relato de Experiência

Organização e planejamento do curso

A coordenação do projeto CODIFICO entrou em contato com a coordenação da escola, que prontamente atendeu a demanda, e solicitou a elaboração de um cartaz para divulgação do curso aos estudantes nos meios de comunicações digitais da escola.

Não foi preciso ir à escola verificar a quantidade de computadores que estavam funcionando, visto possuir uma professora responsável pelas atividades do laboratório de informática que deixou o laboratório pronto para o curso, além de ter acompanhado e dado suporte em todas as aulas.

Foi encaminhado à coordenação da escola um texto explicativo e o cartaz de divulgação do curso (Apêndice F) para que os estudantes fizessem o registro de interesse e participação

no curso. A forma de inscrição foi através do *e-mail* da professora responsável pelo laboratório.

Após a finalização do período de inscrições, sete estudantes fizeram o envio de *e-mail* com a solicitação de participação. Logo em seguida os estudantes foram comunicados do local, datas e horário das aulas. Foi organizado um repositório no Google Drive contendo o repositório do curso.

A seguir será relatado as experiências de aplicação por aula ministrada nessa escola.

Aula 01 - Pensamento Computacional

- **Dia e Horário:** 21/04/2022 – 9h às 11h.
- **Presentes:** Cinco estudantes.
- **Conteúdo programático:** Introdução do curso e ambientação; Pensamento computacional; Pilares do Pensamento Computacional; Fluxograma; Pseudocódigo.
- **Atividade da aula:** Atividade 1 - Elaborar um fluxograma (algoritmo) para resolver algum problema. Utilizar os símbolos apresentados na aula. Utilizar condição e entrada de dados, igual o último exemplo apresentado em aula. Preferencialmente feito à mão. Enviar foto da resolução no repositório da turma. Atividade 2 - Fazer um fluxograma e um pseudocódigo para resolver um problema de sua escolha. Fluxograma: Observar o que significa cada figura geométrica. As setas indicam a ordem de execução e sequência. Pseudocódigo: Observar a “identação” das linhas. A ordem das atividades importa.
- **Observações:** A aula iniciou 09:05 onde foram apresentados o curso, as regras, as orientações, o projeto CODIFICO e o professor. Em seguida foi solicitado aos estudantes que se apresentassem e dissessem o porquê se interessaram e fizeram a inscrição no curso. Após a apresentação de quem quis falar, iniciou-se a aula expositiva e prática do conteúdo programado. Ao longo da aula foi solicitado aos estudantes que dessem exemplos de aplicações dos quatro pilares do Pensamento Computacional no cotidiano. Aproximadamente dois estudantes interagiram. Após finalizada a aula, a apresentação utilizada, o formulário de caracterização do perfil da turma (Apêndice G) e as orientações das atividades foram disponibilizados no Google Drive. O laboratório estava sem conexão com a *internet*, o que impossibilitou o preenchimento do formulário em sala de aula, diante disso, foi solicitado que os estudantes fizessem o preenchimento em casa.

Aula 02 - Introdução à Programação e ao Scratch

- **Dia e Horário:** 28/04/2022 – 9h às 11h.

- **Presentes:** Cinco estudantes.
- **Conteúdo programático:** Revisão do conteúdo da aula anterior; Introdução à programação; Criando conta no Scratch; Explorando o Scratch; Atividade de dez blocos; Apresentação do Scratch; Conceitos de algoritmos; Criação de figuras, efeitos visuais e sonoros no Scratch; Discussão sobre o projeto.
- **Atividade da aula:** Atividade 1 - Explorar o Scratch por dez minutos, sem orientação. Atividade 2 - Criar um projeto no Scratch com apenas os seguintes blocos: Vá para; Deslize; diga; mostre; esconda; defina o tamanho como; Toque o som até o fim; quando esse ator for clicado; espere; repita; cada bloco deve ser usado pelo menos uma vez.
- **Observações:** A aula iniciou 09:05 com a revisão do conteúdo da aula anterior, principalmente os conceitos do Pensamento Computacional. Em seguida ocorreu a apresentação dos objetivos de aprendizagem. Deu-se início a aula expositiva e prática do conteúdo programado. Ao longo da aula os estudantes foram questionados se haviam dúvidas. As dúvidas foram sanadas. Os estudantes criaram a conta no Scratch 3.0. No final da aula foi solicitado que os estudantes formassem duplas ou trios e discutissem as ideias de projetos que queriam desenvolver. Aproximadamente dois estudantes interagiram. Após finalizada a aula, a apresentação utilizada e as orientações das atividades foram disponibilizados no Google Drive.

Aula 03 - Procedimentos e Variáveis no Scratch

- **Dia e Horário:** 05/05/2022 – 9h às 11h.
- **Presentes:** Quatro estudantes.
- **Conteúdo programático:** Revisão do conteúdo da aula anterior; Definição do projeto; Procedimentos; Variáveis; Desenvolvimento do projeto.
- **Atividade da aula:** Criar um projeto no Scratch que utilize variáveis de diferentes tipos. Você deverá: Criar as variáveis; Utilizar pelo menos quatro variáveis; Solicitar dados do usuário e armazená-los; Realizar operações com as variáveis; Mostrar o resultado das operações.
- **Observações:** A aula iniciou 09:05 com a revisão do conteúdo da aula anterior. Em seguida ocorreu a apresentação dos objetivos de aprendizagem. Deu-se início a aula expositiva e prática do conteúdo programado. Ao longo da aula os estudantes foram questionados se haviam dúvidas. As dúvidas foram sanadas. Aproximadamente dois estudantes interagiram. Na metade da aula os estudantes se organizaram

perto de suas duplas ou trios e solicitado que eles falassem a ideia de projeto deles. Após a validação das ideias, os estudantes iniciaram o desenvolvimento dos projetos. Os estudantes tiraram dúvidas e pediram ideias ao professor ao longo do desenvolvimento, porém também interagiram e discutiram as soluções dos problemas com seu grupo. Após finalizada a aula, a apresentação utilizada e as orientações das atividades foram disponibilizados no Google Drive.

Aula 04 - Estruturas de Decisão e Repetição no Scratch

- **Dia e Horário:** 19/05/2022 – 9h às 11h.
- **Presentes:** Quatro estudantes.
- **Conteúdo programático:** Revisão do conteúdo da aula anterior; Definição do projeto; Procedimentos; Variáveis; Desenvolvimento do projeto.
- **Atividade da aula:** Atividade 1 - Criar um projeto no Scratch com pelo menos duas estruturas condicionais. Lembrando que seu programa deve ser diferente dos que foram apresentados em sala de aula. Atividade 2 - Criar um projeto no Scratch com pelo menos duas estruturas de repetição diferentes. Lembrando que seu programa deve ser diferente dos que foram apresentados em sala de aula.
- **Observações:** A aula iniciou 09:05 com a revisão do conteúdo da aula anterior. Em seguida ocorreu a apresentação dos objetivos de aprendizagem. Deu-se início a aula expositiva e prática do conteúdo programado. Ao longo da aula os estudantes foram questionados se haviam dúvidas. As dúvidas foram sanadas. Aproximadamente dois estudantes interagiram. Após a exposição do conteúdo, os estudantes se organizaram em seus grupos e continuaram o desenvolvimento dos projetos. Os estudantes tiraram dúvidas e pediram ideias ao professor ao longo do desenvolvimento, porém também interagiram e discutiram as soluções dos problemas com seu grupo. Após finalizada a aula, a apresentação utilizada e as orientações das atividades foram disponibilizados no Google Drive.

Aula 05 - Processamento de *strings* e Finalização e Apresentação do Projeto

- **Dia e Horário:** 09/06/2022 – 9h às 11h.
- **Presentes:** Quatro estudantes.
- **Conteúdo programático:** Revisão do conteúdo da aula anterior; Processamento de *strings*; Desenvolvimento do projeto; Apresentação do projeto; Avaliação final.
- **Atividade da aula:** Criar um projeto no Scratch que utilize processamento de *strings*.

- **Observações:** A aula iniciou 09:05 com a revisão do conteúdo da aula anterior. Em seguida ocorreu a apresentação dos objetivos de aprendizagem. Deu-se início a aula expositiva e prática do conteúdo programado. Ao longo da aula os estudantes foram questionados se haviam dúvidas. As dúvidas foram sanadas. Aproximadamente dois estudantes interagiram. Após a exposição do conteúdo, os estudantes se organizaram em seus grupos e fizeram os últimos ajustes no projeto. Após finalizados os ajustes nos projetos, foi solicitado que cada grupo fizesse uma apresentação do seu projeto. Primeiro apresentaram o uso e funcionamento e depois os códigos. Após as apresentações, os estudantes puderam socializar e utilizar/jogar o projeto dos demais. O formulário final de avaliação do curso foi enviado a professora para disponibilizar aos estudantes para preenchimento. Após finalizada a aula, a apresentação utilizada e as orientações da atividade foram disponibilizados no Google Drive.

4.2.2.3 Resultados e Discussões

Ao longo da aplicação do curso houve uma desistência considerada esperada. Analisando o percentual e quantidade de presenças por aula, ilustrado na Figura 4.10, pode-se concluir que o percentual (20%) de desistências foi baixo.

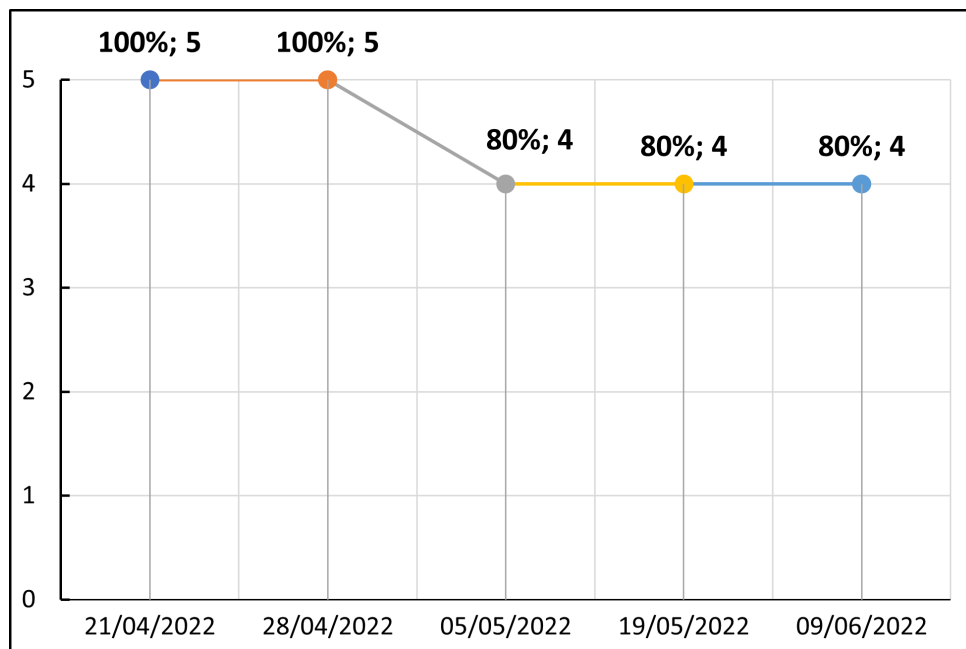
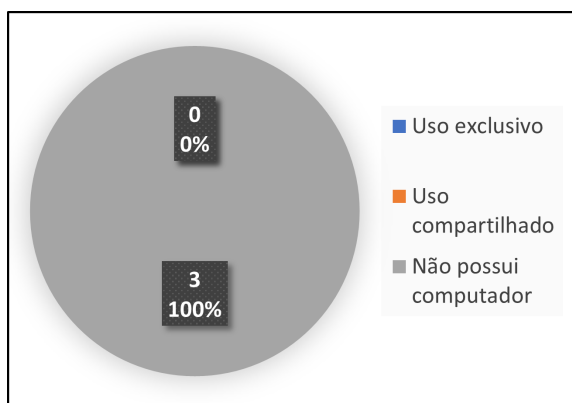
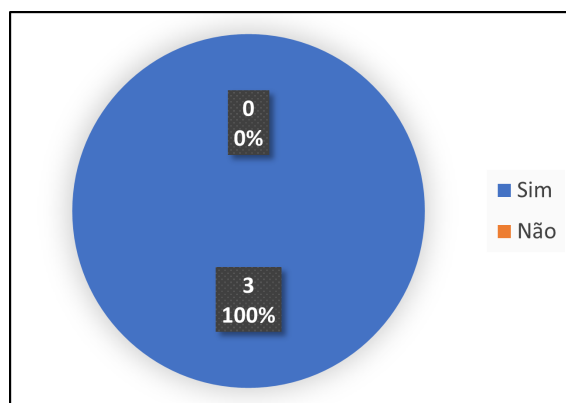


Figura 4.10: Histórico de presença dos estudantes no curso - CEMPF. Elaborado pelo Autor, 2022.

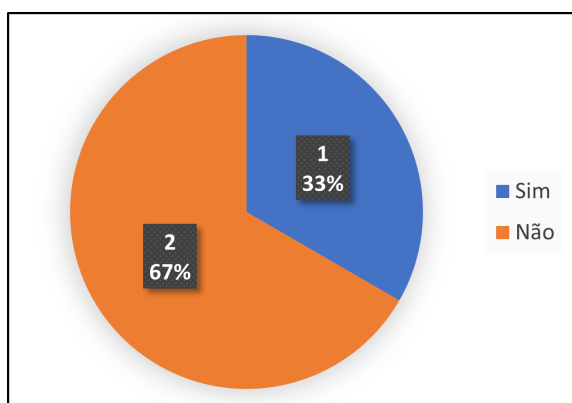
Na primeira aula estava previsto que os estudantes preenchessem o “Formulário de Caracterização do Perfil dos Estudantes” (Apêndice G), porém o laboratório não estava com *internet*, então foi solicitado que os estudantes preenchessem em casa. Apenas três (60%) dos cinco estudantes que finalizaram o curso preencheram o formulário. Os resultados dos dados quantitativos dos três estudantes que preencheram o formulário e finalizaram o curso, estão ilustrados na Figura 4.11.



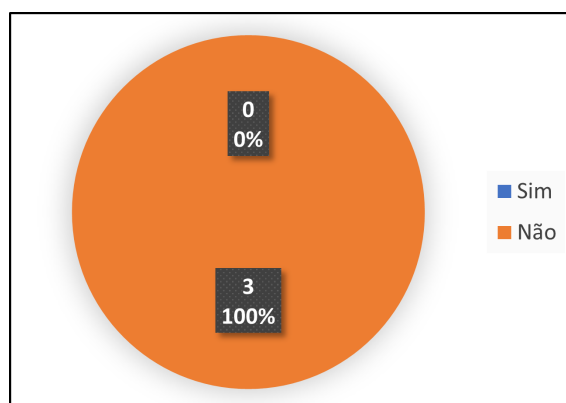
4.11.1: Possui computador?



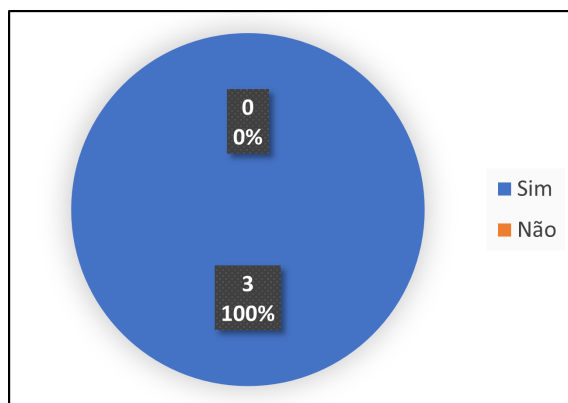
4.11.2: Possui telefone celular?



4.11.3: Possui *internet* móvel?

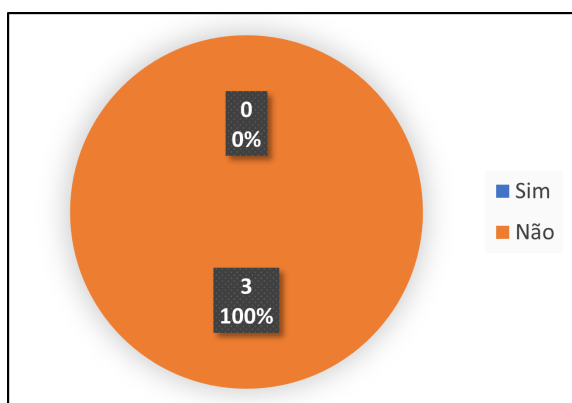


4.11.4: Possui *tablet*?

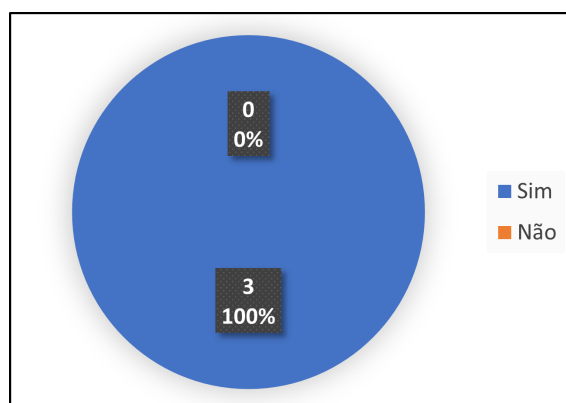


4.11.5: Possui *internet* em casa?

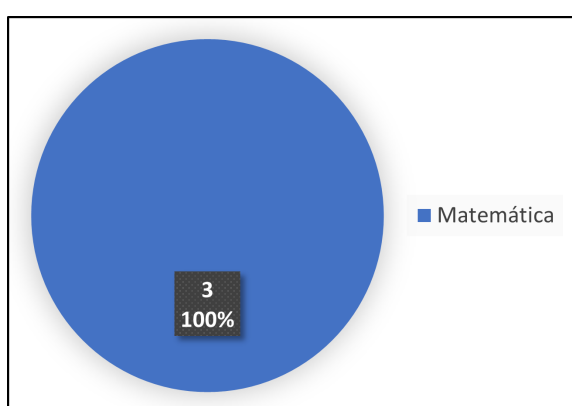
Figura 4.11: Perfil dos estudantes do CEMPF. Elaborado pelo Autor, 2022.



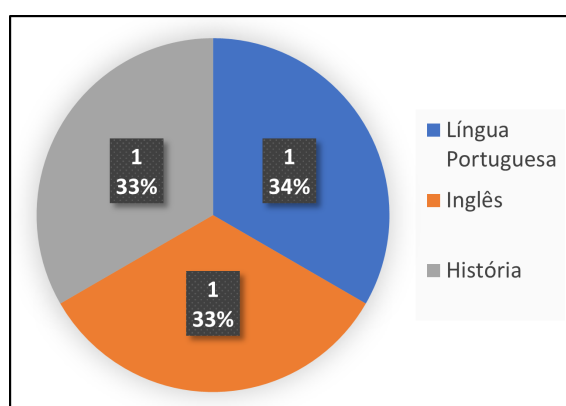
4.11.6: Possui conhecimento em programação?



4.11.7: Se considera bom em raciocínio lógico?



4.11.8: Qual disciplina mais gosta ou tem facilidade de aprender?



4.11.9: Qual disciplina menos gosta ou tem dificuldade de aprender?

Figura 4.11: Continuação: Perfil dos estudantes do CEMPF. Elaborado pelo Autor, 2022.

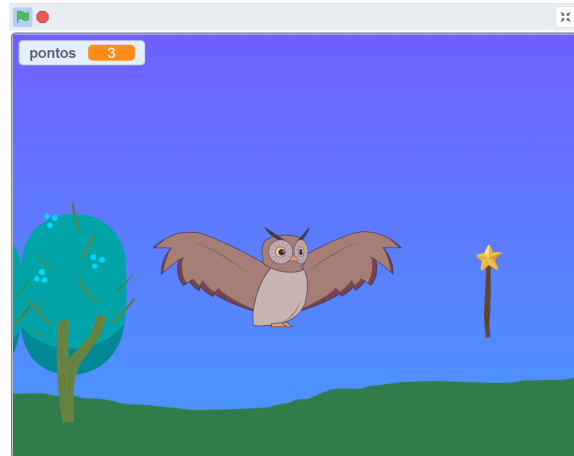
Alguns resultados do perfil dos estudantes que responderam o formulário, merecem destaques. Todos os três (100%) estudantes: não possuíam computador em casa, ou seja, todas as atividades foram desenvolvidas em sala no horário do curso; não possuíam nenhum conhecimento em programação, reforça que o Scratch foi uma ótima escolha para essa escola; se considera bom em raciocínio lógico; e mais gosta/tem facilidade de aprender na disciplina de Matemática, o que pode ter relação com a afinidade na área de Ciências Exatas.

Na aplicação do curso pôde-se perceber que os estudantes estavam engajados na elaboração do projeto e tiravam dúvidas e discutiam sobre as soluções dos problemas. Após o curso foi realizada a análise dos códigos produzidos pelos cinco estudantes (2 projetos) que efetivamente finalizaram o curso e entregaram o projeto final escolhido. Os projetos dos dois grupos foram analisados na ferramenta Dr. Scratch e obtiveram as seguintes notas: Grupo 1 - “Caixa Eletrônico” (estudantes “A”, “D” e “E”) 10 pontos; e Grupo 2 - “Coruja e Varinha” (estudantes “B” e “C”) 11 pontos. A pontuação máxima do Dr. Scratch é de 21 pontos, conforme critérios e pontuações detalhados na Seção 2.4.2, logo

observa-se que os estudantes, em geral, obtiveram notas satisfatórias, considerando que por serem do primeiro ano do Ensino Médio, tiveram mais dificuldades. As telas dos projetos podem ser visualizadas na Figura 4.12 e o detalhamento da avaliação dos projetos na Figura 4.13.

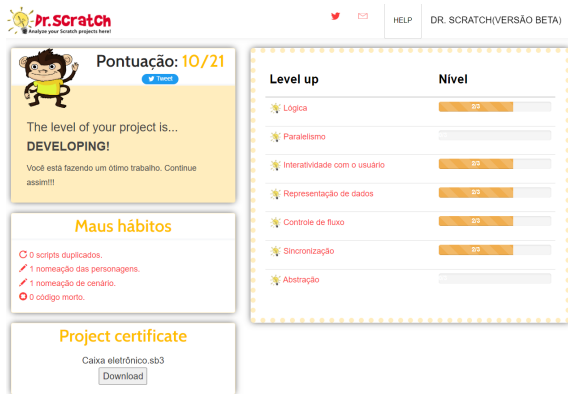


4.12.1: Projeto “Caixa Eletrônico”.

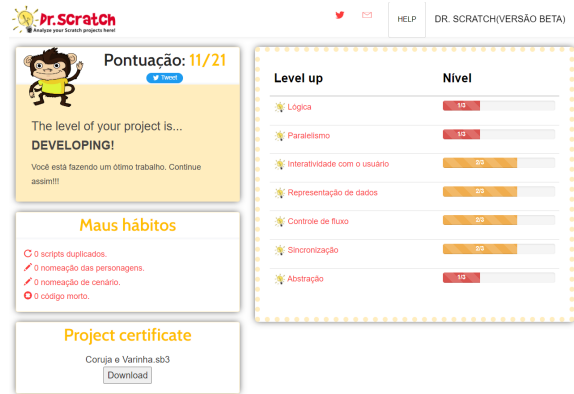


4.12.2: Projeto “Coruja e Varinha”.

Figura 4.12: Telas dos projetos dos estudantes do CEAN. Elaborado pelo Autor, 2022.



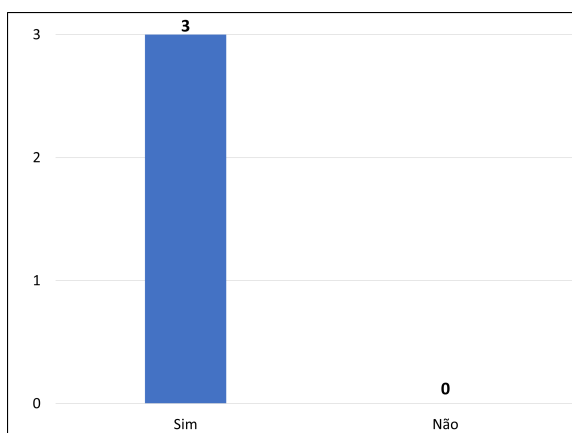
4.13.1: Avaliação Dr. Scratch - Projeto “Caixa Eletrônico”.



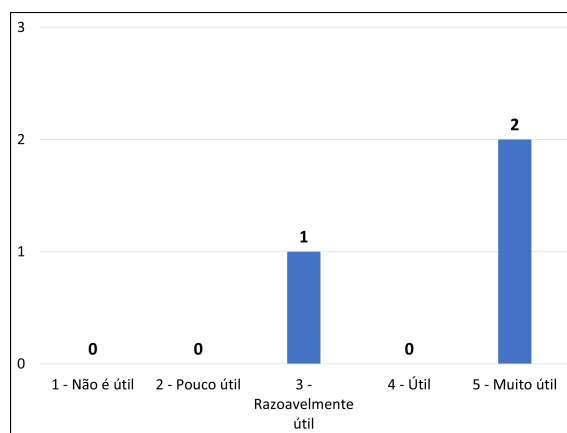
4.13.2: Avaliação Dr. Scratch - Projeto “Coruja e Varinha”.

Figura 4.13: Avaliação dos projetos dos estudantes do CEAN no Dr. Scratch. Elaborado pelo Autor, 2022.

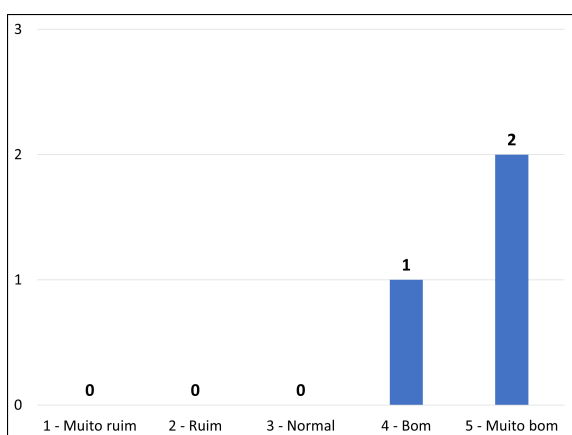
Após a finalização do curso, conforme relatado na quinta aula, os estudantes receberam o questionário final de avaliação (Apêndice H) que teve o objetivo de coletar a percepção dos estudantes em relação ao curso. Os resultados demonstram que os estudantes gostaram do curso, em seus diversos aspectos, conforme ilustrado na Figura 4.14.



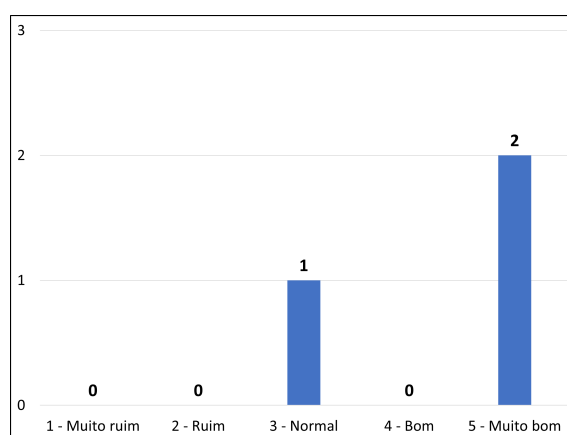
4.14.1: Você gostou do curso?



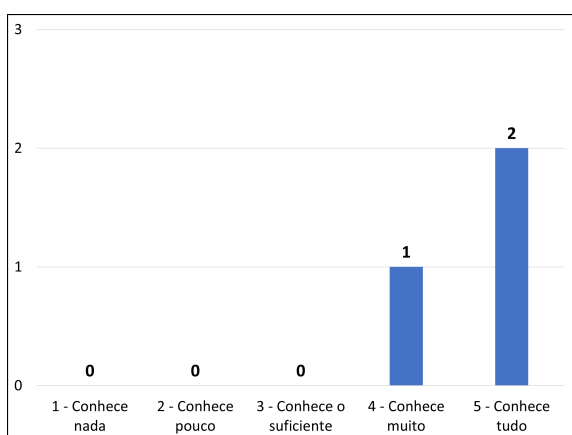
4.14.2: Você considera que a aplicação dos conceitos/temas desse curso em sua vida acadêmica/profissional e pessoal é:



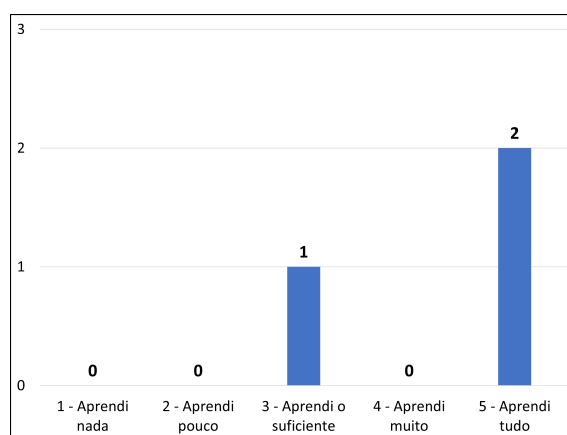
4.14.3: Como você avalia o formato e condução (metodologia) das aulas do curso?



4.14.4: Como você avalia a forma do professor comunicar e aplicar o conteúdo (didática) nas aulas do curso?



4.14.5: Como você avalia o conhecimento que o professor tem sobre o assunto do curso?



4.14.6: Como você avalia a sua aprendizagem ao longo do curso?

Figura 4.14: Avaliação final dos estudantes - CEAN. Elaborado pelo Autor, 2022.

A pergunta sobre o nível de conhecimento em informática dos estudantes foi aplicada tanto no questionário anterior quanto no posterior ao curso, possibilitando uma comparação da percepção do conhecimento geral em informática pelos estudantes. Observa-se que 100% (3) dos estudantes consideraram que, após o curso, seu nível de conhecimento em informática aumentou em média dois pontos, conforme ilustrado na Figura 4.15.

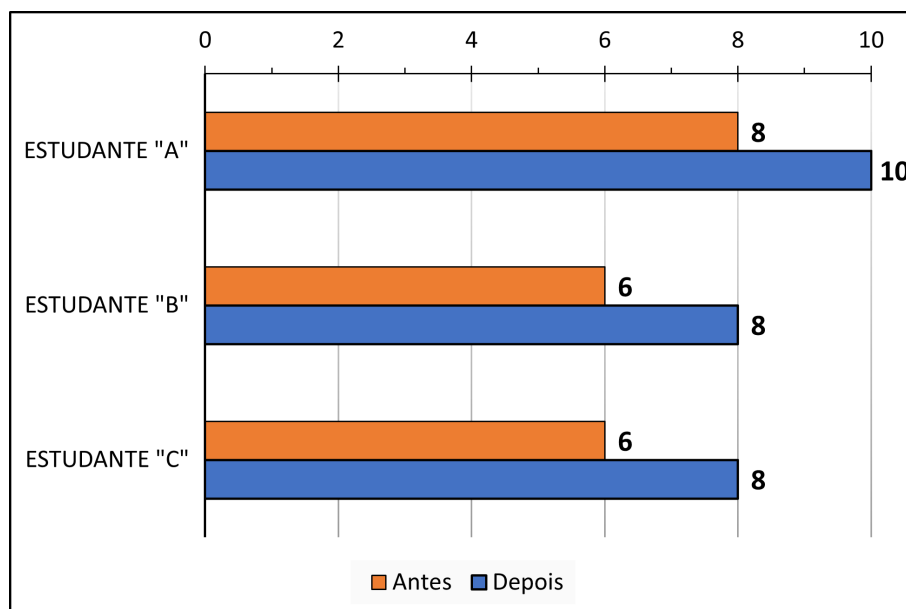


Figura 4.15: Comparativo da percepção dos estudantes quanto ao nível de conhecimento em informática antes e depois do curso - CEMPF. Elaborado pelo Autor, 2022.

As maiores dificuldades encontradas na aplicação dessa turma foram em relação à conexão com a *internet* que atrasou o desenvolvimento do projeto e impossibilitou o preenchimento dos formulários no laboratório.

O curso, em geral, cumpriu com seu objetivo de aprendizagem nessa escola, conforme demonstra as pontuações obtidas nos projetos, bem como os dados analisados dos questionários, apesar de ter tido uma baixa adesão nos preenchimentos.

4.2.2.4 Lições Aprendidas

A partir dessa experiência de aplicação do curso no CEMPF, têm-se as seguintes lições aprendidas:

- O professor precisa criar um ambiente propício a tirar dúvidas para que os estudantes se sintam à vontade para perguntar;
- Os estudantes se sentem mais próximos e acolhidos pelo professor ao ter um diálogo aberto e amigável, favorecendo a aprendizagem e diminuindo as desistências;

- O professor precisa estar preparado para adaptar sua proposta conforme a realidade e contexto social na qual está inserido.
- Apesar de se ter um curso com material didático padrão, é preciso considerar o contexto da turma e fazer algumas adequações, por isso a importância de entender o perfil dos estudantes.

4.3 Resultados, Discussões e Lições Aprendidas das Experiências

As aplicações do curso em duas etapas, três escolas e modalidades diferentes trouxeram uma experiência muito rica, que possibilitou várias análises, discussões e lições aprendidas. Conforme mencionado, a primeira aplicação foi feita no CEM 01 do Paranoá na modalidade de ensino remoto e a segunda aplicação foi feita em duas escolas, CEAN e CEMPF, na modalidade presencial. Essas diferenças de modalidade nas aplicações trouxeram algumas discussões e aprendizados, visto que cada modalidade requer atenção em detalhes diferentes para a efetividade da aplicação do curso. Na Tabela 4.2 são detalhadas as principais percepções entre as diferenças encontradas conforme a modalidade de ensino.

Analisando a segunda aplicação, visto que foi a versão final do curso e que possibilita uma comparação direta entre os dados coletados nas escolas, é possível analisar uma visão geral da aplicação em todas as séries do Ensino Médio, afinal no CEMPF, o público-alvo foi, exclusivamente, estudantes do 1º do Ensino Médio, enquanto no CEAN foi, exclusivamente, estudantes do 2º e 3º ano do Ensino Médio, logo pôde-se obter dados de todas as séries do Ensino Médio, tornando as experiências mais heterogêneas e completas. Essa divisão também possibilitou uma análise de desempenho no desenvolvimento dos projetos, visto que os projetos dos estudantes do 1º ano (CEMPF) obtiveram desempenho inferior aos do 2º e 3º ano (CEAN), conforme ilustrado na Figura 4.16.

As seguintes hipóteses pretendem contextualizar os possíveis motivos que justificam o desempenho inferior dos estudantes do 1º ano: os estudantes do 1º ano (CEMPF) fizeram o curso no turno contrário, ou seja, tinham que sair de casa, exclusivamente, para participar do curso pela manhã, o que pode ter gerado desmotivação, enquanto os estudantes do 2º e 3º ano (CEAN) ficavam após a aula para participar do curso (vespertino); E os estudantes do 1º ano possuem mais dificuldades do que os do 2º e 3º ano, visto que acabaram de sair do Ensino Fundamental e algumas habilidades e competências requeridas para desenvolvimento do PC são mais complexas de serem adquiridas.

Tabela 4.2: Percepções entre as diferenças das modalidades de ensino das experiências. Elaborado pelo Autor, 2022.

Ensino Remoto	Presencial
Menor participação dos estudantes	Maior participação dos estudantes
Dificuldade de desenvolvimento do projeto em grupos	Facilidade de desenvolvimento do projeto em grupos
Dificuldade de identificar se os estudantes possuem dúvidas	Maior facilidade de identificar os estudantes possuem dúvidas
Distanciamento da relação estudante-professor	Proximidade da relação estudante-professor
Maior quantidade de desistências	Menor quantidade de desistências

Conforme ilustrado na Figura 4.16, nenhum dos grupos conseguiram atingir a pontuação máxima (21) no projeto, o que significa que algumas lacunas no processo de ensino-aprendizagem, possivelmente, não foram preenchidas e podem ser exploradas em trabalhos futuros, porém é complexo analisar quais são e como preenchê-las, visto que pode ter sido motivado por vários fatores controláveis e incontroláveis, internos e externos, à aplicação do curso. Ao analisar as habilidades avaliadas pelo Dr. Scratch, a Figura 4.17, observa-se que “Abstração”, “Lógica” e “Paralelismo” foram as três habilidades que os estudantes mais tiveram dificuldades de implementar no projeto, levando às hipóteses de investigação de quais as habilidades que o curso não abordou como deveria ou que os estudantes possuem mais dificuldades de aprender, independente da abordagem.

Em ambas as escolas (CEMPF e CEAN) da segunda aplicação foram enfrentadas dificuldades em relação aos computadores e conexão com a *internet*, o que evidencia que possivelmente não há investimento em manutenções corretivas ou preventivas nos laboratórios, o que foi confirmado em conversas informais com os estudantes e professores das escolas. Essa situação é bastante preocupante, visto que a BNCC [1] enfatiza que as propostas de trabalho para o Ensino Médio precisam priorizar o acesso aos saberes sobre o mundo digital e as práticas da cultura digital, visto que possui impactos nos vários campos de atuação social e despertam o interesse e identificação com as TDIC.

O investimento em recursos tecnológicos e de conexão, trariam várias possibilidades e ferramentas para os professores de toda a escola, para o desenvolvimento das competências e habilidades objetivadas pela BNCC para o Ensino Médio. Além disso, o investimento proporcionaria a inclusão e letramento digital, visto que, conforme observado nos resultados do formulário de caracterização do perfil dos estudantes das duas escolas (Apêndice G), nove (75%) dos doze (100%) estudantes não possuíam computador ou eram de uso compartilhado com seus familiares.

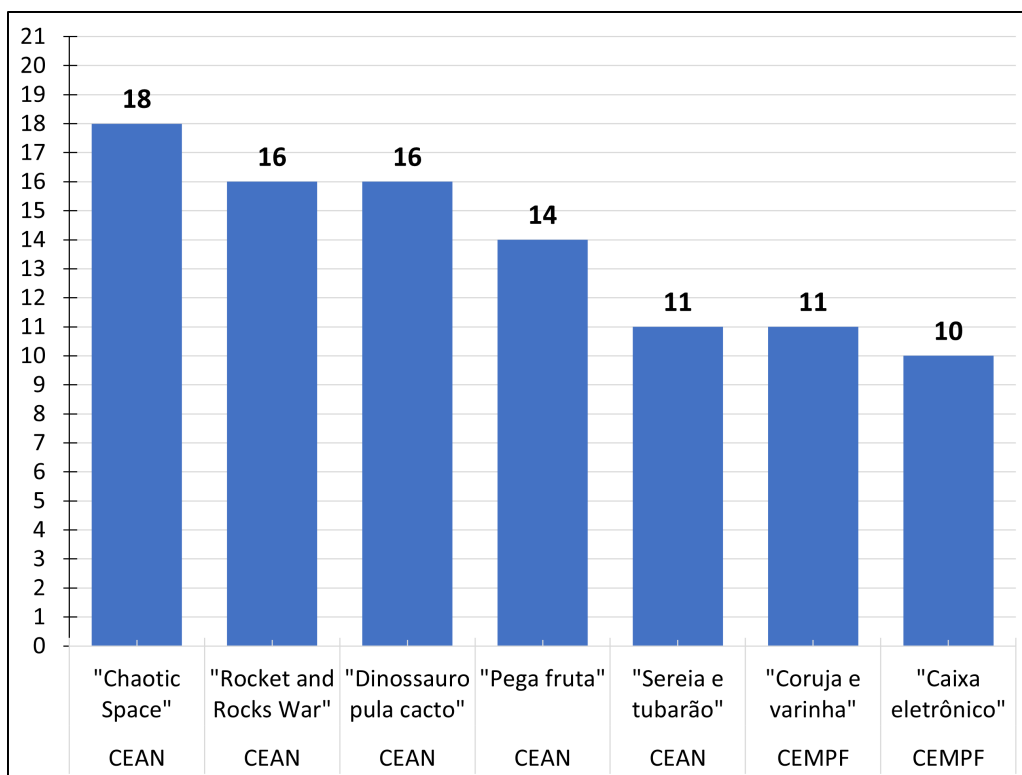


Figura 4.16: Pontuação por projeto por escola - CEAN e CEMPF. Elaborado pelo Autor, 2022.

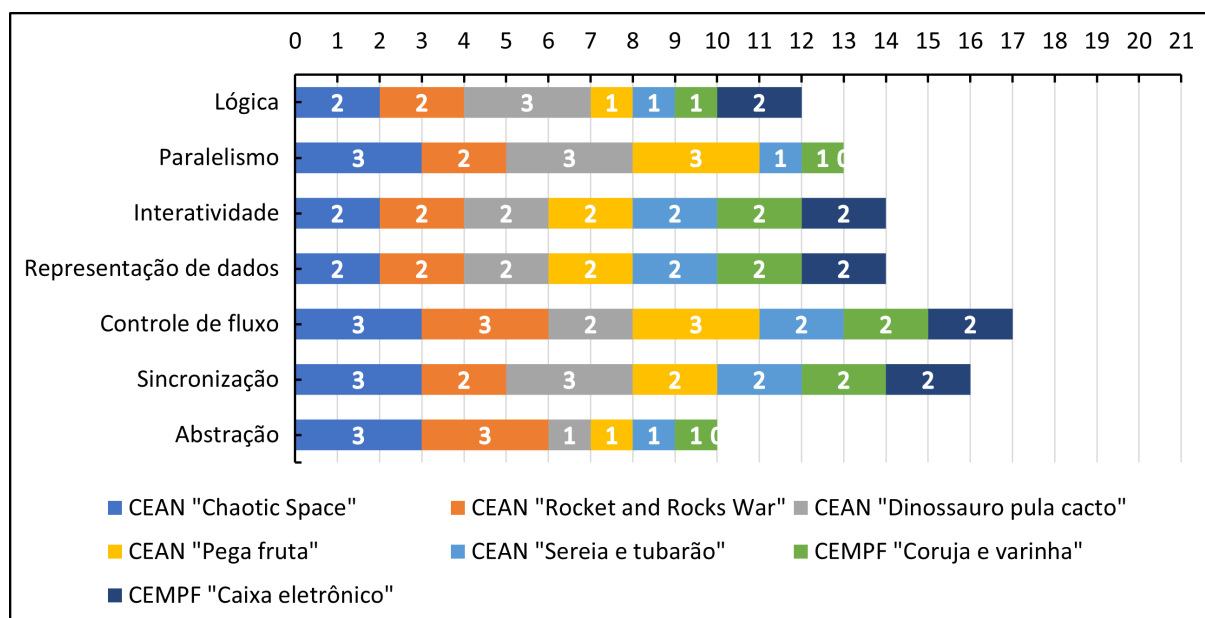


Figura 4.17: Comparação dos projetos por habilidade analisada no Dr. Scratch. Elaborado pelo Autor, 2022.

Outro dado importante e que merece destaque é que oito (67%) dos doze (100%) estudantes relataram que gostam/tem mais facilidade de aprender Matemática, o que pode levar a hipótese da relação de afinidade dos estudantes com Ciências Exatas.

Os estudantes de ambas as escolas relataram que gostaram do curso, em diversos aspectos, e consideraram que a aplicação do curso contribuiu para o aumento do conhecimento em informática, com uma média de mais dois pontos em suas percepções, conforme ilustrado na Figura 4.9 e na Figura 4.15.

As principais lições aprendidas a partir das experiências de aplicação do curso nas escolas são detalhadas a seguir:

- O professor precisa criar um ambiente propício a tirar dúvidas para que os estudantes se sintam à vontade para perguntar, independente da modalidade de ensino.
- Os estudantes se sentem mais próximos e acolhidos pelo professor ao ter um diálogo aberto, amigável e sendo presencial, favorecendo a aprendizagem e diminuindo as desistências.
- O professor precisa estar preparado para adaptar sua proposta conforme a realidade e contexto social na qual está inserido, independente da modalidade de ensino.
- Apesar de se ter um curso com material didático padrão, é preciso considerar o contexto da turma e fazer algumas adequações, por isso a importância de entender o perfil dos estudantes.
- Os estudantes gostam de produzir produtos criativos, logo é preciso dar liberdade, mesmo que guiada, para que os estudantes desenvolvam um projeto que seja representativo para a vivência de ensino-aprendizagem.

Capítulo 5

Conclusões e Trabalhos futuros

Neste trabalho foi elaborada e testada uma metodologia para ensinar o Pensamento Computacional nas escolas públicas de Ensino Médio no Distrito Federal. O Pensamento Computacional e o Scratch foram conceituado e contextualizado na Educação Básica Brasileira, especialmente no Ensino Médio.

O curso intitulado “Aprenda Pensamento Computacional e Scratch” foi elaborado, segundo a literatura e contexto local, e aplicado em duas etapas e três escolas. Os artefatos do curso estão disponíveis para consulta e utilização nas futuras aplicações.¹

Os dados gerados na metodologia de avaliação da aprendizagem dos estudantes comprovou que o curso cumpriu com o seu objetivo, considerando as notas dos projetos desenvolvidos e avaliados no Dr. Scratch. Os dados coletados a partir da percepção dos estudantes sobre o curso mostram que o curso recebeu a média de 4, em uma escala de 1 a 5, em diversas dimensões de análise do curso, o que demonstra que os estudantes gostaram do curso, em diversas dimensões de avaliação.

Conforme demonstrado nas aplicações do curso e relatadas no Capítulo 4, mesmo que as escolas possuam infraestrutura de computadores e *internet*, ela pode ser precária e não suportar tantas conexões, logo isso pode ser um limitante e deve ser analisado antes de iniciar uma aplicação do curso em alguma escola.

As perspectivas de trabalhos futuros a partir deste trabalho são:

- Aplicar o curso em outras escolas públicas para serem gerados mais dados para analisá-los;
- Aprimorar o curso com base nas aplicações e coletas e análises de dados futuros;
- Elaborar uma versão para aplicação do curso em escolas sem infraestrutura adequada para aplicação do curso.

¹Material didático disponível em:
https://github.com/viniciusaguiarmonteiro/TCC_UnB_Vinicius_Aguiar_Monteiro.git.

Referências

- [1] BRASIL, Ministério da Educação: *Base Nacional Comum Curricular*. Brasília, DF, 2018. <http://basenacionalcomum.mec.gov.br/>, acesso em 14-07-2022. ix, 1, 4, 6, 7, 13, 14, 15, 24, 69
- [2] Davidson, Susan e Chris Murphy: *Computational thinking for problem solving*. University of Pennsylvania, 2022. <https://www.coursera.org/learn/computational-thinking-problem-solving>, acesso em 24-07-2022. ix, xiii, 9, 10, 11, 25, 27
- [3] BRASIL, Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (Inep): *A redação do ENEM 2020 - Cartilha do Participante*. Brasília, DF: INEP, 2020. https://download.inep.gov.br/publicacoes/institucionais/avaliacoes_e_exames_da_educacao_basica/a_redacao_do_enem_2020_-_cartilha_do_participante.pdf, acesso em 24-07-2022. ix, 9, 10
- [4] Scratch: *Scratch wiki*. Scratch, 2022. <https://en.scratch-wiki.info/wiki/>, acesso em 06-08-2022. ix, xiii, 17, 18, 19, 20, 21
- [5] CSTA, ISTE: *Computational Thinking in K-12 Education: teacher resources*. ISTE, 2ª edição, 2011. https://cdn.iste.org/www-root/2020-10/ISTE_CT_Teacher_Resources_2ed.pdf, acesso em 06-08-2022. xiii, 7, 8
- [6] Moreno-León, Jesús, Gregorio Robles e Marcos Román-González: *Dr. scratch: Automatic analysis of scratch projects to assess and foster computational thinking*. RED-Revista de Educación a Distancia, setembro 2015. https://www.researchgate.net/publication/281714025_Dr_Scratch_Automatic_Analysis_of_Scratch_Projects_to_Assess_and_Foster_Computational_Thinking, acesso em 26-08-2022. xiii, 22, 23
- [7] Aono, Alexandre, Hugo Rody, Daniela Musa, Vanessa Pereira e Jurandy Almeida: *A utilização do scratch como ferramenta no ensino de pensamento computacional para crianças*. Em *Anais do XXV Workshop sobre Educação em Computação*, Porto Alegre, RS, Brasil, 2017. SBC. <https://sol.sbc.org.br/index.php/wei/article/view/3556>, acesso em 06-10-2021. xiii, 25, 27
- [8] Wing, Jeannette M.: *Computational thinking*. Commun. ACM, 49(3):33-35, mar 2006, ISSN 0001-0782. <https://doi.org/10.1145/1118178.1118215>, acesso em 14-07-2022. 1, 15

- [9] Vieira, Sebastião da Silva e Marcelo Sabbatini: *Pensamento computacional através do scratch numa perspectiva maker*. REVISTA INTERSABERES, 16(37):43–63, 2021, ISSN 1809-7286. <http://dx.doi.org/10.22169/revint.v16i37.1933>, acesso em 31-07-2022. 1, 17
- [10] Lye, Sze Yee e Joyce Hwee Ling Koh: *Review on teaching and learning of computational thinking through programming: What is next for k-12?* Computers in Human Behavior, 41:51–61, 2014, ISSN 0747-5632. <https://www.sciencedirect.com/science/article/pii/S0747563214004634>, acesso em 06-08-2022. 1
- [11] Majed, Marji: *Learn to Program with Scratch*. Novatec Editora, 2014, ISBN 978-85-7522-312-3. 2
- [12] Avila, Christiano, Adriana Bordini, Monica Marques e Luciana Cavalheiro, Simone e Foss: *Desdobramentos do pensamento computacional no brasil*. Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação - SBIE), 27(1):200, 2016, ISSN 2316-6533. <http://ojs.sector3.com.br/index.php/sbie/article/view/6700>, acesso em 31-07-2022. 2, 16
- [13] BRASIL: *Lei nº 9.394, de 20 de dezembro de 1996 - estabelece as diretrizes e bases da educação nacional*. Diário Oficial da República Federativa do Brasil, 1996. http://www.planalto.gov.br/ccivil_03/leis/19394.htm, acesso em 14-07-2022. 6
- [14] Cuny, Jan, Larry Snyder e Jeannette M. Wing: *Research notebook: Computational thinking—what and why?* The Link, página 20–23, 2011. https://www.cs.cmu.edu/sites/default/files/11-399_The_Link_Newsletter-3.pdf, acesso em 14-07-2022. 7
- [15] Brackmann, Christian, Dante Barone, Ana Casali, Rafael Boucinha e Susana Muñoz-Hernandez: *Computational thinking: Panorama of the americas*. 2016 International Symposium on Computers in Education (SIIE), páginas 1–6, 2016. <https://ieeexplore.ieee.org/abstract/document/7751839>, acesso em 24-07-2022. 9, 10, 11, 13
- [16] Tabesh, Yahya: *Computational thinking: A 21st century skill*. Olympiads in Informatics, 11(2):65–70, 2017. https://ioinformatics.org/journal/v11si_2017_65_70.pdf, acesso em 24-07-2022. 9, 10, 11
- [17] Ascencio, Ana F. G. e Edilene A. V. de Campos: *Fundamentos da Programação de Computadores*. Person, 2008. 11
- [18] André, Claudio F: *O pensamento computacional como estratégia de aprendizagem, autoria digital e construção da cidadania*. teccogs - Revista Digital de Tecnologias Cognitivas, 18, jul/dez 2018. <https://revistas.pucsp.br/index.php/teccogs/article/view/48579/32061>, acesso em 27-07-2022. 12
- [19] Bombasar, James, André Raabe, Elisangela Miranda e Rafael Santiago: *Ferramentas para o ensino-aprendizagem do pensamento computacional: onde está alan turing?* Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática

- na Educação - SBIE), 26(1):81, 2015, ISSN 2316-6533. <http://ojs.sector3.com.br/index.php/sbie/article/view/5120>, acesso em 06-08-2022. 16
- [20] Khouri, Cátia Mesquita Brasil, Gidevaldo Novais dos Santos e Maria Silva Santos Barbosa: *Mapeamento sistemático em metodologias de ensino-aprendizagem de programação*. Revista de Ciência da Computação, 2(1):13–27, maio 2020. <https://periodicos2.uesb.br/index.php/recic/article/view/6669>, acesso em 13-08-2022. 16
- [21] Dias, Klissiomara e Miquéia Serrão: *A linguagem scratch no ensino de programação: Um relato de experiência com alunos iniciantes do curso de licenciatura em computação*. Em *Anais do XXII Workshop sobre Educação em Computação*, páginas 189–198, Porto Alegre, RS, Brasil, 2014. SBC. <https://sol.sbc.org.br/index.php/wei/article/view/10973>, acesso em 09-08-2022. 17
- [22] Brennan, Karen e Mitchel Resnick: *New frameworks for studying and assessing the development of computational thinking*. Em *In AERA 2012*, 2012. <http://scratched.gse.harvard.edu/ct/files/AERA2012.pdf>, acesso em 20-08-2022. 21
- [23] Araujo, Ana, Wilkerson Andrade e Dalton Guerrero: *Um mapeamento sistemático sobre a avaliação do pensamento computacional no brasil*. Anais dos Workshops do Congresso Brasileiro de Informática na Educação, 5(1):1147, 2016, ISSN 2316-8889. <http://ojs.sector3.com.br/index.php/wcbie/article/view/7040>, acesso em 26-08-2022. 21, 22
- [24] Alves, Nathalia da Cruz, Christiane Gresse Von Wangenheim e Jean C.R. Hauck: *Approaches to assess computational thinking competences based on code analysis in k-12 education: A systematic mapping study*. Informatics in Education, 18(1):17–39, 2019, ISSN 1648-5831. <https://files.eric.ed.gov/fulltext/EJ1212844.pdf>, acesso em 06-08-2022. 22
- [25] Oliveira, Milena de, Anderson de Souza, Aline Ferreira e Emanuel Barreiros: *Ensino de lógica de programação no ensino fundamental utilizando o scratch: um relato de experiência*. Em *Anais do XXII Workshop sobre Educação em Computação*, páginas 239–248, Porto Alegre, RS, Brasil, 2014. SBC. <https://sol.sbc.org.br/index.php/wei/article/view/10978>, acesso em 28-08-2022. 24, 28
- [26] Schoeffel, Pablo, Paolo Moser, Geraldo Varela, Letícia Durigon, Gustavo Albuquerque e Matheus Niquelatti: *Uma experiência no ensino de pensamento computacional para alunos do ensino fundamental*. Anais dos Workshops do Congresso Brasileiro de Informática na Educação, 4(1):1474, 2015, ISSN 2316-8889. <http://ojs.sector3.com.br/index.php/wcbie/article/view/6333>, acesso em 20-08-2022. 24, 28
- [27] Marinho, Anna Raquel da S., Pauleany S. de Moraes, Givanaldo R. de Souza e Alba S. L. do Nascimento: *Relato de experiência vivenciada no pibid sobre a utilização da computação desplugada, a hora do código e do scratch no ensino médio*. Em *Anais do XXVI Workshop sobre Educação em Computação*, Porto Alegre, RS, Brasil, 2018.

SBC. <https://sol.sbc.org.br/index.php/wei/article/view/3507>, acesso em 28-08-2022. 24, 28

- [28] Costa, Newerlyson Correia, José Ricardo Santos de Lima, Gylmara Kylma Feitosa Carvalhêdo Almeida, Yonara Costa Magalhães e Will Ribamar Mendes Almeida: *O uso da plataforma scratch como ferramenta facilitadora durante o ensino de lógica de programação para alunos do ensino médio: The use of the scratch platform as a facilitating tool during the teaching of programming logic for high school students*. Brazilian Journal of Development, 8(8):59279–59293, Aug. 2022. <https://www.brazilianjournals.com/ojs/index.php/BRJD/article/view/51498>, acesso em 26-08-2022. 24, 28
- [29] Harvard, Creative Computing Lab: *Creative computing curriculum*. Desenvolvido por Laboratório de Computação Criativa da Escola de Graduação em Educação de Harvard e Traduzido por Rede Brasileira de Aprendizagem Criativa (aprendizagemcriativa.org) e Universidade Estadual de Feira de Santana (UEFS), Brasil, 2019. <https://lcl.media.mit.edu/resources/readings/creative-computing-guide.pt.pdf>, acesso em 06-10-2021. 25
- [30] Freeman, Scott, Sarah L. Eddy, Miles McDonough, Michelle K. Smith, Nnadozie Okoroafor, Hannah Jordt e Mary Pat Wenderoth: *Active learning increases student performance in science, engineering, and mathematics*. Proceedings of the National Academy of Sciences, 111(23):8410–8415, 2014. <https://www.pnas.org/doi/abs/10.1073/pnas.1319030111>, acesso em 16-08-2022. 26
- [31] Michael, Joel: *Where's the evidence that active learning works?* Advances in Physiology Education, 30(4):159–167, 2006. <https://doi.org/10.1152/advan.00053.2006>, acesso em 16-08-2022, PMID: 17108243. 26

Apêndice A

Planos de aula: Aprenda Pensamento Computacional e Scratch



UnB

Instituto de
Ciências Exatas

Departamento de
Ciência da Computação

Planos de Aula:

Aprenda Pensamento Computacional e Scratch

Docente orientadora:

Prof.a Dr.a Maristela Terto de Holanda

Discente:

Vinícius Aguiar Monteiro – 16/0072727

Sumário

1. Introdução	2
1.1. Público-alvo	2
1.2. Objetivos	2
1.3. Estrutura do Curso.....	2
2. Planos de aula.....	3
2.1. AULA 01 - Pensamento Computacional	4
2.2. AULA 02 - Introdução à Programação e ao Scratch.....	6
2.3. AULA 03 - Procedimentos e Variáveis no Scratch	9
2.4. AULA 04 - Estruturas de Decisão e Repetição no Scratch.....	11
2.5. AULA 05 - Processamento de Strings e Finalização e Apresentação do Projeto	13

1. Introdução

Este documento descreve as aulas que compõem o curso intitulado “**Aprenda Pensamento Computacional e Scratch**” que foi desenvolvido como parte do Trabalho de Conclusão do Curso de Licenciatura em Computação na Universidade de Brasília. O curso propõe o ensino do Pensamento Computacional (PC) (pensar como um cientista da computação) e da Linguagem de Programação Visual (LPV) Scratch, de maneira inclusiva, para os estudantes de escolas públicas do ensino médio do Distrito Federal, possibilitando aos estudantes o desenvolvimento das competências e habilidades do Pensamento Computacional e as pretendidas pela Base Nacional Comum Curricular (BNCC) para o ensino médio.

1.1. Público-alvo

O público-alvo deste curso são estudantes do ensino médio público do Distrito Federal. O único pré-requisito para participação é que o estudante seja alfabetizado. A metodologia aplicada será aulas expositivas e práticas em formato de exercícios dos conteúdos aprendidos.

O curso tem carga horária prevista para 10 horas. Cada aula possui duração de 2 horas a serem ministradas presencialmente em um laboratório de informática.

1.2. Objetivos

O objetivo geral do curso é o ensino dos conceitos do Pensamento Computacional (Coleta de dados, análise de dados, representação de dados, decomposição de problemas, abstração, algoritmo e procedimentos, automação, simulação e paralelismo), utilizando o Scratch, para estudantes do ensino médio público do Distrito Federal.

Os objetivos específicos são:

Ensinar os conceitos do Pensamento Computacional e de seus pilares.

Contextualizar os pilares do Pensamento Computacional à realidade dos estudantes.

Ensinar sobre as diferentes formas de representação de algoritmo (pseudocódigo, fluxograma e Scratch).

Ensinar sobre a utilização da ferramenta Scratch.

Ensinar os conceitos e ferramentas do Scratch.

Estimular o aprendizado de linguagens de programação no ensino médio.

Expandir as competências habilidades em lógica de programação.

Estimular o desenvolvimento criativo utilizando jogos.

1.3. Estrutura do Curso

O curso está dividido em dois módulos. O módulo I é referente aos conceitos de Pensamento Computacional e o módulo II é referente ao Scratch.

2. Planos de aula

Os planos de aula dos cinco encontro será detalhado a seguir:

2.1. AULA 01 - Pensamento Computacional

Informações do curso:

Curso:	Aprenda Pensamento Computacional e Scratch
Tema:	Pensamento Computacional
Carga horária:	2 horas

Objetivos:

Apresentar o curso e conhecer os estudantes
 Apresentar o significado de Pensamento Computacional
 Descrever os quatro pilares para o pensamento computacional
 Apresentar Conceitos de fluxograma e exemplos
 Apresentar Conceitos de pseudocódigo e exemplos

Conteúdos:

- Introdução do curso e ambientação
- Conhecendo os estudantes
- Pensamento computacional
- Pilares do pensamento computacional
- Fluxograma
- Pseudocódigo

Procedimentos:

Conteúdo	Tempo previsto	Detalhamento	Material utilizado
Introdução do curso e ambientação	0:20	O professor irá se apresentar, apresentar o projeto e o curso que irá ministrar e seu detalhamento. Pedirá aos estudantes que se apresentem com o Nome, a Idade, a Turma e o que os motivou a participar do curso. Este momento tem o objetivo de aproximar os estudantes e conhecê-los melhor para conduzir as aulas.	Slide "Aula 1 - Pensamento Computacional"
Conhecendo os estudantes	0:05	Os estudantes irão preencher o formulário de caracterização do perfil da turma para subsidiar o conteúdo e atividades das próximas aulas.	Questionário "Perfil dos Estudantes"
Pensamento computacional	0:20	Exposição teórica e apresentação do conceito do Pensamento Computacional.	Slide "Aula 1 - Pensamento Computacional"
Pilares do pensamento computacional	0:25	Exposição teórica e apresentação dos pilares do Pensamento Computacional, além de dar exemplos de como aplicá-los na resolução de problemas. A cada explicação de um dos quatro pilares do Pensamento Computacional, é solicitado aos estudantes que deem outros exemplos dos pilares aplicados à vida.	

Fluxograma	0:25	Exposição teórica e apresentação do conceito de Fluxograma como ferramenta de construção do pilar "Algoritmo" do Pensamento Computacional. São apresentados os símbolos e significados das estruturas que formam os fluxogramas. São demonstrados os fluxogramas com exemplos de atividades do cotidiano. Ao final, é passada uma atividade extraclasse sobre construção de fluxogramas.
Pseudocódigo	0:25	Exposição teórica e apresentação do conceito de Pseudocódigo como ferramenta mais avançada e estruturada de construção do pilar "Algoritmo" do Pensamento Computacional. É apresentada a estrutura do pseudocódigo. Exemplificar problemas do cotidiano, resolvidos em fluxogramas e com pseudocódigos, trazendo assim uma comparação direta entre as duas formas de resolver um problema. Ao final, é passada uma atividade extraclasse sobre construção de fluxogramas e pseudocódigos.

Avaliação:

A avaliação ocorrerá durante a apresentação dos exemplos dos estudantes. Deve-se observar não só os estudantes que tenham conhecimentos prévios do assunto, mas também o comportamento dos estudantes, como respeito pelo professor, interesse no conteúdo e interação quando solicitado. É extremamente importante que nesse momento o professor observe o contexto social dos estudantes e seus conhecimentos no uso de tecnologias, para que nas próximas aulas futuras possa citar exemplos de conceitos de programação aplicados às situações conhecidas dos estudantes, estimulando a aprendizagem significativa.

Referências:

AONO, Alexandre Hild; RODY, Hugo Vianna Silva; MUSA, Daniela Leal; PEREIRA, Vanessa Andrade; ALMEIDA, Jurandy. A Utilização do Scratch como Ferramenta no Ensino de Pensamento Computacional para Crianças. In: WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO (WEI), 25. , 2017, São Paulo. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2017 . ISSN 2595-6175. DOI: <https://doi.org/10.5753/wei.2017.3556>.

Davidson, Susan e Chris Murphy: Computational thinking for problem solving. University of Pennsylvania, 2022. <https://www.coursera.org/learn/computational-thinking-problem-solving/>, acesso em 2022-07-24.

Lab, Creative Computing: Creative computing curriculum. Desenvolvido por Laboratório de Computação Criativa da Escola de Graduação em Educação de Harvard e Traduzido por Rede Brasileira de Aprendizagem Criativa (aprendizagemcriativa.org) e Universidade Estadual de Feira de Santana (UEFS), Brasil, 2019. <https://lcl.media.mit.edu/resources/readings/creative-computing-guide.pt.pdf>, acesso em 2021-10-06.

2.2. AULA 02 - Introdução à Programação e ao Scratch

Informações do curso:

Curso:	Aprenda Pensamento Computacional e Scratch
Tema:	Scratch
Carga horária:	2 horas

Objetivos:

- Apresentar a programação para resolução de tarefas cotidianas para os alunos, o ambiente de programação Scratch e os diferentes tipos de comandos para criação de cenários
- Apresentar os conceitos de algoritmos, descrição narrativas de algoritmos, efeitos sonoros e visuais disponíveis no Scratch, utilização de camadas e criação de cenas com animações

Conteúdos:

- Revisão do conteúdo da aula anterior
- Introdução à programação
- Criando conta no Scratch
- Explorando o Scratch
- Atividade 10 blocos
- Apresentação do Scratch
- Conceitos de algoritmos
- Criação de figuras, efeitos visuais e sonoros no Scratch
- Discussão sobre o projeto

Procedimentos:

Conteúdo	Tempo previsto	Detalhamento	Material utilizado
Revisão do conteúdo da aula anterior	0:10	O professor fará uma breve exposição oral com uma revisão dos conceitos (Pensamento Computacional e seus pilares, fluxograma e pseudocódigo) passado na aula anterior com o objetivo de fixar o conteúdo e tirar alguma dúvida que tenha surgido.	
Introdução à programação	0:20	Exposição teórica e apresentação do conceito de programação.	Slide "Aula 2 - Introdução ao Scratch"
Criando conta no Scratch	0:10	O professor orientará os estudantes a realizarem o cadastro no Scratch. O professor deverá, antes da aula, preparar uma turma no Scratch com o recurso para educadores e disponibilizar o link da turma para os estudantes realizarem o cadastro no link disponibilizado, assim o professor terá controle das atividades e poderá propor atividades para todos.	Slide "Aula 2 - Introdução ao Scratch" e Site https://scratch.mit.edu/

Explorando o Scratch	0:10	O professor deixará os estudantes livres para explorar a ferramenta para se familiarizarem e, talvez, até arriscarem a criar algo no Scratch. O objetivo de pedir que eles explorem a ferramenta sem antes ter explicado o que é cada coisa é para que se sintam desafiados a explorar cada botão e funcionalidade.	Site https://scratch.mit.edu/
Atividade 10 blocos	0:10	O professor orientará os estudantes a realizarem a atividade prática proposta (10 blocos), após a exploração do Scratch e disponibilizará uma atividade, dentro da turma, no Scratch para compartilhamento dos estudantes após a finalização. Por ser o primeiro contato com a ferramenta, é esperado que os estudantes tenham dificuldades e apresentem dúvidas de como fazer determinadas ações, porém o professor apenas observará o que farão no Scratch.	Slide "Aula 2 - Introdução ao Scratch" e Site https://scratch.mit.edu/
Apresentação do Scratch	0:10	Exposição prática e apresentação do site da Linguagem de Programação Visual Scratch, navegando no site e mostrando uma visão geral das funcionalidade e estrutura do Scratch para os estudantes.	Slide "Aula 2 - Introdução ao Scratch" e Site https://scratch.mit.edu/
Conceitos de algoritmos	0:20	Exposição teórica e apresentação dos conceitos de algoritmo descritivo e pseudocódigo, mostrando exemplos práticos de detalhamento de um problema.	Slide "Aula 2 - Introdução ao Scratch"
Criação de figuras, efeitos visuais e sonoros no Scratch	0:20	Exposição teórica, prática e apresentação de como criar figuras, efeitos visuais e sonoros no Scratch utilizando um dos exemplos dos tutoriais disponíveis no Scratch. O professor orientará os estudantes a realizarem a atividade prática proposta (seguir um dos tutoriais do Scratch).	Slide "Aula 2 - Introdução ao Scratch" e Site https://scratch.mit.edu/
Discussão sobre o projeto	0:10	Nesse momento os estudantes são organizados em duplas ou trios para discutirem as ideias dos projetos e mediará um debate em grupo para mostrar as possibilidades de desenvolvimento do projeto como jogos, histórias, animações, artes interativas, etc.), além de verificar se já pensaram em algum projeto para desenvolver nas próximas aulas.	Slide "Aula 2 - Introdução ao Scratch" e Site https://scratch.mit.edu/

Avaliação:

A avaliação será feita pelas dúvidas apresentadas pelos estudantes durante a revisão, além do desempenho durante as atividades práticas propostas. O professor irá percorrer o laboratório e observar o desempenho de cada estudante, solucionando dúvidas e auxiliando nas dificuldades apresentadas.

Referências:

AONO, Alexandre Hild; RODY, Hugo Vianna Silva; MUSA, Daniela Leal; PEREIRA, Vanessa Andrade; ALMEIDA, Jurandy. A Utilização do Scratch como Ferramenta no Ensino de Pensamento Computacional para Crianças. In: WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO (WEI), 25. , 2017, São Paulo. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2017 . ISSN 2595-6175. DOI: <https://doi.org/10.5753/wei.2017.3556>.

Davidson, Susan e Chris Murphy: Computational thinking for problem solving. University of Pennsylvania, 2022. <https://www.coursera.org/learn/computational-thinking-problem-solving/>, acesso em 2022-07-24.

Lab, Creative Computing: Creative computing curriculum. Desenvolvido por Laboratório de Computação Criativa da Escola de Graduação em Educação de Harvard e Traduzido por Rede Brasileira de Aprendizagem Criativa (aprendizagemcriativa.org) e Universidade Estadual de Feira de Santana (UEFS), Brasil, 2019. <https://lcl.media.mit.edu/resources/readings/creative-computing-guide.pt.pdf>, acesso em 2021-10-06.

2.3. AULA 03 - Procedimentos e Variáveis no Scratch

Informações do curso:

Curso:	Aprenda Pensamento Computacional e Scratch
Tema:	Scratch
Carga horária:	2 horas

Objetivos:

- Apresentar e Implementar procedimentos em Scratch e técnicas para construção de comandos
- Apresentar e Implementar tipos de dados no Scratch, criar e manipular dados

Conteúdos:

- Revisão do conteúdo da aula anterior
- Definição do projeto
- Procedimentos
- Variáveis
- Desenvolvimento do projeto

Procedimentos:

Conteúdo	Tempo previsto	Detalhamento	Material utilizado
Revisão do conteúdo da aula anterior	0:10	O professor fará uma breve exposição oral com uma revisão dos conceitos (Scratch, figuras e efeitos visuais e sonoros) passado na aula anterior com o objetivo de fixar o conteúdo e tirar alguma dúvida que tenha surgido.	
Definição do projeto	0:10	As duplas ou trios apresentarão as ideias dos projetos e o professor mediará um debate com os grupos para avaliar a viabilidade e escopo do que será desenvolvido por cada um. Aos grupos que não tiverem uma ideia, serão apresentadas três ideias para escolha e definição do escopo.	
Procedimentos	0:15	Exposição teórica, prática e apresentação de procedimentos, contextualizando com exemplos, e como criá-los no Scratch.	
Variáveis	0:25	Exposição teórica, prática e apresentação de variáveis e seus tipos, além dos identificadores, contextualizando com exemplos, e como criá-las no Scratch. É apresentado uma comparação entre um código Scratch e outro em Python, que fazem a mesma coisa, para que os estudantes vejam a dificuldade de linguagens não visuais, ao mesmo	Slide "Aula 3 - Procedimentos e Variáveis" e Site https://scratch.mit.edu/

		tempo que instiguem a querer aprendê-las. O professor orientará os estudantes a realizarem a atividade prática proposta (criar, usar e manipular dados em pelo menos 4 variáveis, além de fazer operações com elas).	
Desenvolvimento do projeto	1:00	O professor irá auxiliar os estudantes no desenvolvimento do projeto, além de dar suporte e dicas, instigando-os a pensarem nas soluções para os problemas que precisam resolver. Os estudantes serão lembrados dos pilares do Pensamento Computacional com o objetivo de fixar a metodologia de resolução de problemas e, depois aplicar à linguagem que estão utilizando, além de exercitar a colaboração na busca da solução com os demais colegas do grupo.	Site https://scratch.mit.edu/

Avaliação:

A avaliação será feita pelas dúvidas apresentadas pelos estudantes durante a revisão, além do desempenho durante a atividade prática proposta e desenvolvimento do projeto. O professor irá percorrer o laboratório e observar o desempenho de cada estudante, solucionando dúvidas e auxiliando nas dificuldades apresentadas.

Referências:

AONO, Alexandre Hild; RODY, Hugo Vianna Silva; MUSA, Daniela Leal; PEREIRA, Vanessa Andrade; ALMEIDA, Jurandy. A Utilização do Scratch como Ferramenta no Ensino de Pensamento Computacional para Crianças. In: WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO (WEI), 25. , 2017, São Paulo. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2017 . ISSN 2595-6175. DOI: <https://doi.org/10.5753/wei.2017.3556>.

Davidson, Susan e Chris Murphy: Computational thinking for problem solving. University of Pennsylvania, 2022. <https://www.coursera.org/learn/computational-thinking-problem-solving/>, acesso em 2022-07-24.

Lab, Creative Computing: Creative computing curriculum. Desenvolvido por Laboratório de Computação Criativa da Escola de Graduação em Educação de Harvard e Traduzido por Rede Brasileira de Aprendizagem Criativa (aprendizagemcriativa.org) e Universidade Estadual de Feira de Santana (UEFS), Brasil, 2019. <https://lcl.media.mit.edu/resources/readings/creative-computing-guide.pt.pdf>, acesso em 2021-10-06.

2.4. AULA 04 - Estruturas de Decisão e Repetição no Scratch

Informações do curso:

Curso:	Aprenda Pensamento Computacional e Scratch
Tema:	Scratch
Carga horária:	2 horas

Objetivos:

- Apresentar e Implementar técnicas para resolução de problemas, uso dos comandos de seleção e fluxo de controle
- Apresentar e Implementar estruturas de repetição para execução de comandos eficientes e variáveis de controle

Conteúdos:

- Revisão do conteúdo da aula anterior
- Operadores relacionais
- Operadores lógicos
- Estruturas condicionais
- Estruturas de repetição
- Desenvolvimento do projeto

Procedimentos:

Conteúdo	Tempo previsto	Detalhamento	Material utilizado
Revisão do conteúdo da aula anterior	0:10	O professor fará uma breve exposição oral com uma revisão dos conceitos (Procedimentos e Variáveis) passado na aula anterior com o objetivo de fixar o conteúdo e tirar alguma dúvida que tenha surgido.	
Operadores relacionais	0:10	Exposição teórica, prática e apresentação de operadores relacionais (Maior que, Maior ou igual, Menor que, Menor ou igual, Igualdade e Diferença), contextualizando com exemplos, e como usá-los no Scratch.	Slide "Aula 4 - Condicionais e Repetição" e Site https://scratch.mit.edu/
Operadores lógicos	0:10	Exposição teórica, prática e apresentação de operadores lógicos ("e" e "ou"), contextualizando com exemplos, e como usá-los no Scratch.	
Estruturas condicionais	0:15	Exposição teórica, prática e apresentação de estruturas condicionais (se então e se então - senão), contextualizando com exemplos, e como usá-los no Scratch. O professor orientará os estudantes a realizarem a atividade prática proposta (criar um programa no Scratch que use pelo menos duas diferentes estruturas condicionais).	

Estruturas de repetição	0:15	Exposição teórica, prática e apresentação de estruturas de repetição (sempre, repita x vezes e repita até que ...), contextualizando com exemplos, e como usá-los no Scratch. O professor orientará os estudantes a realizarem a atividade prática proposta (criar um programa no Scratch que use pelo menos duas diferentes estruturas de repetição).	
Desenvolvimento do projeto	1:00	O professor irá auxiliar os estudantes no desenvolvimento do projeto, além de dar suporte e dicas, instigando-os a pensarem nas soluções para os problemas que precisam resolver. Os estudantes serão lembrados dos pilares do Pensamento Computacional com o objetivo de fixar a metodologia de resolução de problemas e, depois aplicar à linguagem que estão utilizando, além de exercitar a colaboração na busca da solução com os demais colegas do grupo.	Site https://scratch.mit.edu/

Avaliação:

A avaliação será feita pelas dúvidas apresentadas pelos estudantes durante a revisão, além do desempenho durante a atividade prática proposta e desenvolvimento do projeto. O professor irá percorrer o laboratório e observar o desempenho de cada estudante, solucionando dúvidas e auxiliando nas dificuldades apresentadas.

Referências:

AONO, Alexandre Hild; RODY, Hugo Vianna Silva; MUSA, Daniela Leal; PEREIRA, Vanessa Andrade; ALMEIDA, Jurandy. A Utilização do Scratch como Ferramenta no Ensino de Pensamento Computacional para Crianças. In: WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO (WEI), 25. , 2017, São Paulo. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2017 . ISSN 2595-6175. DOI: <https://doi.org/10.5753/wei.2017.3556>.

Davidson, Susan e Chris Murphy: Computational thinking for problem solving. University of Pennsylvania, 2022. <https://www.coursera.org/learn/computational-thinking-problem-solving/>, acesso em 2022-07-24.

Lab, Creative Computing: Creative computing curriculum. Desenvolvido por Laboratório de Computação Criativa da Escola de Graduação em Educação de Harvard e Traduzido por Rede Brasileira de Aprendizagem Criativa (aprendizagemcriativa.org) e Universidade Estadual de Feira de Santana (UEFS), Brasil, 2019. <https://lcl.media.mit.edu/resources/readings/creative-computing-guide.pt.pdf>, acesso em 2021-10-06.

2.5. AULA 05 - Processamento de Strings e Finalização e Apresentação do Projeto

Informações do curso:

Curso:	Aprenda Pensamento Computacional e Scratch
Tema:	Scratch
Carga horária:	2 horas

Objetivos:

- Apresentar e Implementar técnicas para processamento de strings
- Finalizar o projeto e apresentar

Conteúdos:

- Revisão do conteúdo da aula anterior
- Processamento de strings
- Desenvolvimento do projeto
- Apresentação do projeto
- Avaliação final

Procedimentos:

Conteúdo	Tempo previsto	Detalhamento	Material utilizado
Revisão do conteúdo da aula anterior	0:10	O professor fará uma breve exposição oral com uma revisão dos conceitos (Operadores relacionais e lógicos e Estruturas condicionais e de repetição) passado na aula anterior com o objetivo de fixar o conteúdo e tirar alguma dúvida que tenha surgido.	
Processamento de strings	0:10	Exposição teórica, prática e apresentação de processamento de strings, contextualizando com exemplos, e como usá-los no Scratch. O professor orientará os estudantes a realizarem a atividade prática proposta (criar um programa no Scratch que use processamento de strings).	Slide "Aula 4 - Condicionais e Repetição" e Site https://scratch.mit.edu/
Desenvolvimento do projeto	1:00	O professor irá auxiliar os estudantes no desenvolvimento do projeto, além de dar suporte e dicas, instigando-os a pensarem nas soluções para os problemas que precisam resolver. Os estudantes serão lembrados dos pilares do Pensamento Computacional com o objetivo de fixar a metodologia de resolução de problemas e, depois aplicar à linguagem que estão utilizando, além de exercitar a	Site https://scratch.mit.edu/

		colaboração na busca da solução com os demais colegas do grupo. Como é a última aula, os estudantes farão ajustes necessários e corrigirão possíveis erros que ainda não conseguiram resolver para o pleno funcionamento do projeto.	
Apresentação do projeto	0:30	O professor irá chamar cada grupo para apresentar o projeto que fizeram para o curso. Os estudantes deverão apresentar o os integrantes do grupo, o nome e o que o programa faz, usar o programa e por fim abrir o código para mostrar o que cada parte do algoritmo faz).	Site https://scratch.mit.edu/
Avaliação final	0:10	Os estudantes irão preencher o formulário de avaliação do curso. Este questionário tem o objetivo avaliar a percepção dos estudantes sobre diversos itens do curso.	Questionário "Avaliação final do curso"

Avaliação:

A avaliação será feita pelas dúvidas apresentadas pelos estudantes durante a revisão, além do desempenho durante a atividade prática proposta e desenvolvimento do projeto. O professor irá percorrer o laboratório e observar o desempenho de cada estudante, solucionando dúvidas e auxiliando nas dificuldades apresentadas. A avaliação final do curso será realizada com a entrega e análise da apresentação dos projetos.

Referências:

AONO, Alexandre Hild; RODY, Hugo Vianna Silva; MUSA, Daniela Leal; PEREIRA, Vanessa Andrade; ALMEIDA, Jurandy. A Utilização do Scratch como Ferramenta no Ensino de Pensamento Computacional para Crianças. In: WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO (WEI), 25. , 2017, São Paulo. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2017 . ISSN 2595-6175. DOI: <https://doi.org/10.5753/wei.2017.3556>.

Davidson, Susan e Chris Murphy: Computational thinking for problem solving. University of Pennsylvania, 2022. <https://www.coursera.org/learn/computational-thinking-problem-solving/>, acesso em 2022-07-24.

Lab, Creative Computing: Creative computing curriculum. Desenvolvido por Laboratório de Computação Criativa da Escola de Graduação em Educação de Harvard e Traduzido por Rede Brasileira de Aprendizagem Criativa (aprendizagemcriativa.org) e Universidade Estadual de Feira de Santana (UEFS), Brasil, 2019. <https://lcl.media.mit.edu/resources/readings/creative-computing-guide.pt.pdf>, acesso em 2021-10-06.

Apêndice B

Formulário de Inscrição do Curso no Centro de Ensino Médio 01 do Paranoá

Inscrição para o Curso CODIFICO: Programação para Tod@s

A Universidade de Brasília está ofertando um curso que tem como objetivo o ensino do pensamento computacional (pensar como um cientista da computação) e da linguagem de programação visual Scratch para os estudantes do Centro de Ensino Médio 01 do Paranoá.

Ao final do curso o estudante será capaz de desenvolver jogos/programas usando o Scratch!

Se o estudante participar até o final do curso poderá receber um prêmio como forma de incentivo pela dedicação e conclusão do curso!

Inscrições até 21/10/2021

VAGAS LIMITADAS!

INÍCIO DO CURSO: A partir do dia 25/10/2021

ENCERRAMENTO DO CURSO: até o dia 17/12/2021

*Obrigatório

Você sabia?

Não precisa ter nenhum conhecimento prévio para esse curso. Qualquer pessoa pode participar! Chamem seus amigos!

1. Você sabia que poderá ter um prêmio para quem finalizar o curso? *

Marcar apenas uma oval.

- Sim
- Não
- Estou sabendo agora! :D

2. Nome completo *

3. E-mail *

4. Celular (Preferencialmente WhatsApp) *

5. Em qual ano do Ensino Médio você está? *

Marcar apenas uma oval.

- 1o ano
- 2o ano
- 3o ano

6. Qual sua turma? *

Marcar apenas uma oval.

- A
- B
- C
- D
- E
- F
- G
- H
- I
- J
- K
- L
- M

7. Qual turno você estuda? *

Marcar apenas uma oval.

- Matutino
- Vespertino

8. Disponibilidade de horários (marcar todos possíveis) *

Marque todas que se aplicam.

- Segunda e Quarta - 8h às 9h
- Segunda e Quarta - 9h às 10h
- Segunda e Quarta - 10h às 11h
- Segunda e Quarta - 11h às 12h
- Segunda e Quarta - 12h às 13h
- Terça e Quinta - 12h às 13h
- Quarta e Sexta - 12h às 13h

9. Você possui computador para utilizar durante o curso? (não é necessário ter um) *

Marcar apenas uma oval.

- Sim
- Não

10. Caso não tenha computador, possui celular para utilizar durante o curso? *

Marcar apenas uma oval.

- Sim
- Não
- Tenho computador

Apêndice C

Cartaz de Divulgação do Curso no
Centro de Ensino Médio 01 do
Paranoá



UnB | DEX

PROGRAMAÇÃO PARA TOD@S



Início: a partir do dia 25/10/2021

Fim: até o dia 15/12/2021

Não é necessária experiência prévia

Fica até o final que vai ter brinde :D

INSCRIÇÕES ATÉ 21/10/2021

VAGAS LIMITADAS!

BASTA PREENCHER O FORMULÁRIO

Apêndice D

Formulário de Inscrição do Curso no Centro de Ensino Médio da Asa Norte

Inscrição para o Curso CODIFICO: Pensamento Computacional e Scratch - CEAN

A Universidade de Brasília está ofertando um curso que tem como objetivo o ensino do pensamento computacional (pensar como um cientista da computação) e da linguagem de programação visual Scratch para os estudantes do Centro de Ensino Médio da Asa Norte - CEAN.

Ao final do curso o estudante será capaz de entender o Pensamento Computacional e desenvolver jogos/programas usando o Scratch!

Se o estudante participar até o final do curso poderá receber um prêmio como forma de incentivo pela dedicação e conclusão do curso!

Inscrições até 01/05/2022

VAGAS LIMITADAS!

INÍCIO DO CURSO: 02/05/2022

ENCERRAMENTO DO CURSO: 30/05/2022

Toda segunda-feira das 14:00 às 16:00

*Obrigatório

Você sabia?

Não precisa ter nenhum conhecimento prévio para esse curso. Qualquer pessoa pode participar! Chamem seus amigos!

1. Você sabia que poderá ter um prêmio para quem finalizar o curso? *

Marcar apenas uma oval.

- Sim
- Não
- Estou sabendo agora! :D

2. Nome completo *

3. E-mail (que veja os e-mails) *

4. Celular (Preferencialmente WhatsApp) *

5. Em qual ano do Ensino Médio você está? *

Marcar apenas uma oval.

1o ano

2o ano

3o ano

6. Qual sua turma? *

Marcar apenas uma oval.

A

B

C

D

E

F

G

H

I

J

K

L

M

7. Qual turno você estuda? *

Marcar apenas uma oval.

Matutino

Vespertino

8. Possui disponibilidade para participar das aulas nos dias e horários mencionados no cabeçalho deste formulário? *

Marcar apenas uma oval.

Sim

Não

Este conteúdo não foi criado nem aprovado pelo Google.

Google Formulários

Apêndice E

Cartaz de divulgação do Curso no
Centro de Ensino Médio da Asa
Norte



APRENDA PENSAMENTO COMPUTACIONAL E **SCRATCH**

A Universidade de Brasília está ofertando um curso que tem como objetivo o ensino do pensamento computacional e da linguagem de programação visual Scratch

De 02/05/2022 a 30/05/2022

Toda segunda-feira das 14:00 às 16:00

Não é necessário nenhuma experiência prévia

Aulas presenciais no Lab. de Informática do CEAN



INSCRIÇÕES ATÉ 01/05/2022
BASTA PREENCHER O FORMULÁRIO
VAGAS LIMITADAS

Apêndice F

Cartaz de divulgação do Curso no
Centro de Ensino Médio Paulo Freire



UnB | DEX

APRENDA PENSAMENTO COMPUTACIONAL E SCRATCH

A Universidade de Brasília está ofertando um curso que tem como objetivo o ensino do pensamento computacional e da linguagem de programação visual Scratch



Datas e horários a definir
Não é necessário nenhuma experiência prévia
Aulas presenciais no Centro de Ensino Médio Paulo Freire

INSCRIÇÕES ATÉ 28/02/2022

ENVIAR E-MAIL PARA:

SIMONE.NASCIMENTO@EDU.SE.DF.GOV.BR

Apêndice G

Formulário de Caracterização do Perfil dos Estudantes

Perfil do estudante

A Universidade de Brasília está ofertando um curso que tem como objetivo o ensino do pensamento computacional e da linguagem de programação visual Scratch para os estudantes do Ensino Médio das escolas públicas do Distrito Federal.

Ao final do curso o estudante será capaz de desenvolver jogos/programas usando o Scratch!

Se o estudante participar até o final do curso poderá receber um prêmio como forma de incentivo pela dedicação e conclusão!

Estas perguntas tem o objetivo de caracterizar o perfil dos estudantes que estão participando do curso para melhor aplicação dos métodos aulas.

Ao responder as perguntas, estou de acordo com o uso dos meus dados sobre o ensino do Pensamento Computacional e do Scratch nas escolas públicas de Ensino Médio do Distrito Federal, exclusivamente para fins de pesquisa.

*Obrigatório

1. Nome completo *

2. E-mail *

3. Celular (Preferencialmente WhatsApp) *

4. Sexo? *

Marcar apenas uma oval.

- Feminino
- Masculino
- Prefiro não dizer

5. Qual Região Administrativa você reside? *

Marcar apenas uma oval.

- I Plano Piloto
- II Gama
- III Taguatinga
- IV Brazlândia
- V Sobradinho
- VI Planaltina
- VII Paranoá
- VIII Núcleo Bandeirante
- IX Ceilândia
- X Guará
- XI Cruzeiro
- XII Samambaia
- XIII Santa Maria
- XIV São Sebastião
- XV Recanto das Emas
- XVI Lago Sul
- XVII Riacho Fundo 1
- XVIII Lago Norte
- XIX Candangolândia
- XX Águas Claras
- XXI Riacho Fundo 2
- XXII Sudoeste/Octogonal
- XXIII Varjão
- XXIV Park Way
- XXV Estrutural/Scia
- XXVI Sobradinho II
- XXVII Jardim Botânico
- XXVIII Itapoã
- XXIX SIA
- XXX Vicente Pires
- XXXI Fercal
- XXXII Sol Nascente/Pôr do Sol
- XXXIII Arniqueira
- Entorno do DF

6. Qual seu meio de locomoção para a escola? (Selecione quantas opções quiser) *

Marque todas que se aplicam.

- A pé
- Bicicleta
- Carro particular
- Moto
- Ônibus intermunicipal
- Ônibus urbano

7. Em qual ano do Ensino Médio você está? *

Marcar apenas uma oval.

- 1o ano
 2o ano
 3o ano

8. Qual sua turma? *

Marcar apenas uma oval.

- A
 B
 C
 D
 E
 F
 G
 H
 I
 J
 K
 L
 M

9. Qual turno você estuda? *

Marcar apenas uma oval.

- Matutino
 Verspertino

10. Qual escola você estuda? *

Marcar apenas uma oval.

- Centro de Ensino Médio da Asa Norte - CEAN
 Centro de Ensino Médio Paulo Freire - CEMPF

11. Você possui computador em casa? *

Marcar apenas uma oval.

- Sim
 Não

12. Se possui computador em casa, é de uso exclusivo ou compartilhado com os demais membros da casa? *

Marcar apenas uma oval.

- Exclusivo para meu uso
 Uso compartilhado
 Não possuo computador em casa

13. Você possui celular? *

Marcar apenas uma oval.

- Sim
 Não

14. Você possui internet móvel no celular? *

Marcar apenas uma oval.

- Sim
 Não
 Não possuo celular.

15. Você possui tablet? *

Marcar apenas uma oval.

- Sim
 Não

16. Você possui acesso à internet em casa? *

Marcar apenas uma oval.

- Sim
 Não

17. Qual seu nível de conhecimento em informática? *

Marcar apenas uma oval.

	0	1	2	3	4	5	6	7	8	9	10	
Nenhum	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Alto

18. Antes de fazer esse curso você possuía algum conhecimento em programação? *

Marcar apenas uma oval.

- Sim
 Não

19. Considera bom em raciocínio lógico? *

Marcar apenas uma oval.

Sim

Não

20. Escolha as opções que melhor definem seu uso/nível de conhecimento em informática. (Selecione quantas opções quiser) *

Marque todas que se aplicam.

Já fiz algum curso de informática básica

Sei navegar na internet (sites, blogs, notícias, Google...)

Uso para jogar

Sei usar ferramentas como editores de texto (Word), planilhas (Excel) e apresentações (PowerPoint)

Uso para acessar as aulas no Google Sala de Aula

Uso redes sociais (Facebook, Instagram, Twitter, Whatsapp, Telegram, Youtube, TikTok, Kwai, Pinterest...)

Não sei usar nenhuma ferramenta ou não uso

Sei/aprendi a programar

Fiz/faço curso técnico em informática

Outro: _____

21. Qual disciplina você MAIS gosta/tem facilidade de aprender na escola? *

Marcar apenas uma oval.

Artes

Biologia

Educação Física

Espanhol

Filosofia

Física

Geografia

História

Inglês

Língua Portuguesa

Matemática

Química

Sociologia

22. Qual disciplina você MENOS gosta/tem facilidade de aprender na escola? *

Marcar apenas uma oval.

- Artes
- Biologia
- Educação Física
- Espanhol
- Filosofia
- Física
- Geografia
- História
- Inglês
- Língua Portuguesa
- Matemática
- Química
- Sociologia

23. Por quê você quis participar desse curso? *

24. O que você espera aprender nesse curso? *

25. Qual curso superior pretende fazer ao terminar o ensino médio? Qual profissão deseja seguir? *

26. Autorização para o uso dos dados *

Marcar apenas uma oval.

- Permito que meus dados sejam utilizados, exclusivamente, para fins da pesquisa sobre o ensino do Pensamento Computacional e do Scratch nas escolas públicas de Ensino Médio do Distrito Federal.

Este conteúdo não foi criado nem aprovado pelo Google.

Google Formulários

Apêndice H

Formulário de Avaliação Final do Curso

Avaliação Final - PC & Scratch

A Universidade de Brasília está ofertando um curso que tem como objetivo o ensino do pensamento computacional e da linguagem de programação visual Scratch para os estudantes do Ensino Médio das escolas públicas do Distrito Federal.

Ao final do curso o estudante será capaz de desenvolver jogos/programas usando o Scratch!

Se o estudante participar até o final do curso poderá receber um prêmio como forma de incentivo pela dedicação e conclusão!

Estas perguntas tem o objetivo de avaliar o curso do ponto de vista dos estudantes participantes.

Ao responder as perguntas, estou de acordo com o uso dos meus dados sobre o ensino do Pensamento Computacional e do Scratch nas escolas públicas de Ensino Médio do Distrito Federal, exclusivamente para fins de pesquisa.

***Obrigatório**

1. Nome completo *

2. E-mail *

3. Celular (Preferencialmente WhatsApp) *

4. CPF

Apenas deixe o CPF abaixo se quiser receber o certificado do curso.

5. Sexo? *

Marcar apenas uma oval.

- Feminino
 Masculino
 Prefiro não dizer

6. Qual escola você estuda? *

Marcar apenas uma oval.

- Centro de Ensino Médio da Asa Norte - CEAN
 Centro de Ensino Médio Paulo Freire - CEMPF

7. Agora, após o curso, qual seu nível de conhecimento em informática? *

Marcar apenas uma oval.

	0	1	2	3	4	5	6	7	8	9	10	
Nenhum	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Alto

8. Você gostou do curso? *

Marcar apenas uma oval.

Sim

Não

9. Você considera que a aplicação dos conceitos/temas desse curso em sua vida acadêmica/profissional e pessoal é: *

1 - Não é útil; 2 - Pouco útil; 3 - Razoavelmente útil; 4 - Útil; 5 - Muito útil

Marcar apenas uma oval.

1 2 3 4 5

Não é útil Muito útil

10. O que você mais gostou no curso? *

11. O que menos gostou do curso? *

12. Explique o que você entende por pensamento computacional? *

13. Como você avalia o formato e condução (metodologia) das aulas do curso? *

1 - Muito ruim; 2 - Ruim; 3 - Normal; 4 - Bom; 5 - Muito bom

Marcar apenas uma oval.

1 2 3 4 5

Muito ruim Muito bom

14. Como você avalia a forma do professor comunicar e aplicar o conteúdo (didática) nas aulas do curso? *

1 - Muito ruim; 2 - Ruim; 3 - Normal; 4 - Bom; 5 - Muito bom

Marcar apenas uma oval.

1 2 3 4 5

Muito ruim Muito bom

15. Como você avalia o conhecimento que o professor tem sobre o assunto do curso? *

1 - Conhece nada; 2 - Conhece pouco; 3 - Conhece o suficiente; 4 - Conhece muito; 5 - Conhece tudo

Marcar apenas uma oval.

1 2 3 4 5

Conhece nada Conhece tudo

16. Como você avalia a sua aprendizagem ao longo do curso? *

1 - Aprendi nada; 2 - Aprendi pouco; 3 - Aprendi o suficiente; 4 - Aprendi muito; 5 - Aprendi tudo

Marcar apenas uma oval.

1 2 3 4 5

Aprendi nada Aprendi tudo

17. Quais críticas ou sugestões você pode fazer para contribuir para melhorias do curso? *

18. Quantas aulas você participou? *

Marcar apenas uma oval.

0 1 2 3 4 5

nenhuma todas

19. Autorização para o uso dos dados *

Marcar apenas uma oval.

Permito que meus dados sejam utilizados, exclusivamente, para fins da pesquisa sobre o ensino do Pensamento Computacional e do Scratch nas escolas públicas de Ensino Médio do Distrito Federal.

Este conteúdo não foi criado nem aprovado pelo Google.

Google Formulários