



Universidade de Brasília

Instituto de Ciências Exatas
Departamento de Ciência da Computação

Preparação de estudantes de ensino médio para a Olimpíada Brasileira de Informática: um Relato de Experiência

Brunno Costa Martins

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

Orientador

Prof. Dr. Edson Alves da Costa Júnior

Brasília
2025



Preparação de estudantes de ensino médio para a Olimpíada Brasileira de Informática: um Relato de Experiência

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

Brasília, 24 de julho de 2025

Dedicatória

Eu dedico este trabalho à minha esposa e à minha mãe, pessoas que acreditaram em mim quando eu mesmo duvidei. Como diz a canção “O Anjo Mais Velho”, de O Teatro Mágico, “enquanto houver você(s) do outro lado aqui do outro eu consigo me orientar”.

Agradecimentos

Agradeço a Deus pelo dom da vida e pela oportunidade de chegar a esse momento. “*Todas as coisas foram feitas por Ele, e sem Ele nada do que foi feito se fez*” (Jo 1,3). Toda honra e glória sejam dadas a Ele.

Este trabalho foi construído a várias mãos e aconteceu somente porque encontrou as pessoas certas, no momento certo e com o propósito certo em seus corações. Por esse motivo eu agradeço imensamente ao professor Herberth Guimarães, que me recebeu de braços abertos no CED 04 em 2023 e agradeço à professora Raquel que nos acolheu nas aulas de robótica e que foi a responsável por me conectar com a professora Regiane. Também à professora Regiane Quezia, que não só me recebeu e me acolheu em sala de aula, mas que foi a grande motivadora do projeto de preparação para a OBI, sem ela esse projeto não seria viável. Quero deixar, também, meu agradecimento ao professor Edson Alves, que me incluiu no projeto e me apoiou com instruções e direcionamentos desde o início, credito a ele os bons resultados que tivemos e espero conseguir ser referência no ensino de computação, assim como ele é para mim.

Por fim, quero deixar um agradecimento à minha família, em especial à minha esposa Raquel Reis, meu eterno amor e parceira de todas as horas. À minha mãe Sandra Costa, sem a qual eu não estaria aqui, e aos meus sogros Reginaldo e Elza, que sempre torcem e oram por mim.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES), por meio do Acesso ao Portal de Periódicos.

Resumo

Este trabalho apresenta um estudo de caso que examina o design, implementação e impacto de um curso preparatório para a Olimpíada Brasileira de Informática (OBI) direcionado a estudantes do ensino médio de uma escola pública do Distrito Federal, Brasil. O projeto empregou a metodologia de Aprendizagem Baseada em Problemas (PBL) combinada com plataformas de programação competitiva para preparar os estudantes para a competição, ao mesmo tempo em que promoveu o desenvolvimento de habilidades de Pensamento Computacional. A pesquisa utilizou uma abordagem de métodos mistos, incorporando análises qualitativas e quantitativas. O desempenho dos estudantes nas fases da OBI foi sistematicamente analisado, e as percepções dos participantes foram coletadas por meio de entrevistas semiestruturadas. Os resultados demonstram a eficácia desta intervenção educacional. Notavelmente, o programa alcançou a primeira classificação de estudantes participantes para a segunda fase da competição, um marco significativo para a instituição. Mais importante ainda, a iniciativa gerou impacto positivo substancial nas escolhas de trajetórias acadêmicas e profissionais dos estudantes. A maioria dos participantes relatou aumento do interesse em áreas relacionadas a tecnologias, com essa influência diretamente refletida em suas escolhas subsequentes de cursar ensino superior e carreiras em áreas relacionadas à tecnologia. O estudo conclui que uma abordagem prática para a educação em computação se mostra altamente eficaz na promoção da aprendizagem ativa e do engajamento estudantil. Esta experiência serviu como uma valiosa porta de entrada para oportunidades em tecnologia, reforçando o papel crucial da educação pública no cultivo de novos talentos e vocações no setor tecnológico.

Palavras-chave: Ensino de Computação, Olimpíada Brasileira de Informática, Aprendizagem Baseada em Problemas, Pensamento Computacional, Escola Pública

Abstract

This paper presents a case study examining the design, implementation, and impact of a preparatory course for the Brazilian Informatics Olympiad (OBI) targeting high school students from a public school in the Federal District, Brazil. The project employed Problem-Based Learning (PBL) methodology, combined with competitive programming platforms, to prepare students for the competition while fostering the development of Computational Thinking skills. The research used a mixed-methods approach, incorporating both qualitative and quantitative analyses. Student performance across OBI phases was systematically analyzed, and participant perceptions were gathered through semi-structured interviews. The findings demonstrate the effectiveness of this educational intervention. Notably, the program achieved the first-time qualification of participating students to the second phase of the competition, a significant milestone for the institution. More importantly, the initiative generated substantial positive impact on students' academic and career trajectory choices. The majority of participants reported increased interest in technology fields, with this influence reflected in their subsequent decisions to pursue higher education and careers in technology-related areas. The study concludes that a practical, hands-on approach to computing education proves highly effective in promoting active learning and student engagement. This experience served as a valuable gateway to opportunities in technology, reinforcing the crucial role of public education in cultivating new talent and vocations in the technology sector.

Keywords: Computer Science Education, Brazilian Olympiad in Informatics, Problem Based Learning, Computational Thinking, Public School

Lista de Figuras

- 3.1 Lista de desafios para iniciantes, em ordem de dificuldade dos problemas. . 19

Lista de Tabelas

4.1	Números de desafios realizados pelos participantes em cada lista proposta .	32
4.2	Desempenho dos participantes na Primeira Fase da OBI	34
4.3	Desempenho dos participantes na Segunda Fase da OBI	38
A.1	Níveis cognitivos e tópicos a serem trabalhados	77
A.2	Atividades a serem desenvolvidas	78

Sumário

1	Introdução	1
1.1	Justificativa	3
1.2	Objetivos	3
1.3	Estrutura do Trabalho	4
2	Fundamentação Teórica	6
2.1	Pensamento computacional	6
2.2	Olimpíadas de conhecimento	7
2.3	Olimpíada Brasileira de Informática	9
2.4	Aprendizagem Baseada em Problemas	11
2.5	Trabalhos Correlatos	12
3	Materiais e Métodos	15
3.1	Pesquisa Bibliográfica	15
3.2	Treinamento	16
3.3	Eventos	22
3.3.1	Mini-OBI 1	23
3.3.2	Mini-OBI 2	25
3.3.3	OBI Fase 1	27
3.3.4	OBI Fase 2 e CF	27
3.4	Entrevista de avaliação qualitativa	28
4	Resultados	30
4.1	Resultados do Treinamento	30
4.1.1	Resultados da Fase 1 da OBI	32
4.1.2	Resultados da Fase 2 da OBI	34
4.1.3	Resultados da CF - OBI	37
4.2	Análise Qualitativa do Treinamento	39
4.2.1	Limitações do Estudo	42

5	Considerações Finais	44
	Apêndice	49
A	Planos de Ensino	50
A.1	Plano 1 - Impressão no terminal	50
A.1.1	Princípios Educacionais	50
A.1.2	Objetivos Educacionais	50
A.1.3	Conceitos a serem desenvolvidos	51
A.1.4	Público-Alvo	52
A.1.5	Tema ou situação de contexto (organizadores prévios)	52
A.1.6	Níveis Cognitivos	52
A.1.7	Descrição da Atividade	52
A.1.8	Recursos Materiais e Tecnológicos	52
A.1.9	Avaliação	53
A.2	Plano 2 - Leitura de dados e operadores aritméticos	56
A.2.1	Princípios Educacionais	56
A.2.2	Objetivos Educacionais	56
A.2.3	Conceitos a serem desenvolvidos	57
A.2.4	Público-Alvo	57
A.2.5	Tema ou situação de contexto (organizadores prévios)	57
A.2.6	Níveis Cognitivos	58
A.2.7	Descrição da Atividade	58
A.2.8	Recursos Materiais e tecnológicos	58
A.2.9	Avaliação	59
A.3	Plano 3 - Controle de fluxo/condicionais	61
A.3.1	Princípios Educacionais	61
A.3.2	Objetivos Educacionais	61
A.3.3	Conceitos a serem desenvolvidos	62
A.3.4	Público-Alvo	62
A.3.5	Tema ou situação de contexto (organizadores prévios)	62
A.3.6	Níveis Cognitivos	62
A.3.7	Descrição da Atividade	63
A.3.8	Recursos Materiais e Tecnológicos	63
A.3.9	Avaliação	63
A.4	Plano 4 - Laços de repetição - while	66
A.4.1	Princípios Educacionais	66
A.4.2	Objetivos Educacionais	66

A.4.3	Conceitos a serem desenvolvidos	67
A.4.4	Público-Alvo	67
A.4.5	Tema ou situação de contexto (organizadores prévios)	67
A.4.6	Níveis Cognitivos	67
A.4.7	Descrição da Atividade	67
A.4.8	Recursos Materiais e Tecnológicos	68
A.4.9	Avaliação	69
A.5	Plano 5 - Laços de repetição - for	71
A.5.1	Princípios Educacionais	71
A.5.2	Objetivos Educacionais	71
A.5.3	Conceitos a serem desenvolvidos	72
A.5.4	Público-Alvo	72
A.5.5	Tema ou situação de contexto (organizadores prévios)	72
A.5.6	Níveis Cognitivos	72
A.5.7	Descrição da Atividade	72
A.5.8	Recursos Materiais e Tecnológicos	73
A.5.9	Avaliação	74
A.6	Plano 6 - Mini OBI	76
A.6.1	Princípios Educacionais	76
A.6.2	Objetivos Educacionais	76
A.6.3	Conceitos a serem desenvolvidos	76
A.6.4	Público-Alvo	77
A.6.5	Tema ou situação de contexto (organizadores prévios)	77
A.6.6	Níveis Cognitivos	77
A.6.7	Descrição da Atividade	78
A.6.8	Recursos Materiais e Tecnológicos	78
A.6.9	Avaliação	78
A.7	Plano 7 - Laços de repetição - <i>Arrays</i>	80
A.7.1	Princípios Educacionais	80
A.7.2	Objetivos Educacionais	80
A.7.3	Conceitos a serem desenvolvidos	81
A.7.4	Público-Alvo	81
A.7.5	Tema ou situação de contexto (organizadores prévios)	81
A.7.6	Níveis Cognitivos	81
A.7.7	Descrição da Atividade	81
A.7.8	Recursos Materiais e Tecnológicos	81
A.7.9	Avaliação	83

Capítulo 1

Introdução

O atual cenário educacional traz consigo desafios significativos, que são potencializados pelo rápido avanço tecnológico vivido nas últimas décadas. A inclusão de habilidades características do pensamento computacional na Base Nacional Comum Curricular – BNCC¹ evidencia uma realidade em que a tecnologia se torna cada vez mais presente na formação dos estudantes.

Segundo Wing[1], o pensamento computacional envolve a resolução de problemas, o *design* de sistemas e a compreensão do comportamento humano, baseando-se em conceitos fundamentais da ciência da computação. Dessa forma, a aquisição dessas habilidades pode transformar o aprendizado em tecnologia em uma ferramenta de transformação social por meio do ensino técnico, tanto pela criação de oportunidades profissionais quanto pelo desenvolvimento de competências necessárias para utilizar tecnologias na solução de desafios contemporâneos.

Nesse contexto, olimpíadas de conhecimento podem desempenhar um papel fundamental no incentivo ao estudo dos conteúdos presentes nos desafios propostos, auxiliando no desenvolvimento de habilidades analíticas associadas às competições. Campagnolo [2] analisa o caráter incentivador de olimpíadas de conhecimento, mais especificamente a Olimpíada Brasileira de Astronomia e Astronáutica (OBA)², pela perspectiva dos participantes da competição, e afirma que há influência positiva da olimpíada no fomento à busca de conhecimento científico. Com o objetivo de despertar o interesse dos estudantes pela área, a Olimpíada Brasileira de Informática – OBI foi escolhida como foco para o desenvolvimento de um treinamento preparatório destinado a estudantes de uma escola pública de Ensino Médio.

A formação em computação tem se mostrado cada vez mais importante no ingresso ao mercado de trabalho. De acordo com o Relatório Perspectivas do Mercado de Trabalho

¹<https://basenacionalcomum.mec.gov.br/abase/>

²<http://www.oba.org.br/site/>

do Macrossetor de TIC [3], há um descasamento de 30,2% entre a demanda e a oferta de profissionais de tecnologia, sendo que entre 2019 e 2024 houve uma demanda por esses profissionais de cerca de 665 mil vagas, frente a aproximadamente 464 mil vagas ofertadas entre 2018 e 2023. Também é possível perceber em publicações como a do Blog “Google Brasil”, onde são mencionados 90 casos de IA na América Latina que estão moldando o futuro da Inovação [4], que apesar dos dados apontarem que ainda há bastante demanda por profissionais de TI, percebe-se que o aumento do uso de ferramentas de Inteligência Artificial tem sido intensificado, elevando o nível de exigência para profissionais em estágio inicial ou em transição de carreira. Esse ponto é explorado na revista Harvard Business Review[5], que cita que, para além das necessidades de qualificação da força de trabalho, o impacto da inteligência artificial pode demandar até mesmo requalificação da mão de obra de diversas áreas.

Diante desse contexto de crescente exigência técnica, a incorporação do pensamento computacional à base curricular desde os primeiros anos da educação básica apresenta-se como uma abordagem estratégica. Tal inserção possibilita aos estudantes o desenvolvimento gradual desses conceitos em conjunto com outras disciplinas que compõem o currículo de formação escolar, preparando-os para o futuro mercado de trabalho e para a participação em uma sociedade que tende a ser cada vez mais digital.

Contudo, implementar o ensino de computação desde a educação básica é um desafio complexo e que possui obstáculos significativos no contexto de escolas públicas do Distrito Federal. Cabe ressaltar que os desafios estão relacionados à necessidade de contratação de professores especializados em computação, demanda que não é completamente atendida na rede pública local. Além disso, nem todas as escolas possuem uma infraestrutura suficiente para que as atividades relacionadas ao ensino de computação possam ser devidamente realizadas.

Os dados disponibilizados através dos anuários estatísticos pela Universidade de Brasília³ mostram que, entre 2017 e 2023, apenas 118 alunos se formaram no curso de Licenciatura em Computação. Considerando que, segundo a agência Brasília⁴, há 708 escolas públicas no DF, esse número evidencia uma defasagem crítica na formação de professores especializados. Ademais, o último concurso público realizado pela secretaria de Educação⁵ contou com somente 5 vagas imediatas e 20 vagas para o cadastro reserva para professores de informática, revelando a limitada capacidade de expansão da área nas escolas públicas.

³https://dpo.unb.br/index.php?option=com_content&view=article&id=47&Itemid=872

⁴<http://bit.ly/3TYdpdv>

⁵https://www.quadrix.org.br/web/visualizar.html?file=https://www.quadrix.org.br/Archives/General/26736/26737/26739/7F5EA8209E7F/10_SEEDF_concurso_publico_2022_edital_31_abertura_atualizado.pdf

1.1 Justificativa

A crescente presença da tecnologia em diversos segmentos da sociedade exige que, cada vez mais, a formação básica dos alunos promova competências e habilidades como o raciocínio lógico e o pensamento computacional. Apesar disso, no cenário atual da educação brasileira, os espaços formais que oferecem essas oportunidades aos estudantes, ainda são escassos.

Em alinhamento ao Complemento À Base Nacional Comum Curricular - Computação na Educação Básica⁶, que estabelece o pensamento computacional como um dos eixos norteadores de habilidades e competências no ensino básico, o presente trabalho tem como proposta oferecer o ensino complementar em computação, promovendo a interdisciplinaridade e criando ambientes para o desenvolvimento de habilidades interpessoais entre os estudantes.

Olimpíadas de conhecimento como a OBI, representam uma oportunidade de fortalecimento do vínculo entre teoria e prática no ato do ensino, e proporcionam aos estudantes desafios intelectuais e motivação. Criar ambientes estimulantes aos estudantes pode gerar engajamento, motivação e criar possibilidades de desenvolvimento de habilidades importantes para o futuro ingresso no mercado de trabalho.

Mais do que uma mera competição de programação, a OBI se configura como uma ferramenta poderosa para o desenvolvimento do pensamento computacional, conforme definido por WING (2006)[1]. A resolução dos desafios presentes nas provas da OBI requer que os estudantes não apenas codifiquem, mas que decomponham problemas complexos e partes menores, reconheçam padrões, abstraia detalhes irrelevantes e criem algoritmos eficientes. Neste sentido, o treinamento para a OBI atua como um exercício prático das competências centrais do pensamento computacional, que são essenciais para a formação dos estudantes.

Este trabalho busca preencher lacunas na formação oferecida pela escola pública e contribuir para a literatura educacional sobre práticas pedagógicas e sobre o ensino de programação na rede pública. Além disso, busca-se apresentar uma proposta replicável e passível de adaptação para diferentes cenários da educação pública.

1.2 Objetivos

O objetivo geral do presente trabalho é promover o desenvolvimento do pensamento computacional e preparar estudantes do Ensino Médio para a Olimpíada Brasileira de Informática (OBI) por meio de um curso de programação baseado em resolução de problemas,

⁶https://basenacionalcomum.mec.gov.br/images/historico/anexo_parecer_cneceb_n_2_2022_bncc_computacao.pdf

aplicando a metodologia de Aprendizado Baseado em Problemas – PBL, com o auxílio de uma abordagem expositiva.

Objetivos Específicos

Para atingir o objetivo geral, foram definidos os seguintes objetivos específicos:

- Propor uma sequência didática, como material didático de apoio ao docente, para a preparação de estudantes para a Olimpíada Brasileira de Informática;
- Avaliar o impacto do curso preparatório na proficiência dos estudantes em lógica de programação e resolução de problemas, com base na análise comparativa dos desempenhos nas atividades propostas;
- Investigar a percepção dos estudantes sobre a estrutura pedagógica e os recursos didáticos do curso, considerando sua aplicabilidade na preparação para a Olimpíada Brasileira de Informática.

1.3 Estrutura do Trabalho

O presente trabalho possui 5 capítulos, os quais abordam aspectos importantes sobre a idealização, execução e análise de resultados obtidos ao fim do projeto aqui apresentado. Inicialmente, o presente capítulo visa explicitar a motivação do trabalho, apresentando os objetivos gerais e específicos pretendidos pelo autor do trabalho. Em seguida, o Capítulo 2 apresenta os conceitos que fundamentaram o trabalho aqui proposto, abordando tópicos como o pensamento computacional e sua relação com a matriz de competências e habilidades apresentada no complemento à Base Nacional Comum Curricular (BNCC) de computação, assim como a abordagem de PBL, metodologia escolhida para a dinâmica de ensino na sala de aula; e termina trazendo trabalhos correlatos, apontando diferenças e semelhanças com o que aqui foi produzido.

O Capítulo 3 tem como objetivo detalhar as etapas de planejamento e execução do presente projeto, explicitando desde os termos usados na pesquisa bibliográfica de trabalhos correlatos até as estratégias escolhidas para a prática dos conteúdos por parte dos alunos, optando por separar listas de exercícios com base na dificuldade dos problemas escolhidos. Buscou-se fundamentar cada passo escolhido, desde a escolha da linguagem de programação adotada para o ensino de lógica até a previsão das datas de realização das aulas e seus conteúdos, de modo a entender de forma sistemática como o projeto foi estruturado e como o mesmo pode ser replicado.

Os resultados obtidos ao final da fase de execução do presente trabalho são trazidos no Capítulo 4, onde são detalhados os desempenhos dos participantes no treinamento e nas provas da Olimpíada Brasileira de Informática (OBI) e também é feita uma análise qualitativa acerca de aspectos fundamentais sobre o treinamento disponibilizado na escola, por meio de uma entrevista semi-estruturada aplicada com participantes ao final do curso.

Por fim, o Capítulo 5 traz um contraste entre objetivos propostos e os resultados alcançados, refletindo sobre os pontos negativos e positivos do treinamento, e deixando, ao final, uma sugestão para um possível seguimento do trabalho.

Capítulo 2

Fundamentação Teórica

Este capítulo tem como objetivo fundamentar e contextualizar a proposta do presente trabalho, abordando conceitos importantes para o desenvolvimento do mesmo, bem como suas referências. Inicialmente, apresenta-se o conceito de pensamento computacional, tema de importante relevância para o contexto de ensino de programação em escolas públicas, estando diretamente ligado às habilidades e competências a serem desenvolvidas por estudantes do Ensino Básico, como sugere o complemento à Base Nacional Comum Curricular (BNCC¹) de computação. Em seguida, discute-se o papel das olimpíadas de conhecimento no estímulo ao aprendizado, destacando as oportunidades trazidas pelas competições, citando exemplos notáveis como a OBMEP e OBA, e mencionando a janela de oportunidades disponível, como o uso de medalhas para ingresso na Educação Superior. A terceira seção traz informações sobre a Olimpíada Brasileira de Informática (OBI) – tema motivador do presente trabalho –, explicando sua estrutura e regras, e discutindo sobre o conteúdo programático da competição. Na sequência, apresenta-se o método de Aprendizagem Baseada em Problemas (PBL) com o foco no ensino de programação, discutindo sobre sua contribuição para o desenvolvimento de habilidades analíticas, técnicas e interpessoais, que podem contribuir para um bom desempenho em competições do porte da OBI. Por fim, o capítulo se encerra com uma revisão de trabalhos correlatos que fundamentam e dialogam com a proposta desse trabalho.

2.1 Pensamento computacional

No ano de 2006, Wing [1] popularizou o termo “pensamento computacional” ao publicar um artigo que reflete sobre a necessidade de se aprender um conjunto de habilidades que estruturam o pensamento lógico e usam soluções de problemas que conhecemos para resolver novos problemas. A ideia do pensamento computacional é desenvolver habilidades

¹<http://bit.ly/44wqUpu>

como reconhecimento de padrões, abstração e decomposição de problemas para aplicá-las na resolução de desafios complexos, que podem surgir de diversas áreas de conhecimento, não estando necessariamente ligadas à computação.

O complemento à Base Nacional Comum Curricular (BNCC) que trata da Computação na Educação Básica [6] aborda o pensamento computacional como uma competência transversal, que permeia diversas áreas do conhecimento como matemática, ciências da natureza e até mesmo linguagens, bem como diversas fases diferentes da vida dos alunos, uma vez que é citado desde os primeiros até o último ano do Ensino Básico.

Podemos perceber a influência direta do pensamento computacional em várias áreas do conhecimento. Na matemática, por exemplo, há uma clara influência exercida no desenvolvimento do pensamento algorítmico, o que pode auxiliar os alunos a conseguirem organizar uma sequência lógica de passos a serem seguidos para atacar problemas que demandem um raciocínio mais elaborado, desenvolvendo nos estudantes a capacidade de resolver problemas complexos. Em outras disciplinas, como ciências, o mesmo pensamento lógico e sequencial pode ajudar os alunos na estruturação e organização de experimentos químicos. Indiretamente, a influência do pensamento computacional é perceptível ao aplicar a habilidade de reconhecimento de padrões durante o processo de aprendizado de um novo idioma ou até mesmo para compreender e debater situações que influenciam o cenário global na área da geopolítica.

No relatório “Future of Jobs 2023” do World Economic Forum [7], no topo de habilidades desejáveis para os empregos do futuro estavam o pensamento analítico e o pensamento criativo. Já a versão mais recente do relatório “Future of Jobs 2025” [8] menciona que o pensamento analítico continua sendo a habilidade mais procurada entre empregadores, e que 7 entre 10 dentre os entrevistados consideram essa habilidade essencial no ano de 2025, seguida pela resiliência, flexibilidade e agilidade, juntamente com liderança e influência social.

Essas são algumas das evidências que reforçam que, ainda que estejamos em uma era de grande influência da inteligência artificial, habilidades analíticas, que podem ser desenvolvidas através da incorporação do pensamento computacional ao processo de aprendizagem, são atemporais e úteis, e podem ser aplicadas em diversas áreas da vida de um indivíduo.

2.2 Olimpíadas de conhecimento

As olimpíadas de conhecimento se tornaram conhecidas ao redor do mundo como ações estratégicas para divulgação e promoção das mais diversas áreas das ciências. As olimpíadas científicas, como também são conhecidas, vêm se tornando cada vez mais populares

no Brasil, sendo a Olimpíada Brasileira de Matemática – OBM – a primeira olimpíada científica aplicada no país, por volta do ano de 1979². Atualmente no Brasil, os estudantes podem encontrar cerca de 22 modalidades de olimpíadas para competirem no ensino básico, médio e superior.

Um dos grandes objetivos das olimpíadas de conhecimento é incentivar o estudo de suas áreas através dos desafios propostos em provas, instigando a curiosidade e dedicação por parte dos estudantes. As olimpíadas proporcionam um ambiente de competição saudável, onde os participantes podem testar seus conhecimentos, se desafiarem e, ao conhecerem melhor a área da olimpíada para a qual estão se preparando, podem acabar descobrindo uma vocação e/ou uma profissão para seguir.

Segundo informações do site do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) [9], na primeira Olimpíada de Astronomia participaram somente 21 escolas de 8 cidades brasileiras; já na 25^a edição, aplicada no ano de 2022, a competição contou com a participação de mais de 1,1 milhão de alunos, 12.369 escolas e um total de 59.649 professores, o que mostra a importância das olimpíadas para o engajamento de todos os atores participantes do cenário de educação, bem como a proporção que tomaram no país.

Existem diversas modalidades olímpicas no país e, até fevereiro de 2025, somente no Distrito Federal, pelo menos 14 olimpíadas tiveram seus editais publicados, segundo o Calendário das Olimpíadas Científicas 2025 [10]. Dentre elas estão a Olimpíada Brasileira de Matemática das Escolas Públicas (OBMEP Nacional)³, Olimpíada Nacional em História do Brasil (ONHB)⁴, Olimpíada Brasileira de Astronomia e Astronáutica (OBA)⁵, Olimpíada Brasileira de Agropecuária (OBAP)⁶, Olimpíada Brasileira de Matemática Financeira (OBMF)⁷, Olimpíada Brasileira de Física das Escolas Públicas (OBFEP)⁸, Olimpíada Brasileira de Língua Inglesa (OBLI)⁹, Olimpíada Nacional de Ciências (ONC)¹⁰, entre outras. No âmbito regional, podemos ainda citar a Olimpíada de Matemática do Distrito Federal (OMDF)¹¹, organizada pela Secretaria de Estado de Educação (SEE).

Os benefícios da participação em olimpíadas de conhecimento vão além da recompensa que envolve a aquisição de novas habilidades nas disciplinas. No quesito premiação, as olimpíadas recompensam os melhores desempenhos com medalhas, menções honrosas e,

²<https://www.obm.org.br/>

³<http://www.obmep.org.br/>

⁴<https://www.olimpiadadehistoria.com.br/>

⁵<http://www.oba.org.br/site/>

⁶<https://obap.ifsuldeminas.edu.br/>

⁷<https://www.seletaeducacao.com.br/obmf>

⁸<https://www1.fisica.org.br/~obfep/>

⁹<https://www.seletaeducacao.com.br/obli>

¹⁰<https://www.onciencias.org/>

¹¹<https://www.omdf.com.br/>

em alguns casos, como na OBMEP, até mesmo bolsas de iniciação científica. As medalhas, por si só, poderiam se configurar como um bom prêmio, mas vão além. Algumas universidades conceituadas a nível nacional aceitam medalhas como forma de ingresso no Ensino Superior. A Universidade de São Paulo (USP) é uma delas, segundo o Edital N^o 1/2025 - Competições do conhecimento - 2025 [11], disponibilizado no Diário Oficial do Estado de São Paulo, onde estão descritas as regras de ingresso e a pontuação concedida por medalha.

2.3 Olimpíada Brasileira de Informática

A Olimpíada Brasileira de Informática (OBI¹²) é uma competição de conhecimentos com foco na área da ciência da computação, que funciona nos mesmos moldes de outras olimpíadas de conhecimentos, como as olimpíadas de matemática, física e astronomia. Destinada a estudantes do ensino fundamental, médio e dos anos iniciais do ensino superior, o objetivo da OBI é despertar o interesse dos alunos para a área de informática, que a cada dia se torna mais presente na formação básica.

Organizada pela Sociedade Brasileira de Computação (SBC¹³) desde 1999, a OBI desempenha um papel importante no cenário de olimpíadas de conhecimento. Ao longo de seus mais de 20 anos de história, a OBI conseguiu alcançar o patamar de competição nacional, aumentando sua relevância e atraindo cada vez mais participantes em suas edições. A popularização da área de informática e sua influência cada vez mais presente, desde a formação básica da educação, podem ser alguns dos motivos pelos quais a OBI conseguiu ser reconhecida pelo país.

A competição é dividida em duas modalidades: Iniciação e Programação. A principal diferença entre as duas modalidades se dá pelo fato de que, na modalidade Iniciação, os estudantes não fazem uso de computadores para resolver os problemas de lógica e raciocínio computacional. Nesta etapa, por não saberem programar ainda, os alunos resolvem os desafios usando papel e lápis. Já na modalidade Programação, estudantes de ensino fundamental, médio e anos iniciais do ensino superior fazem uso do computador e de seus conhecimentos em programação para resolver os desafios propostos.

Segundo o regulamento da competição¹⁴, a estrutura das modalidades é definida conforme apresenta o Quadro 1. Após inscreverem os estudantes na competição, a escola interessada em participar da OBI deve garantir que os participantes tenham acesso a um computador para que possam desenvolver as soluções dos desafios em um editor de

¹²<https://olimpiada.ic.unicamp.br/info/>

¹³<https://www.sbc.org.br/>

¹⁴<https://olimpiada.ic.unicamp.br/info/regulamento/>

texto, e que também possam submeter os programas na plataforma da organização. A competição é dividida em três fases: Local, Estadual e Nacional.

Quadro 1: Modalidades, níveis e elegibilidade da OBI

Modalidade	Nível	Elegibilidade
Iniciação	Júnior	alunos do 4º e 5º anos do Ensino Fundamental.
	1	Alunos do 6º e 7º anos do Ensino Fundamental.
	2	Alunos do 8º e 9º anos do Ensino Fundamental.
Programação	Júnior	alunos do 4º e 5º anos do Ensino Fundamental.
	1	Alunos do 6º e 7º anos do Ensino Fundamental.
	2	Alunos do 8º e 9º anos do Ensino Fundamental.
	Sênior	Alunos do 4º ano do Ensino Técnico e alunos que estão cursando o 1º ano do Ensino Superior pela primeira vez.

Fonte: o Autor.

A fase Local ocorre na escola pela qual o participante se inscreveu e são convocados para a etapa Estadual os participantes que na etapa Local:

- (a) obtiverem ao menos 1/3 da pontuação total da prova;
- (b) estiverem entre os melhores colocados da etapa Local em cada escola, obedecendo ao limite de até 15% do total de participantes efetivos da escola na respectiva modalidade e nível;
- (c) estiverem entre os melhores colocados nacionalmente na etapa Local, obedecendo ao limite de até 5% do total de participantes efetivos na respectiva modalidade e nível.

A fase Estadual pode ou não ser realizada na escola pela qual o participante se inscreveu e são convocados para a etapa Nacional os participantes que na etapa Estadual:

- (a) obtiverem ao menos 1/3 da pontuação total da prova;
- (b) estiverem entre os melhores colocados na etapa Estadual em cada Estado, obedecendo ao limite de até 15% do total de participantes efetivos do Estado na respectiva modalidade e nível;
- (c) estiverem entre os melhores colocados nacionalmente na etapa Estadual, obedecendo ao limite de até 5% do total de participantes efetivos do Estado na respectiva modalidade e nível.

A etapa Nacional é a última fase da competição. A classificação final dos competidores é dada pelos seus respectivos desempenhos na prova, em cada modalidade e nível. Em

caso de empate, o desempenho na etapa Estadual é usado como critério de desempate. Caso o empate ainda persista, o desempate é feito usando o desempenho do participante na etapa Local.

O conteúdo cobrado na OBI é apresentado na sessão de “ementa” presente no site oficial da OBI [12]. Há uma diversidade de assuntos abrangidos, dos quais pode-se mencionar fundamentos da matemática, como álgebra, geometria e lógica; conceitos da matemática discreta, conceitos de grafos e conceitos fundamentais da computação, que são essenciais para a codificação das resoluções dos desafios, onde os alunos precisarão ter o domínio das ferramentas de escrita, compilação e submissão dos códigos. Com o grande volume de conteúdos, em contraste com o número de questões apresentadas nas provas, que pode variar entre 4 e 5 questões, torna-se um desafio abordar cada um dos tópicos mencionados, de modo que os desafios acabam por abordar combinações entre tópicos, o que torna a prova um tanto desafiadora.

A OBI ainda conta com evento extra, a Competição Feminina da OBI (CF-OBI), que visa promover a participação feminina na competição. É uma competição separada, e não é afetada pelo desempenho das participantes na OBI, e que engloba somente uma etapa de prova. As participantes podem ser inscritas nos níveis Júnior, Nível 1 ou Nível 2, e todas as competidoras recebem um certificado de participação.

A premiação de medalhas para cada nível é destinada às competidoras com os melhores resultados, e como premiação adicional, as duas melhores competidoras dos níveis Júnior e 1 são convidadas a participar da Semana Olímpica da OBI¹⁵ no Instituto de Computação da Unicamp. Já as duas melhores colocadas do Nível 2 são convidadas a participar da Seletiva para Competições Internacionais, que também ocorre na Semana Olímpica da OBI.

2.4 Aprendizagem Baseada em Problemas

A aprendizagem baseada em problemas (*Problem-Based Learning* - PBL) é uma abordagem pedagógica construtivista, onde o aluno é posicionado como o protagonista do processo educativo e usa os problemas propostos para promover o aprendizado. Essa metodologia teve origem na educação médica, na Universidade McMaster [13], e hoje é uma abordagem amplamente usada em áreas chamadas de STEM, sigla em inglês usada para descrever as áreas de ciência, tecnologia, engenharia e matemática.

Na medicina, podemos pensar que exista, hipoteticamente, um caso de uma doença rara que ainda não tenha registros de tratamento ou mesmo de casos registrados. A abordagem da PBL parte do princípio de que o problema proposto serve como um caso de

¹⁵[urlhttps://olimpiada.ic.unicamp.br/semana/](https://olimpiada.ic.unicamp.br/semana/)

estudo, e que entender o problema é o passo inicial para que sejam definidos quais assuntos serão estudados e/ou aprofundados para que o problema possa, enfim, ser resolvido.

Contextualizando para a área de computação, podemos ver uma semelhança com princípios do pensamento computacional, como a decomposição de problemas, e a abordagem da PBL. A semelhança desses assuntos é perceptível, uma vez que em ambos o aluno encontra no centro do processo de resolução de um problema e é desafiado a buscar conhecimentos para que possa chegar a um resultado. Em problemas de lógica, a possibilidade de quebrar um desafio em pequenas partes potencializa o sentimento de aprendizado, graças ao feedback instantâneo que é proporcionado quando o aluno consegue resolver parcialmente cada etapa dos desafios, criando assim ciclos de motivação que reforçam os assuntos aprendidos e fortalecem a autoconfiança dos alunos.

No contexto específico da preparação para a OBI, a PBL apresenta uma certa convergência com as demandas da competição, uma vez que é exigido dos alunos o pensamento crítico e analítico, habilidades que são desenvolvidas no processo de resolução de problemas. Outra habilidade que pode ser desenvolvida pelos alunos na preparação é o trabalho em equipe. Apesar de a competição ser disputada de forma individual, exclusivamente, os alunos podem praticar suas habilidades interpessoais ao formarem grupos de estudos durante a preparação, onde são estimulados a compartilhar suas resoluções para os desafios e também a pensar coletivamente em como abordar desafios mais elaborados.

A sinergia entre a PBL e o treinamento para a Olimpíada Brasileira de informática é notável. A própria natureza da competição, que apresenta problemas complexos que exigem soluções algorítmicas criativas, funciona como o gatilho do processo de aprendizagem. Nesta dinâmica, os problemas apresentados aos alunos promovem a busca do conteúdo necessário para avançar e, com a mediação do professor, os conhecimentos podem ser obtidos para que os estudantes possam alcançar a resolução dos problemas. Esse processo torna a aprendizagem mais significativa e trabalha também a autonomia e a resiliência dos estudantes, habilidades cruciais para um bom desempenho na prova.

2.5 Trabalhos Correlatos

Assim como o presente trabalho, outros projetos também foram executados visando preparar alunos para competir na Olimpíada Brasileira de Informática. Nesta seção serão pontuadas as diferenças de trabalhos considerados correlatos e o trabalho de preparação executado na escola por este autor.

No trabalho de Rossi et al [14] é possível observar que o projeto de extensão que existia desde 2017 foi criado com o intuito de fomentar a participação das escolas de Videira-SC na Olimpíada Brasileira de Informática. Desde a concepção até os primeiros passos do

projeto, nota-se uma diferença na abordagem, uma vez que o projeto de extensão mencionado envolve apoiar escolas desde o cadastro das mesmas como participantes da OBI até dos responsáveis pela aplicação da prova e os alunos participantes. A Instituição fica sempre à disposição para tirar dúvidas, ajuda a traçar a estratégia de preparação, porém o curso de preparação de lógica e programação de computadores foi oferecido somente aos alunos que chegaram na fase nacional da competição. Essa talvez seja a maior diferença a este trabalho, visto que a ação principal envolveu a ministração de aulas preparatórias para as primeiras etapas da OBI de 2024, evoluindo para um acompanhamento dos alunos que eventualmente passassem para as próximas etapas.

Em outro exemplo de preparatório para a OBI, Ramos et al [15] relatam o processo de preparação feito com alunos de ensino médio, assim como neste trabalho, porém houve uma divisão dos cursos ministrados, sendo o primeiro sobre programação básica, usando o Scratch como ferramenta, e posteriormente usando Python para dar sequência aos tópicos a serem aprendidos. Em um segundo momento, houve a ministração de um curso com técnicas de programação e resolução de problemas mais avançados. Nota-se que a ministração dos cursos foi feita no ambiente da Instituição Federal e as aulas foram ministradas por professores universitários, tendo a ajuda de discentes da UFPB como monitores durante as aulas. A realização de uma mini-maratona ao final da introdução dos conceitos básicos, com o intuito de simular um ambiente de competição e proporcionar aos alunos uma ideia do que é a prova da OBI, foi uma semelhança notável com o presente trabalho.

Gonçalves [16] ressalta que o treinamento para olimpíadas é essencial para que os processos internos funcionem e se tornem fluidos. No projeto citado, há o desenvolvimento de um trabalho com o primeiro, segundo e terceiro ano do Curso Técnico em Informática para Internet Integrado ao Ensino Médio, bem como o primeiro período do Curso Técnico em Informática para Internet Concomitante/Subsequente. A partir dos resultados e do interesse dos alunos, foi criada uma fábrica de software pela instituição, projeto que incentivou os alunos a terem um contato real com programação e solução de problemas. Tanto o trabalho de Gonçalves quanto o presente trabalho trazem esse apelo de desenvolver a comunidade local e promover o ensino de informática como a emancipação dos alunos.

Ainda sobre o ensino de informática através de um curso preparatório para a Olimpíada Brasileira de Informática, Dos Santos et al [17] citam a experiência de ensino de conceitos relacionados à lógica de programação, com o foco em resolução de problemas do nível da OBI, para alunos do sexto e sétimo anos do Ensino Fundamental. O mais interessante desse projeto é que, pelo fato de trabalharem com alunos mais novos, houve uma decisão de trabalharem primeiramente com jogos, ideias lúdicas para introdução de conceitos básicos, para mais na frente entrarem em resoluções de provas passadas da Olimpíada. Diferente do projeto executado deste trabalho, que abordou a submissão de problemas

desde o dia zero.

Lopes et al [18] citam o desempenho tido como satisfatório dos alunos que participaram do curso de preparação oferecido pelo IF Goiano. Apesar da estrutura de curso ser bem semelhante, especialmente na questão da escolha da plataforma URI Online Judge (atual BeeCrowd) para a submissão de problemas, houve uma divergência na escolha da linguagem abordada. O Instituto Federal Goiano adotou Python como a linguagem padrão, enquanto presente trabalho decidiu começar o curso preparatório usando a linguagem C, fato que será melhor explicado na seção 3.2.

Moreira [19] apresenta uma experiência similar à deste trabalho, voltada à promoção do pensamento computacional por meio de um treinamento para a OBI. Embora compartilhe métodos semelhantes, o autor relata que parte das aulas incluiu conteúdos matemáticos, como frações, segmentos de reta e noções de trigonometria. Essa abordagem diverge da proposta do presente estudo, que se concentrou exclusivamente nos fundamentos de lógica computacional, revisando conteúdos complementares apenas quando necessário.

Outra diferença relevante está no contexto institucional. O treinamento de Moreira foi realizado em uma escola técnica de ensino médio voltada à tecnologia da informação, enquanto o presente estudo ocorreu em uma escola pública de ensino médio regular.

Por fim, observou-se uma discrepância significativa no engajamento dos participantes. Dois fatores podem explicar essa diferença: (i) a seleção de alunos, que neste estudo envolveu exclusivamente estudantes já matriculados em uma disciplina eletiva de preparação para olimpíadas, o que favoreceu a participação de interessados em programação; e (ii) a dinâmica das aulas, que desde o primeiro encontro priorizou atividades práticas e resolução de problemas, garantindo que todos os encontros incluíssem programação efetiva.

Capítulo 3

Materiais e Métodos

Este capítulo tem como objetivo descrever, de forma detalhada, os aspectos práticos e metodológicos que orientaram a execução deste projeto. A primeira seção aborda a pesquisa bibliográfica, tornando explícitos os procedimentos usados na procura de referências teóricas, citando as bases de dados usadas, os termos usados nas pesquisas e o volume de resultados obtidos. Em seguida, apresenta-se a estrutura idealizada para a execução do presente trabalho, citando as reuniões realizadas previamente, os critérios abordados para a escolha da linguagem de programação que seria ensinada, as estratégias de seleção de plataforma, exercícios e conteúdos a serem abordados nas aulas ministradas na escola. Em seguida, são mencionados os eventos planejados para que os alunos pudessem ter um cenário de imersão em competições de programação, visando um melhor preparo para o dia da aplicação das provas. Por fim, discute-se um modelo de pesquisa semi-estruturada escolhido para obter dados a fim de analisar o projeto de forma qualitativa.

3.1 Pesquisa Bibliográfica

A pesquisa bibliográfica para o presente trabalho foi conduzida com o objetivo de levantar os trabalhos relacionados à preparação de estudantes para a Olimpíada Brasileira de Informática, tendo como foco estudantes do Ensino Básico e do Ensino Médio. Para tal propósito, foi realizada uma busca em duas bases principais de artigos acadêmicos. A primeira base usada foi o Portal de Periódicos da Capes¹, acessado através do login institucional pela Universidade de Brasília, e a segunda base utilizada foi o *Google Scholar*², ferramenta de busca acadêmica aberta.

Inicialmente, as buscas foram feitas usando o termo “OBI”, porém essas pesquisas trouxeram resultados de baixa relevância para os fins pretendidos. Em seguida, optou-se

¹<https://www.periodicos.capes.gov.br/index.php/acesso-cafe.html>

²<https://scholar.google.com.br/?hl=pt>

por termos mais precisos e alinhados com o tema da pesquisa, como “Olimpíada Brasileira de Informática”. Desta vez, os resultados apresentaram maior relação com o objetivo da pesquisa. Durante a busca, os termos foram pesquisados de forma separada e também com combinações, como os exemplos abaixo:

- (a) “OBI” AND “Ensino Médio”
- (b) “OBI” AND “Curso Preparatório”;
- (c) “Olimpíada Brasileira de Informática”.

Considerando a base de dados do Portal de Periódicos da CAPES, foram encontrados 15 artigos, enquanto na base de dados Google Scholar, foram inspecionadas as 5 primeiras páginas, totalizando aproximadamente 50 artigos. Após a leitura dos resumos e partes introdutórias, foram identificados 5 artigos – dentre os 20 mais relevantes encontrados – que se assemelham à ideia do presente trabalho. Esses artigos foram escolhidos levando em conta alguns critérios, como relevância temática e data de publicação, uma vez que foi dada preferência por artigos mais recentes, contexto educacional compatível e dinâmica de aulas.

Ao final do processo de busca de trabalhos correlatos que se assemelhassem à proposta deste trabalho, foram escolhidos os artigos descritos no Quadro 2.

Quadro 2: Trabalhos correlatos identificados na pesquisa bibliográfica.

#	Título	Autor(es)	Ano
1	Olimpíada Brasileira de Informática	Rossi <i>et al.</i>	2021
2	Ensino de Programação para Olimpíada Brasileira de Informática	Ramos <i>et al.</i>	2015
3	Raciocínio Lógico e Computação: Descobrimos Estratégias de ensino por meio da Olimpíada Brasileira de Informática	dos Santos <i>et al.</i>	2015
4	Uma análise da influência da aplicação de curso preparatório de olimpíadas científicas de informática com estudantes de Ensino Médio	Lopes <i>et al.</i>	2018
5	A importância de treinamentos para Olimpíada Brasileira de Informática	Gonçalves, F. F.	2021

Fonte: o Autor.

3.2 Treinamento

O treinamento foi idealizado a partir de uma conversa com a professora de matemática responsável por uma disciplina de preparação para competições em uma escola pública do

Ensino Médio de Taguatinga/DF. No semestre da aplicação do curso de preparação para a OBI, a escola disponibilizava essa disciplina de preparação de estudantes para a participação em olimpíadas de conhecimento em matemática – OBMEP e OMDF. Nesse cenário, foi identificada a possibilidade de inserção de um projeto de capacitação dos estudantes para a participação na OBI, concomitantemente com as outras duas olimpíadas.

A concepção do projeto considerou o tempo disponível para as aulas e treinamentos dos fundamentos de lógica de programação – levando em conta que os encontros ocorreriam uma vez por semana, às quintas-feiras, no laboratório de informática, e teriam a duração de aproximadamente 100 minutos por aula – e o dia da aplicação da prova da Fase 1 da OBI, dia 14 de junho de 2024, no período da manhã.

Definidas as datas das aulas e da aplicação da prova, com o apoio dos professores envolvidos no projeto, foi feita uma análise das provas de edições anteriores e da ementa disponível no site da OBI³. Após esta análise, constatou-se que, com o tempo disponível entre o início das atividades na escola e a data de aplicação da prova da OBI, não seria possível abordar em sala de aula todos os conhecimentos mencionados na ementa, na qual, para o nível 2 – categoria na qual participam estudantes do 2º e 3º ano do Ensino Médio – são listados tópicos como Algoritmos e Estruturas de Dados, Grafos, Fundamentos de Lógica, Matemática Discreta, Fundamentos de Análise de Algoritmos, Algoritmos de Busca e Ordenação, dentre outros que demandariam mais tempo para a compreensão e prática de exercícios. Portanto, decidiu-se que, para capacitar os estudantes a competir na Fase 1, uma estratégia seria abordar os conceitos de entrada e saída de dados, estruturas de controle de fluxo (condicionais), laços de repetição e *arrays*. Houve então um entendimento de que esse conjunto de conteúdos poderia proporcionar uma base sólida de conhecimentos para os participantes, tendo em vista a prova da Fase 1.

O início das aulas foi planejado para o dia 25 de abril de 2024, e a primeira aula⁴ do curso teria o propósito de motivar alunos matriculados na disciplina eletiva “Tópicos em Matemática” a participar do curso de preparação para a Olimpíada Brasileira de Informática. Essa estratégia foi traçada a fim de selecionar os alunos que estavam em um ritmo de preparação mais avançado devido ao fato da proximidade das provas das olimpíadas de matemática, e filtrar estudantes que realmente teriam interesse em conhecer ou se aprofundar na área de computação. Após reunião entre os membros participantes do projeto, ficou decidido um conjunto de conteúdos a ser ministrado até o dia da prova da Fase 1 da OBI, que estava prevista para o dia 14/06/2024. Além disso, foi planejada uma assistência para além da Fase 1, abrangendo conceitos importantes para a aplicação da prova da Fase 2 da OBI, em caso de aprovação de algum estudante na primeira etapa. As

³https://olimpiada.ic.unicamp.br/prepare/ementas/ementa_prog/

⁴disponível em <http://bit.ly/3IzdfH6>

datas dos encontros planejados se encontram no Quadro3, juntamente com os conteúdos selecionados e o discente responsável pela aula.

Quadro 3: Plano Geral do Curso de Preparação para a OBI.

Aula	Data	Conteúdo	Discente
01	25/04/24	Apresentação do curso	Brunno
02	02/05/24	<code>printf()</code>	Brunno
03	09/05/24	<code>scanf()</code>	Brunno
04	16/05/24	<code>if/else</code>	Brunno
05	23/05/24	<code>while</code>	Brunno
06	06/06/24	<code>for</code>	Brunno
07	13/06/24	Simulado 1	Lucas
08	20/06/24	Exercícios de laços de repetição	Lucas
09	27/06/24	Exercícios de laços de repetição	Lucas
10	04/07/24	<i>Arrays</i>	Lucas
11	11/07/24	<i>Arrays</i>	Lucas
12	18/07/24	Exercícios de <i>Arrays</i>	Lucas
13	25/07/24	Simulado 2	Brunno

Fonte: o Autor.

As aulas foram planejadas para serem ministradas no laboratório de informática da escola, onde cada aluno teria à sua disposição uma máquina com o sistema operacional GNU/Linux. As aulas teriam o intuito de trazer uma abordagem focada na resolução de problemas e na submissão das soluções propostas para os desafios em plataformas como BeeCrowd⁵, MOJ⁶ e Codeforces⁷. A escolha por usar as plataformas citadas teve como objetivo criar e simular ambientes semelhantes aos utilizados em eventos de programação competitiva, para que os estudantes desenvolvessem familiaridade com o processo de resolução de problemas formatados em texto e com casos de teste, e também habituá-los com o processo de submissão de códigos, uma vez que no dia da aplicação da prova da OBI a dinâmica seria semelhante à descrita.

A plataforma BeeCrowd⁸ foi escolhida como a principal fonte para os desafios propostos, e essa escolha se deu porque a plataforma disponibiliza problemas na língua materna dos estudantes (Português) e também pela organização de problemas na plataforma. O site do BeeCrowd disponibiliza uma área de iniciantes, que é formada por problemas ordenados em ordem crescente de dificuldade (conforme mostra a Figura 3.1), partindo dos desafios mais simples e triviais como o problema “Hello World”⁹ – no qual o participante

⁵<https://judge.beecrowd.com/pt>

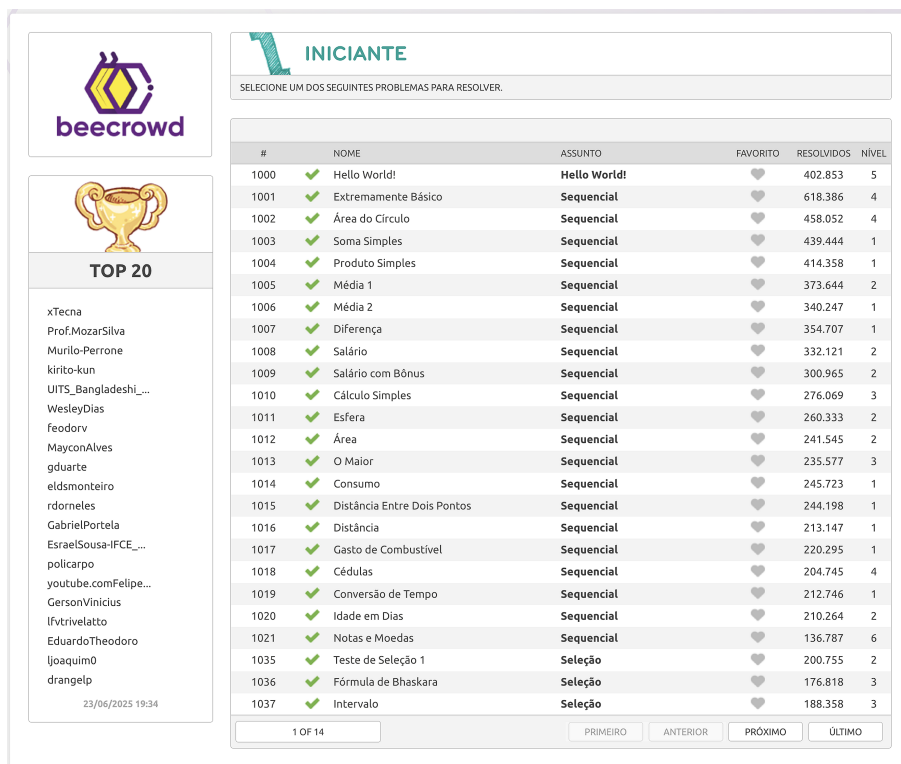
⁶<https://moj.naquadah.com.br/cgi-bin/index.sh>

⁷<https://codeforces.com/>

⁸<https://judge.beecrowd.com/pt>

⁹<https://judge.beecrowd.com/pt/problems/view/1000>

Figura 3.1: Lista de desafios para iniciantes, em ordem de dificuldade dos problemas.



#	NOME	ASSUNTO	FAVORITO	RESOLVIDOS	NÍVEL
1000	✓ Hello World!	Hello World!	♥	402.853	5
1001	✓ Extremamente Básico	Sequencial	♥	618.386	4
1002	✓ Área do Círculo	Sequencial	♥	458.052	4
1003	✓ Soma Simples	Sequencial	♥	439.444	1
1004	✓ Produto Simples	Sequencial	♥	414.358	1
1005	✓ Média 1	Sequencial	♥	373.644	2
1006	✓ Média 2	Sequencial	♥	340.247	1
1007	✓ Diferença	Sequencial	♥	354.707	1
1008	✓ Salário	Sequencial	♥	332.121	2
1009	✓ Salário com Bônus	Sequencial	♥	300.965	2
1010	✓ Cálculo Simples	Sequencial	♥	276.069	3
1011	✓ Esfera	Sequencial	♥	260.333	2
1012	✓ Área	Sequencial	♥	241.545	2
1013	✓ O Maior	Sequencial	♥	235.577	3
1014	✓ Consumo	Sequencial	♥	245.723	1
1015	✓ Distância Entre Dois Pontos	Sequencial	♥	244.198	1
1016	✓ Distância	Sequencial	♥	213.147	1
1017	✓ Gasto de Combustível	Sequencial	♥	220.295	1
1018	✓ Cédulas	Sequencial	♥	204.745	4
1019	✓ Conversão de Tempo	Sequencial	♥	212.746	1
1020	✓ Idade em Dias	Sequencial	♥	210.264	2
1021	✓ Notas e Moedas	Sequencial	♥	136.787	6
1035	✓ Teste de Seleção 1	Seleção	♥	200.755	2
1036	✓ Fórmula de Bhaskara	Seleção	♥	176.818	3
1037	✓ Intervalo	Seleção	♥	188.358	3

Fonte: Beecrowd <https://beecrowd.com/>.

deve imprimir a mensagem “*Hello World*”, seguida do caractere de quebra de linha – até problemas que exigem conhecimento prévio em alguma linguagem, como o de estruturas condicionais, por exemplo, usadas no desafio “Imposto de Renda”¹⁰, no qual os alunos precisam escrever um programa que calcule o valor de Imposto de Renda devido, de acordo com o salário dado como entrada.

Após decidirmos a plataforma na qual os alunos submeteriam suas soluções, foi necessário escolher qual linguagem de programação acompanharia o ensino dos conceitos de lógica. De acordo com o regulamento¹¹, as linguagens aceitas pela OBI são C, C++, Java, Javascript e Python. Optou-se pela adoção da linguagem C principalmente por sua tipagem explícita e baixo nível de abstração em comparação com outras linguagens como Python e JavaScript. Essa característica permite que os estudantes tenham uma compreensão mais profunda e precisa sobre os tipos de dados manipulados nos programas e seus respectivos comportamentos. Esses aspectos são de grande importância para a formação inicial na área.

Além disso, a escolha da linguagem também considerou a experiência prévia dos docentes em formação envolvidos no projeto, o que facilitou o planejamento didático e a

¹⁰<https://judge.beecrowd.com/pt/problems/view/1051>

¹¹<https://olimpiada.ic.unicamp.br/info/regulamento/>

dinâmica das aulas. Outro fator relevante é que a linguagem C apresenta um bom equilíbrio entre controle e desempenho, promovendo a reflexão sobre a eficiência e a execução dos programas desenvolvidos pelos estudantes, conceitos que são fundamentais na formação de uma base sólida em computação.

Embora a escolha da linguagem C ser justificada por sua eficiência e controle, reconheceu-se o desafio pedagógico de introduzi-la a estudantes com pouca ou nenhuma experiência prévia em programação, dada a sintaxe rigorosa da linguagem e a necessidade de entendimento mais profundo sobre tipos de dados e alocação de memória, em tópicos mais avançados. Para mitigar essa dificuldade, a estratégia de ensino focou em uma progressão gradual, começando com o ensino de conceitos de entrada e saída de dados (`printf()` e `scanf()`) e utilizando a plataforma BeeCrowd para fornecer feedback imediato e automatizado aos estudantes, permitindo-os ver seus erros e aprender através de experimentação e correção de código.

A avaliação dos alunos foi um dos desafios identificados logo na fase de planejamento do projeto, uma vez que trazer exemplos de problemas de programação competitiva serviria como motivação para a apresentação de novos conteúdos. Nesse cenário, onde a cada nova estrutura aprendida os alunos acompanhariam um exemplo na projeção em sala de aula, seria de grande importância garantir que os alunos estariam aprendendo a aplicar o conteúdo aprendido na solução de desafios, e não somente copiando códigos na linguagem de programação proposta. A partir dessa preocupação, foi selecionado um conjunto de problemas da plataforma BeeCrowd, classificados de acordo com os conteúdos aprendidos. O conjunto de exercícios foi separado em 5 listas conforme mostram os Quadros 4, 5, 6, 7 e 8. O nível de cada problema, apresentado na terceira coluna, é fornecido pela própria plataforma, e constitui um índice de dificuldade do problema numa escala de 1 a 10, sendo 1 o nível mais fácil e 10 o mais difícil.

Quadro 4: Lista 1 – Variáveis e Operadores.

Identificador	Nome	Nível
BEE 1002	Área do Círculo	4
BEE 1003	Soma Simples	1
BEE 1006	Média 2	1
BEE 1007	Diferença	1
BEE 1011	Esfera	2
BEE 1012	Área	2
BEE 1013	O Maior	3
BEE 1015	Distância Entre Dois Pontos	1
BEE 1017	Gasto de Combustível	1
BEE 1019	Conversão de Tempo	1

Fonte: o Autor.

Quadro 5: Lista 2 – Condicionais.

Numero	Nome	Nível
BEE 1036	Fórmula de Baskhara	3
BEE 1037	Intervalo	3
BEE 1038	Lanche	1
BEE 1041	Coordenadas de um Ponto	3
BEE 1042	Sort Simples	2
BEE 1043	Triângulo	2
BEE 1044	Múltiplos	2
BEE 1051	Imposto de Renda	2
BEE 1061	Tempo de um Evento	3
BEE 2375	Sedex	1

Fonte: o Autor.

Quadro 6: Lista 3 – Laços.

Numero	Nome	Nível
BEE 1021	Notas e Moedas	6
BEE 1059	Números Pares	1
BEE 1072	Intervalo 2	1
BEE 1113	Crescente e Decrescente	1
BEE 1132	Múltiplos de 13	1
BEE 1151	Fibonacci Fácil	2
BEE 1154	Idades	1
BEE 1160	Crescimento Populacional	5
BEE 1164	Número Perfeito	2
BEE 1165	Número Primo	2

Fonte: o Autor.

Quadro 7: Lista 4 – Vetores.

Numero	Nome	Nível
BEE 1176	Fibonacci em Vetor	3
BEE 1180	Menor e Posição	3
BEE 1181	Linha na Matriz	3
BEE 1929	Triângulo	3
BEE 1930	Tomadas	1
BEE 1960	Numeração Romana para Números de Página	2
BEE 1961	Pula Sapo	2
BEE 1973	Jornada nas Estrelas	7
BEE 1984	O Enigma do Pronalândia	2
BEE 2028	Sequência de Sequência	5

Fonte: o Autor.

Quadro 8: Lista 5 – Strings

Numero	Nome	Nível
BEE 1332	Um-Dois-Três	2
BEE 1581	Conversa Internacional	2
BEE 1632	Variações	1
BEE 1768	Árvore de Natal	1
BEE 1803	Matring	1
BEE 1828	Bazinga	4
BEE 1871	Zero vale Zero	1
BEE 1914	De Quem é a Vez?	2
BEE 2137	Biblioteca do Senhor Severino	1
BEE 2694	Problema com a Calculadora	1

Fonte: o Autor.

A elaboração de listas com os problemas selecionados contribuiria para a identificação de estudantes que apresentassem dificuldades na resolução dos desafios propostos. A plataforma BeeCrowd Academic¹², onde as listas seriam disponibilizadas, traria ao professor métricas como o número de submissões, tentativas, acertos e erros, permitindo identificar os problemas que os estudantes considerariam mais fáceis – aqueles com maior número de submissões e acertos – quanto os problemas que eles teriam maior nível de dificuldade, os quais teriam menor número de submissões e acertos. Essa estratégia nos possibilitaria uma intervenção direcionada, permitindo a abordagem dos alunos com maiores dificuldades e possibilitando o trabalho específico de conceitos de lógica ou do pensamento computacional que, porventura, não viessem a ser completamente compreendidos.

3.3 Eventos

Para preparar os alunos para as aplicações oficiais da OBI, foram organizados simulados denominados Mini-OBIs, cujo objetivo era familiarizar os participantes com a dinâmica de resolução de problemas em tempo limitado. Cada simulado foi planejado para contemplar os conteúdos exigidos nas provas oficiais, e suas datas foram definidas de forma a ocorrer o mais próximo possível das respectivas etapas da OBI.

A seguir, são descritos os procedimentos adotados nos dias de simulado, bem como as questões selecionadas para cada edição. Na sequência, apresentam-se os procedimentos planejados para a aplicação das provas oficiais da Olimpíada.

¹²<https://academic.becrowd.com/pt>

3.3.1 Mini-OBI 1

O primeiro simulado foi planejado para o dia 08 de junho de 2024. O laboratório de informática da escola, mesmo ambiente no qual os estudantes iriam participar da prova da Fase 1 da OBI, foi o local escolhido para que a dinâmica pudesse ocorrer. Foram convidados todos os estudantes que, em algum momento, participaram do curso ministrado.

Inicialmente, os estudantes precisariam ter um cadastro ativo na plataforma Codeforces¹³, onde estariam disponíveis os desafios para os participantes. Dessa forma, os primeiros 30 minutos serviriam de suporte para que os alunos pudessem se cadastrar, logar na plataforma e acessar o link do desafio, disponibilizado no projetor da sala de aula. Além disso, os alunos teriam esse tempo para tirar quaisquer dúvidas antes do início da dinâmica. Após o início da dinâmica, os alunos teriam disponíveis 3 horas para submeterem o máximo de resoluções possíveis.

Foram escolhidos 7 desafios para o primeiro simulado, considerando os conteúdos que seriam abordados em sala de aula e os conteúdos identificados como prováveis temas a serem cobrados na prova da Fase 1 da OBI. Cada um dos problemas é explicado abaixo:

A - Média Final

Neste primeiro desafio, o objetivo é exercitar conceitos básicos de saída e entrada de dados, bem como a aplicação de operações aritméticas simples. Para resolvê-lo os alunos precisam criar um programa que calcule a média final de um aluno a partir da nota de três avaliações. A média é obtida por meio da fórmula

$$MF = \frac{P1 + P2 + P3}{3}$$

onde $P1$, $P2$ e $P3$ são notas de 0,0 a 10,0.

B - Impa ou pá?

Este problema propõe a simulação de uma brincadeira tradicional de “Par ou Ímpar”. O desafio consiste em ler dois números inteiros de 0 a 5, representando as escolhas de Lucas e Pedro. É enfatizado que Pedro sempre fica com a escolha de um resultado ímpar, de forma que se o resultado for ímpar, o nome de Pedro deve ser impresso, seguido de uma quebra de linha. Se o resultado for par, o nome de Lucas deve ser impresso, seguido de uma quebra de linha. Esse desafio promove o uso de condicionais, promovendo o uso das estruturas **if** e **else**.

¹³<https://codeforces.com/>

C - Plantação de Soja

Este desafio foi incluído por reforçar habilidades matemáticas relacionadas à porcentagem e à manipulação da aritmética de ponto flutuante. O problema pede que seja calculada a área original de um terreno, considerando que o valor de entrada corresponde a um aumento de 25% do terreno original como uma medida do governo de incentivar o plantio de soja.

D - Torneio de Educação Física

O quarto problema é um desafio de viabilidade na formação de grupos de alunos para um torneio esportivo. A partir de dois inteiros lidos, A e N , que representam o número de alunos do professor, e o número mínimo de alunos por grupo, respectivamente, o estudante deve verificar se é possível organizar os participantes em equipes homogêneas, atendendo às seguintes regras:

- Todos os alunos devem participar da atividade;
- Nenhum aluno pode fazer parte de mais de uma equipe;
- Todas as equipes devem ter o mesmo número de alunos.

Caso seja possível formar um grupo, a palavra “sim” deve ser impressa, caso o contrário, a palavra “não” deve ser impressa. O critério para a validação da entrada está na verificação da divisibilidade de A por N , incentivando que os alunos usem o operador resto da divisão para verificar se $A \bmod N = 0$.

E - Caminhão de Carga

O problema das cargas envolve o uso de laços de repetição. Neste desafios, os alunos precisam ler um inteiro N , que representa a quantidade de contêineres a serem transportados. A partir do valor de N os alunos precisam descobrir a quantidade total de peso a ser carregado pelos caminhões.

F - Dados

O penúltimo desafio envolve a identificação de elementos ocultos em um dado de seis faces. O objetivo do desafio é encontrar a soma das faces ocultas, tendo como referência as faces aparentes. Para que o estudante consiga ler a combinação de faces aparentes, é preciso usar laços de repetição e a solução pode ser encontrada subtraindo a soma dos números aparentes da soma total das faces do dado.

G - Tamanho, em blocos, de um arquivo

O sétimo e último desafio propõe um problema de alocação de espaço em disco por meio de blocos fixos. Neste desafio, os estudantes precisam ler dois inteiros B e N , que representam, respectivamente, o tamanho de um bloco e o tamanho de um arquivo, ambos em bytes. O objetivo é determinar o número mínimo de blocos necessários para armazenar um arquivo de N bytes em um sistema com blocos de B bytes. Este desafio foi escolhido para que os alunos pudessem trabalhar com o conceito de divisão inteira.

Para o encerramento do simulado, foi planejada uma celebração e premiação dos participantes que obtivessem a maior pontuação, assim como a resolução dos problemas presentes no simulado, de forma coletiva, visando esclarecer dúvidas e incentivar os alunos a debaterem sobre suas formas de resolver os desafios.

3.3.2 Mini-OBI 2

O segundo simulado foi planejado para a semana da prova da Fase 2 da OBI, que ocorreria no dia 16 de agosto de 2024. Os mesmos procedimentos adotados na realização do primeiro simulado foram adotados para o segundo evento, contando apenas com a alteração dos exercícios escolhidos, uma vez que seria necessário adaptar o nível de dificuldade das questões, com o objetivo de preparar os estudantes para a prova da etapa seguinte. A descrição dos exercícios e os requisitos para as resoluções encontram-se descritas nas próximas subseções.

Entrando em Forma

Este problema apresenta uma aplicação direta da fórmula do índice de Massa Corporal (IMC), e pede para que o estudante calcule o a massa que uma pessoa deve perder para alcançar um IMC desejado. O cálculo é feito através da relação já conhecida do IMC

$$IMC = \frac{m}{h^2}$$

Esse desafio explora o uso de variáveis de ponto flutuante, manipulação de expressões algébricas e saída de dados com precisão numérica.

Forca I

O desafio da Forca tem como propósito promover a prática de leitura e manipulação de strings. Inspirado no desafio tradicional da Forca, o problema pede para que os estudantes contem o número total de consoantes e de vogais em uma palavra com letras maiúsculas,

e os imprima ao final do programa. Para resolver esse desafio, os alunos precisam ter o conhecimento básico de strings, leitura de caracteres e manipulação de *arrays*.

Formatação de Campos

Este desafio propõe a manipulação de strings. Seu objetivo é formatar um número de telefone inserido no formato bruto com 10 dígitos contínuos (DD) NNNNNNNN para o formato padrão de armazenamento (DD) NNNN-NNNN. Para resolver este desafio os estudantes precisariam saber manipular strings.

Banco de Horas

Este desafio simula o controle de jornada de trabalho de um funcionário ao longo dos dias úteis de um mês. A tarefa consiste verificar se a carga horária mensal está abaixo, acima ou igual à carga mínima mensal esperada, com base em 8 horas diárias. A partir de um inteiro D , que representa o número de dias úteis trabalhados, seguido de uma sequência de D inteiros H_i , representando as horas trabalhadas em cada dia, o estudante precisa calcular o total de horas trabalhadas e imprimir se o funcionário deve pagar horas, ficar com horas no banco ou se o funcionário trabalhou o número de horas esperado. Este problema foi escolhido pois reforça o uso de laços de repetição, trabalha com somatórios e estruturas condicionais.

Self-Service

Este desafio aborda o cálculo inverso do preço por unidade de massa com desconto percentual aplicado. O problema pede que o estudante determine o valor pago por quilo de comida em um restaurante, a partir do valor final pago, um desconto e da quantidade consumida (em gramas). Este problema reforça a habilidade de manipulação de números reais, proporções, porcentagem e controle de precisão.

Outliers

O penúltimo problema da lista consiste em calcular a média aritmética de um conjunto de leituras, desconsiderando valores considerados como *outliers*, isto é, que estejam fora de um intervalo $[a, b]$. Este é um desafio que promove o uso de estruturas de repetição eficientes e exige o conhecimento de controle de precisão numérica.

Ângulo Entre Vetores

O último problema envolve a aplicação do produto interno entre vetores, conceito fundamental da álgebra linear, para classificar se o ângulo formado entre eles. A tarefa consiste

em calcular o produto escalar entre dois vetores e, com base no resultado, verificar se o ângulo formado é classificado como *Reto*, *Agudo* ou *Obtuso*. Esse desafio envolve o uso de estruturas de repetição e operações aritméticas sobre vetores.

3.3.3 OBI Fase 1

Para a aplicação da Fase 1 da OBI, foram disponibilizadas as datas de 10 a 14 de junho, e o planejamento considerava realizar a prova em 14/06/2024 (sexta-feira). A preparação envolvia verificar previamente o número de alunos inscritos, a fim de garantir que cada participante tivesse acesso a um computador com conexão à internet. Além disso, estava previsto comunicar a direção da escola e reservar o laboratório de informática para uso exclusivo durante o período da manhã.

O planejamento também incluía organizar os alunos em dois turnos, considerando a disponibilidade de cada participante no horário escolar. A primeira turma faria a prova das 8h às 10h e a segunda das 10h às 12h. No dia do exame, os estudantes seriam orientados a se identificar, assinar a lista de presença e receber seus logins e senhas, previamente impressos pela equipe responsável. Em seguida, cada aluno realizaria o login na plataforma oficial da OBI e iniciaria a prova, lendo os problemas, implementando as soluções e submetendo suas respostas.

Após essa etapa, estava previsto que o curso continuaria normalmente, visando preparar os possíveis classificados para a Fase 2 e, simultaneamente, manter o engajamento e o interesse dos demais alunos, mesmo daqueles que não avançassem para a fase seguinte.

3.3.4 OBI Fase 2 e CF

O planejamento para a Fase 2 seguiu a mesma lógica da Fase 1. Após a divulgação dos resultados da primeira etapa, os alunos aprovados confirmariam sua participação. A data definida para a aplicação da prova na escola seria 16/08/2024, uma sexta-feira, com reserva prévia do laboratório de informática para o período da manhã e comunicação oficial à escola sobre o evento.

A rotina prevista para o dia da prova seria semelhante à anterior: os participantes assinariam a lista de presença, receberiam seus logins e acessariam o sistema oficial para iniciar o exame. Caso houvesse classificados para a terceira etapa da OBI ou para a OBI-CF, os mesmos procedimentos seriam adotados, repetindo a logística planejada nas fases anteriores.

3.4 Entrevista de avaliação qualitativa

O presente estudo utilizou a técnica de entrevista semi-estruturada para compreender, a partir da perspectiva dos participantes, o impacto do curso preparatório para a OBI em suas trajetórias educacionais. Segundo Triviños [20], a entrevista semi-estruturada não só valoriza a presença do investigador, mas também proporciona ao entrevistado um ambiente de espontaneidade e liberdade, o que pode enriquecer os dados coletados para uma análise qualitativa dos resultados do projeto. Uma análise com riqueza de detalhes e que tenha espaço para o ponto de vista dos participantes pode proporcionar uma melhor análise do projeto, permitindo contrastar o planejamento com a execução do projeto.

Ainda segundo Triviños, uma entrevista semi-estruturada pode ser descrita como “aquela que parte de certos questionamentos básicos, apoiados em teorias e hipóteses, que interessam à pesquisa e que, em seguida, oferecem amplo campo de interrogativas, fruto de novas hipóteses que vão surgindo à medida que se recebem as respostas do informante” [20, p. 42]. Nesse sentido, seriam selecionados alguns alunos do Ensino Médio, participantes do curso preparatório para a OBI, para participar de uma entrevista semi-estruturada, com o objetivo de coletar suas percepções, sentimentos, expectativas e resultados percebidos ao longo do projeto.

A escolha por uma pesquisa qualitativa visava compreender o impacto do presente projeto em dimensões mais amplas do ambiente educacional, que extrapolassem o espaço da sala de aula, abrangendo também aspectos motivacionais e emocionais envolvidos. A entrevista semi-estruturada permite que o entrevistado possa contribuir para a análise com reflexões e percepções, sem se limitar a respostas objetivas, ainda que a dinâmica seja orientada por um roteiro prévio de perguntas.

Desta forma, foi planejada uma entrevista semi-estruturada, que seria gravada, transcrita e, posteriormente, analisadas e categorizadas, segundo propõe Bardin [21]. Esse método permite organizar e interpretar informações qualitativas por meio de categorização sistemática, identificando padrões, recorrências e significados nas falas dos participantes.

Como preparação para as entrevistas, foram inicialmente listadas 17 perguntas que abordavam diferentes dimensões da participação dos estudantes no projeto de preparação para a OBI. Dentre essas dimensões, estavam o perfil prévio do aluno, as dificuldades enfrentadas ao longo do curso, impacto da experiência, entre outros.

No entanto, visando o foco analítico e o tempo planejado para a pesquisa, as perguntas foram divididas em eixos temáticos, e desse subconjunto foram escolhidas perguntas-chave para servirem como norteadoras para a entrevista, visando proporcionar ao participante um ponto de partida para compartilhamento de experiências e de aspectos relevantes para o estudante ao longo do curso. As perguntas foram selecionadas com o objetivo de captar dados sobre a percepção dos participantes sobre aspectos relevantes ao projeto também,

como o uso da linguagem C, a metodologia e a dinâmica adotadas para o preparatório e o impacto formativo do projeto. As perguntas norteadoras são apresentadas no Quadro 9.

Quadro 9: Entrevista Semi-Estruturado

Pergunta
Qual era sua relação com programação antes do curso?
Quais foram os conceitos mais difíceis de aprender durante as aulas ministradas?
Como você se sentiu ao participar dos simulados?
Você percebeu alguma mudança na forma como pensa e resolve os problemas?
Participar do curso teve algum impacto na sua vida?

Fonte: o Autor.

Para as entrevistas, seriam selecionados alguns estudantes que participaram do curso no ano de 2024. Idealmente, dentre os selecionados, deveriam estar participantes de diferentes experiências e resultados, isto é, seria desejável convidar participantes que competiram na OBI, porém não obtiveram êxito na segunda fase, estudantes que conseguissem chegar a fases mais avançadas da competição e participantes que competissem na OBI-CF.

As entrevistas foram planejadas para acontecer entre 25 de maio e 01 de junho de 2025. Os estudantes convidados participariam de uma entrevista semi-estruturada, com o objetivo de relatarem suas experiências participando do curso de preparação e suas participações na competição da Olimpíada Brasileira de Informática. A dinâmica prevista consistia em convidar os estudantes individualmente, através das plataformas “WhatsApp”¹⁴ e “Discord”¹⁵, de acordo com a preferência do entrevistado, e usar as perguntas previamente selecionadas para ouvir os relatos dos participantes. Durante a entrevista, seriam apresentados o termo de consentimento e uma breve explicação do objetivo e da dinâmica da entrevista.

¹⁴<https://www.whatsapp.com/>

¹⁵<https://discord.com/>

Capítulo 4

Resultados

Este capítulo tem como objetivo discutir e analisar os resultados obtidos após a execução do projeto e a aplicação das provas da OBI, focando em dados qualitativos e quantitativos. Inicialmente, apresentam-se os resultados relativos ao curso idealizado, incluindo o número de encontros, a frequência dos estudantes, relação do contraste entre as listas de exercícios propostas e realizadas pelos participantes, além dos *rankings* dos simulados realizados. Em seguida, a segunda seção detalha o desempenho dos participantes nas etapas da OBI, com destaque para o número de participantes classificados para a segunda etapa, mencionando as dificuldades enfrentadas nas aplicações da prova e trazendo um panorama das questões apresentadas nas duas etapas da OBI e o conteúdo por elas requerido. Por fim, esse capítulo se encerra trazendo uma análise qualitativa do presente trabalho através dos dados coletados por meio de entrevista realizada com participantes do treinamento. Essa seção descreve o processo aplicado e discute pontos abordados pelos entrevistados como percepção de aprendizagem, organização e duração do curso.

4.1 Resultados do Treinamento

Ao final do treinamento idealizado pelo presente trabalho, foram realizados 16 encontros entre 25/04/2024 e 13/07/2024. No Quadro 10 encontra-se a relação de datas dos encontros, juntamente com os conteúdos abordados durante o treinamento e o discente responsável pela aula. Nota-se que o treinamento oferecido aos estudantes do ensino médio teve duração de aproximadamente 12 semanas, o que possibilitou um acompanhamento para participantes que competiram na primeira fase da OBI do ano de 2024, bem como para os aprovados para a segunda fase, que continuaram no treinamento até pelo menos a aplicação da prova da Fase 2 da OBI. Isso se deu porque, paralelamente às aulas realizadas na escola durante as quintas-feiras no período da manhã, após a aplicação da prova da Fase 1 da OBI, aulas adicionais foram ministradas pelo professor Edson Alves com o

intuito de reforçar os conteúdos abordados durante a semana e ensinar estratégias de resolução de problemas. Esses encontros aconteceram, em sua maioria, no período da noite e foram realizados de forma remota. Todos os participantes do curso foram convidados, mas nem todos compareceram de forma assídua às reuniões.

A turma que iniciou o treinamento em abril de 2024 contou com 32 participantes cuja a grande maioria possuía pouco ou nenhum conhecimento prévio acerca de assuntos relacionados à programação ou sobre a Olimpíada Brasileira de Informática. Para cada tópico abordado em sala de aula, foi proposto um exercício da plataforma Beecrowd¹, de modo que uma lista de exercícios foi montada para acompanhar o desempenho dos alunos e apoiar estudantes que apresentassem quaisquer dificuldades ao longo do curso. A condição para que os estudantes pudessem realizar os exercícios propostos nas listas de exercícios, apresentadas nos Quadros de 4 a 8, era o ingresso, através de um link de convite, no grupo identificado pelo nome “Preparação OBI 2024 - CED 04”. Foram inscritos ao todo 20 participantes, dos quais 12 eram alunos que estiveram presentes nas aulas do curso desde seu início.

Quadro 10: Plano Geral do Curso de Preparação para a OBI.

Aula	Data	Conteúdo	Discente
01	25/04/24	Apresentação do curso	Brunno
02	02/05/24	<code>printf()</code>	Brunno
03	09/05/24	<code>scanf()</code>	Brunno
04	11/05/24	<code>if/else</code>	Brunno
05	17/05/24	<code>while</code>	Brunno
06	25/05/24	<code>for</code>	Brunno
07	06/06/24	Exercícios de laços de repetição	Brunno
08	08/06/24	Simulado 1	Brunno
09	13/06/24	Exercícios de laços de repetição	Lucas
10	20/06/24	<i>Arrays</i>	Lucas
11	27/06/24	<i>Arrays</i>	Lucas
12	04/07/24	Exercícios de <i>Arrays</i>	Lucas
13	18/07/24	Resolução do exercício Concurso	Edson
14	24/07/24	Exercícios de <i>Arrays</i>	Brunno
15	01/07/24	<i>Strings</i>	Brunno
16	10/08/24	Simulado 2	Brunno

Fonte: o Autor.

A Tabela 4.1 apresenta a quantidade de exercícios realizados por aluno nas cinco listas propostas, compostas por 10 desafios cada. Ao todo, os 15 participantes participantes inscritos no grupo “Preparação OBI 2024” resolveram 424 exercícios, resultando em uma

¹judge.beecrowd.com/

Tabela 4.1: Números de desafios realizados pelos participantes em cada lista proposta

Aluno	Lista 1	Lista 2	Lista 3	Lista 4	Lista 5	total
A	10	3	1	2	0	16
B	10	9	8	4	1	32
C	10	7	7	8	5	37
D	10	3	2	0	0	15
E	10	9	9	6	0	34
F	10	4	10	0	0	24
G	10	10	10	7	0	37
H	10	10	10	10	0	40
I	10	10	10	7	0	37
J	10	9	8	6	1	34
K	10	10	8	0	0	28
L	10	10	0	0	0	20
M	10	10	10	6	0	36
N	1	0	0	0	0	1
O	10	10	10	3	0	33

Fonte: o Autor.

média de aproximadamente 28,3 exercícios por estudante. Observa-se uma redução no número de exercícios realizados por aluno à medida que o nível de dificuldade das listas aumentava. Esse fato é evidenciado ao perceber que a Lista 1 obteve média de 9,4 exercícios resolvidos por aluno, ao passo que a lista 5 obteve somente 0,5.

No entanto, um dos participantes resolveu somente um exercício ao longo de todo o curso. Esse comportamento configura um *outlier*, termo estatístico que designa um valor atípico, significativamente diferente do restante dos dados observados. Esse caso pode representar um aluno que iniciou o curso, mas não o acompanhou integralmente, ou que apresentou baixo engajamento nas atividades propostas. Desconsiderando esse valor atípico, os demais 14 estudantes resolveram juntos 423 exercícios, elevando a média para aproximadamente 30,2 exercícios por aluno. Esse contexto revela maior engajamento por parte da maioria da turma.

4.1.1 Resultados da Fase 1 da OBI

A aplicação da prova da Fase 1 da OBI foi realizada no dia 14/06/2024, na escola onde foi realizado o treinamento descrito no presente trabalho. A aplicação contou com 23 participantes, cuja maioria foi assídua durante o período de preparação. Como o treinamento foi realizado com estudantes do 2º e 3º ano do Ensino Médio, a fase Local da OBI contou somente com a aplicação da prova de Nível 2 da modalidade Programação.

A prova apresentou 4 desafios na primeira etapa, sendo eles “Ogro”, “Relógio”, “Concurso” e “Jogo da Vida”, cujas descrições e requisitos para resolução encontram-se a seguir².

Ogro

No desafio “Ogro” é apresentado um problema onde o participante precisa ler dois números inteiros e retornar a soma de ambos, ou o dobro da diferença entre eles, baseado em uma condição que envolve uma comparação entre os números jogados. Para que o aluno possa resolver esse problema, é necessário ter o domínio de técnicas de entrada e saída de dados, manipulando valores com operações aritméticas, assim como o conhecimento de estruturas condicionais. Todos os assuntos exigidos pelo problema foram amplamente abordados durante o curso.

Relógio

O desafio “Relógio” exige dos participantes a habilidade de manipular e converter formatos de tempo, uma vez que o problema pede para que o participante ajuste o horário inicial de um jogo, somando uma quantidade de segundos de atraso, e convertendo esse valor para o formato de horas, minutos e segundos, respectivamente. Para esse desafio, os participantes precisariam estar habituados a efetuar operações com múltiplas variáveis, dominar o uso do operador resto da divisão (representado pelo caractere % na linguagem C). Os assuntos exigidos também foram abordados ao longo do treinamento e estiveram presentes nas listas de desafios propostos na plataforma do BeeCrowd.

Concurso

O desafio da questão “Concurso” consiste em ler um conjunto de notas e o número mínimo, chamado de K , que representa o número mínimo de aprovados. Tendo esses números, o participante deve determinar a maior nota de corte, chamada de C , que permite que pelo menos K participantes sejam aprovados. Esse desafio requer o domínio de conceitos como *arrays*, laços de repetição, usados para preencher e percorrer *arrays*, e também noções sobre ordenação de *arrays*. Dos conteúdos necessários para a resolução do desafio descrito, somente a parte de ordenação de *arrays* não foi praticada em diversos exercícios, em relação aos demais tópicos, apesar de ter sido mencionada em uma das aulas remotas ministradas pelo professor Edson Alves.

²O PDF com o caderno de problemas está disponível no site oficial da OBI, no link https://olimpiada.ic.unicamp.br/static/extras/obi2024/provas/ProvaOBI2024_f1ps.pdf.

Tabela 4.2: Desempenho dos participantes na Primeira Fase da OBI

Aluno	Modalidade	Pontuação	Classificado
1	P2	200	Sim
2	P2	160	Sim
3	P2	120	Sim
4	P2	100	Sim
5	P2	100	Sim
6	P2	100	Sim
7	P2	100	Sim
8	P2	100	Sim
9	P2	100	Sim

Fonte: o Autor.

Jogo da Vida

Sendo o último desafio apresentado, a questão “Jogo da Vida” foi a mais desafiadora da primeira fase em termos de conteúdos requeridos. O problema descreve o funcionamento de um Jogo da Vida de Conway³, cuja proposta é simular alterações em grupos de seres vivos, usando um padrão de regras. Essas alterações acabam moldando a matriz dada no início a cada rodada, seguindo as regras estabelecidas no início da questão. Para resolver tal desafio, os participantes precisariam dominar conceitos como *arrays* multidimensionais, também conhecidos como matrizes, e uso aninhado de laços de repetição para leitura e impressão de matrizes. Devido ao fato de o treinamento proposto pelo presente trabalho abordar conceitos fundamentais de lógica de programação para estudantes com pouco ou nenhum conhecimento prévio em programação, não houve tempo hábil para praticar exercícios com matrizes antes da data de aplicação da prova da Fase 1 da OBI.

Ao final da prova da Etapa Local da OBI (ou Fase 1), a escola obteve 9 estudantes aprovados para a segunda etapa. Os competidores que avançaram para a etapa seguinte obtiveram suas pontuações (disponibilizadas na Tabela 4.2) resolvendo questões completas e também pontuando com as subtarefas propostas pela prova onde, em algumas questões, foi possível obter entre 30% e 70% da pontuação total da questão, resolvendo parcialmente o desafio proposto. Ao final da prova, os estudantes que obtiveram, em média, uma questão e meia resolvida puderam avançar para a etapa seguinte.

4.1.2 Resultados da Fase 2 da OBI

A Fase 2 da OBI foi aplicada no dia 16/08/2024 e contou com a participação de 9 candidatos. A aplicação seguiu os mesmos procedimentos da etapa anterior, na qual os alunos

³https://pt.wikipedia.org/wiki/Jogo_da_vida

compareceram à mesma escola onde foi ministrado o treinamento descrito neste trabalho. Os estudantes se posicionaram, cada um em uma máquina, efetuaram o login no sistema da OBI e dispuseram de duas horas para resolver e submeter os desafios propostos pela prova. Como esperado, a prova da Fase 2 da OBI trouxe desafios de maior nível de dificuldade, cujas descrições encontram-se apresentadas a seguir⁴.

Cubo Preto

No desafio “Cubo Preto” é apresentado um problema que envolve uma análise tridimensional de um cubo pintado de preto. O participante deve ler um valor N e, a partir desse valor, determinar quantos cubinhos de dimensão $1 \times 1 \times 1$ terão 3, 2 ou 1 face pintada, após o cubo original de lado N ter sido cortado. Nessa questão, os participantes precisam entender bem o conceito de iteração e ter noções de contagem. Apesar do conteúdo sobre abstração tridimensional não ter sido diretamente abordado ao longo do curso, o problema poderia ser resolvido com boas noções de lógica e contagem.

Alfabeto Alienígena

Neste desafio, o objetivo é verificar se uma mensagem foi escrita com os caracteres válidos ou não. No problema, os alunos precisam ler dois números inteiros K e N , que correspondem ao tamanho da string de caracteres válidos e ao tamanho da mensagem digitada, respectivamente. Em seguida, os participantes precisam ler duas strings. Ao final deve ser retornado “S”, caso a string use somente caracteres válidos ou “N”, caso haja caracteres inválidos na segunda string lida. Para esse desafio, os alunos precisariam ter boas noções sobre laços de repetição, strings e vetores. Aulas complementares sobre *Arrays* e *Strings* foram ministradas de forma remota durante e após o treinamento da escola, porém foi um tópico um pouco praticado pelos estudantes, como se pode notar no placar de exercícios apresentado na Tabela 4.1.

Concatena Dígitos

A questão “Concatena Dígitos” exige um conhecimento mais avançado de programação, uma vez que demanda que os alunos leiam dois números inteiros N e Q referentes ao número de dígitos em uma lista e, em seguida, o número de intervalos a serem consultados dentro dessa lista contendo N elementos. O grande desafio consiste em, sabendo o intervalo de possíveis pares, retornar o somatório de todos os pares possíveis dentro deste

⁴O caderno de Nível 2 da Fase 2 da OBI de 2024 pode ser visualizado no site oficial da OBI no endereço https://olimpiada.ic.unicamp.br/static/extras/obi2024/provas/ProvaOBI2024_f2p2.pdf.

intervalo. Apesar de terem sido abordados os conceitos de *arrays* e laços de repetição, nenhum exercício de tal grau de dificuldade foi apresentado aos alunos anteriormente.

Jogo do Poder

No último desafio apresentado na prova, que simula um jogo em uma matriz onde o herói deve acumular poder e derrotar monstros, os alunos precisariam ter boas noções de *arrays* bidimensionais (matrizes) para calcular o maior número de poder que pode ser atingido pelo herói, considerando sua posição inicial em uma célula da matriz e que as jogadas seriam feitas de forma ótima. Além do tópico não abordado (matrizes), esse problema traz restrições de tempo de execução e demanda o conhecimento de travessias e de componentes conectados de um grafo, conceitos aos quais os alunos não foram previamente apresentados.

Ao final da aplicação da segunda fase da OBI, dois dos nove participantes conseguiram ter uma questão aprovada com 100% de acerto. Porém, ao final da prova, ocorreu um problema técnico na escola na qual a prova foi aplicada e também houve uma instabilidade geral no sistema de provas da organização. Esse conjunto de fatores levou a uma segunda aplicação da Fase 2 nacionalmente. No site da OBI, esse fato fica evidente na disponibilização das provas do ano de 2024. Na página “Fase 2 – Turno B – Modalidade Programação”⁵ é possível encontrar o caderno de questões que foi disponibilizado para a segunda aplicação da prova da Fase 2 da competição. As descrições das questões estão apresentadas a seguir.

Game Show

Neste desafio, os alunos precisam simular um jogo onde se entra em salas numeradas e, a partir de uma sequência de direções (esquerda ou direita), encontrar a sala final, que corresponde a operações feitas, orientadas pelas escolhas de direção. Para este problema, os alunos precisariam ter o conhecimento de variáveis, condicionais, laços de repetição e leitura de strings. Todos os tópicos foram devidamente abordados ao longo do treinamento.

Salada de Frutas

Na questão “Salada de Frutas” os participantes devem calcular o número máximo de tipos diferentes de frutas que se pode comprar com uma quantidade fixa de dinheiro e uma quantidade de pares de fruta disponível do formato tipo $\times$ preço. Esse problema

⁵<https://olimpiada.ic.unicamp.br/passadas/OBI2024/fase2b/programacao/cadernos/>

envolve noções de *arrays*, laços de repetição, estratégias de algoritmos gulosos e ordenação. Apesar de terem sido citados em momentos do treinamentos, os tópicos mais avançados não foram trabalhados em um grande volume de exercícios.

Trio de Palitinhos

Este desafio aborda um problema conhecido de geometria que envolve a condição de formação de um triângulo. No desafio os estudantes precisam ler um *array* de tamanho N e identificar, com base nos elementos do *array*, quantos trios de palitos podem formar um triângulo, respeitando a desigualdade triangular. Para esse desafio, é necessário saber manipular *arrays*, laços aninhados e comparação de elementos de *arrays*. Os conteúdos necessários foram trabalhados em sala de aula ao decorrer do treinamento.

Remove Dígitos

O último problema da Fase 2 é um desafio que simula um jogo onde o participante remove um dígito de um número a cada rodada. O objetivo do jogo é eliminar todos os dígitos com o menor número de jogadas possível. Um detalhe importante é que somente se pode subtrair um dígito presente no número atual. Nesse problema, os alunos precisariam ter noções de *Arrays* e extração de dígitos. Nem todos os assuntos foram abordados pelo treinamento.

Ao final da segunda aplicação da Fase 2 da OBI, nenhum dos 9 participantes conseguiu ser aprovado para a Fase 3 (Etapa Nacional) da competição. Apesar de não obter êxito na segunda etapa, esse foi o melhor desempenho registrado na escola, que nunca havia registrado aprovações para a Fase 2. Os alunos se mostraram motivados ao final de suas participações na OBI, tendo ainda a modalidade feminina a ser aplicada na escola, o que permitiria uma outra chance para as participantes que fizeram parte do treinamento. Na Tabela 4.3, é possível conferir o desempenho dos alunos que foram aprovados para a Fase 2.

4.1.3 Resultados da CF - OBI

A CF-OBI ocorreu no dia 13 de setembro e contou com 4 participantes. Diferentemente das outras modalidades, essa conta com apenas uma etapa e a prova trouxe 5 problemas que estão descritos a seguir:

Tabela 4.3: Desempenho dos participantes na Segunda Fase da OBI

Aluno	Modalidade	Pontuação	Classificado
1	P2	100	Não
2	P2	0	Não
3	P2	118	Não
4	P2	33	Não
5	P2	0	Não
6	P2	0	Não
7	P2	0	Não
8	P2	100	Não
9	P2	60	Não

Fonte: o Autor.

Bibi e a Árvore

Este problema propõe a previsão do crescimento de uma planta a partir de dados observados nos primeiros meses de vida. A tarefa consiste em determinar a altura da árvore em um mês X , considerando que, a partir do sexto mês, o crescimento se torna constante e igual à diferença entre a altura do sexto e do quinto mês. Para este desafio, o participante deve lidar com leitura e saída de dados, condicionais e aritmética básica. Além disso, o problema requer conhecimento de tipos de dados, uma vez que trabalha com entrada e armazenamento de números inteiros muito grandes.

Mistura de Poções

Neste desafio, os estudantes precisam identificar quantas formas diferentes existem de remover poções das extremidades de uma lista, mantendo um segmento contíguo com pelo menos K tipos diferentes de poções para criar um feitiço. Para esse desafio, os participantes necessitariam ter conhecimentos sobre vetores, estruturas de repetição e manipulação de índices.

Tabuleiro

Este desafio pede para que o participante calcule o número de maneiras de preencher um tabuleiro com dimensões $3 \times N$. Para resolver este problema, as participantes precisariam dominar *arrays bidimensionais*, estruturas de repetição aninhadas e noções sobre programação dinâmica. Além de estruturas de repetição, nenhum tópico foi trabalhado em sala de aula.

Christina e os bombons

Neste problema de otimização em grafo, Christina precisa encontrar o caminho de A até B carregando o máximo de bombons possível, considerando que o tempo de travessia depende do peso carregado e da dificuldade do terreno. Algumas cidades possuem monstros e devem ser evitadas. Para resolver este desafio, as participantes precisariam saber utilizar *arrays* bidimensionais (matrizes) e algoritmos de busca em grafos. Nenhum dos tópicos foi trabalhado em sala de aula.

Estradas em Nlogônia

O último desafio é um problema complexo de estruturas de dados que simula a construção incremental de estradas entre cidades, com a restrição de que após conectar duas cidades, uma delas não pode mais receber novas conexões. É necessário responder consultas sobre a soma das distâncias entre todos os pares de cidades em uma componente. Este foi o desafio mais complexo da prova, e as alunas não tiveram contato com conteúdos de estruturas de dados e grafos, tópicos necessários para resolver a questão.

Apesar da participação, nenhuma das estudantes inscritas na CF-OBİ conseguiu ser premiada, ou obter uma menção honrosa. Notou-se que a exigência dos conteúdos da prova da CF-OBİ extrapolou os conteúdos trabalhados durante o treinamento realizado na escola, o que pode ter impactado no baixo desempenho por parte das competidoras. Ainda assim, o engajamento das alunas e a disposição de participarem da modalidade, mostraram que houve impacto positivo no fomento ao interesse pela área de Tecnologia nas alunas presentes no treinamento.

4.2 Análise Qualitativa do Treinamento

Durante o período de 24/05/2025 a 01/06/2025, cerca de 10 meses após o último encontro do treinamento, foram entrevistados 8 alunos que participaram do treinamento para a Olimpíada Brasileira de Informática, tema do presente trabalho, com o intuito de coletar informações para análise qualitativa do treinamento proposto.

As 8 entrevistas foram realizadas de forma remota, usando as plataformas “Discord”⁶ e “WhatsApp”⁷, de acordo com a preferência de cada participante. As entrevistas tiveram uma média de 15 minutos de duração e todas começaram com a apresentação do termo de consentimento, enviado posteriormente assinado.

⁶<https://discord.com/>

⁷<https://www.whatsapp.com/>

Ao total, 9 estudantes foram convidados para participar de uma entrevista semi-estruturada, dos quais 8 aceitaram. A entrevista teve como objetivo captar a percepção dos participantes sobre o curso, saber se algum deles havia algum contato prévio com programação e entender se os tópicos abordados em sala prepararam os estudantes para a aplicação das provas oficiais da OBI. Para fins de análise do treinamento, os alunos serão aqui identificados como A, B, C, D, E, F, G e H.

Nas entrevistas, buscou-se conhecer as experiências prévias em programação com o objetivo de compreender quais dos participantes já teriam se engajado no aprendizado de lógica de programação. Dentre todos os entrevistados, o estudante A foi o participante que mais se aproximou de uma experiência prévia ao treinamento. Segundo o entrevistado, houve um primeiro contato com programação na escola em uma disciplina eletiva, onde era necessário usar C++ para escrever o código que, após compilado e exportado, seria executado em uma placa Arduino. Além de uma breve experiência com C++, houve o contato com a linguagem Python, segundo o entrevistado, por curiosidade. O estudante relatou que procurou na internet conceitos iniciais da linguagem. Os estudantes D e H também mencionaram que em momentos anteriores ao curso, houve uma tentativa de aprender programação - o Estudante D por meio de um familiar que tentou apresentar conceitos de programação quando o estudante ainda era mais novo, e o Estudante H por curiosidade de saber como funcionavam os jogos por trás da parte gráfica. Apesar do breve contato, nenhum dos dois chegou a praticar minimamente qualquer linguagem de programação.

Sobre o tema “duração”, os participantes foram questionados sobre suas percepções da duração de aula e do número de encontros realizados, com o objetivo de entender se, para eles, o tempo disponibilizado na escola foi suficiente para a apresentação dos conteúdos. Os entrevistados C e G foram categóricos em mencionar que a duração do curso seria uma sugestão de melhoria. No entendimento de ambos, apesar de o conteúdo disponibilizado ao longo do curso ter servido como base para o que foi cobrado na OBI, nas duas etapas nas quais a escola participou, o ideal seria que houvesse mais horas de aula. Um dos estudantes, inclusive, afirmou: “E, se eu pudesse[sugerir uma alteração na duração do curso], era o ano inteiro”. As aulas ministradas na escola tiveram duração de cerca de 5 horas diárias, sendo elas divididas em 3 turmas diferentes, com 100 minutos cada aula, seguidas de um intervalo de 15 minutos. Esses encontros aconteceram, em sua maioria, em quintas-feiras, começando às 7:15 da manhã e terminando por volta de 12:15. Adicionalmente, e em alguns sábados houve encontros no laboratório de informática, para aulas extras de exercícios e simulados previamente planejados, nestes casos, a duração foi sempre de 8 da manhã ao meio do dia. A relação das datas, conteúdos e responsáveis pelos encontros está descrita no Quadro 10. Nota-se que entre o primeiro encontro com

os alunos e a aplicação da primeira fase da OBI, houve um total de 9 aulas ministradas em um espaço de pouco mais de um mês. Entende-se que a percepção dos alunos pode ser de que, para a primeira prova, a realização de mais aulas os teria preparado melhor.

Outro aspecto explorado com os participantes foi a experiência prévia com olimpíadas de conhecimento. Um dos objetivos do curso foi apresentar o cenário de tecnologia e incentivar a participação dos estudantes na Olimpíada Brasileira de Informática – OBI. Neste contexto, os participantes foram questionados sobre seus conhecimentos e vivências anteriores com esse tipo de competição. Notou-se um baixo nível de familiaridade com a OBI, entretanto, os alunos A, B, C, D, F, G e H relataram ter participado ou estudado para a Olimpíada Brasileira de Matemática das Escolas Públicas – OBMEP. A exposição a competições como a OBMEP pode estar ligada ao desenvolvimento de habilidades e interesses por desafios que envolvem lógica de programação e resolução de problemas, o que possivelmente contribuiu para a adesão e o engajamento dos alunos durante o curso oferecido.

Em relação à linguagem C e ao conteúdo de lógica de programação, os alunos foram perguntados sobre conteúdos que, eventualmente, pudessem ter sido motivo de dúvida ou dificuldades. Os estudantes C, F e H afirmaram que o conteúdo referente aos laços de repetição usando o comando `while` foi um dos obstáculos do curso. Segundo palavras de uma das pessoas entrevistadas, estudante C: “eu fiquei mais em dúvida, acho que foi só no `while` e no `do-while`, mas depois vocês apresentaram o `for`, que aí facilitou mais”. O participante A relatou que teve certa dificuldade inicialmente com o conceito de laços de repetição, entretanto, frisou que a dificuldade se dava ao uso de laços aninhados, conteúdo que foi abordado no ensino de vetores e em resoluções de exercícios nas aulas ministradas pelo professor Edson.

Em relação aos simulados aplicados ao longo do semestre, todos os entrevistados afirmaram que a estratégia serviu como boa preparação para o dia da prova. O intuito da aplicação dos simulados era não somente preparar os estudantes para a OBI, mas também apresentá-los ao ambiente de maratonas competitivas de programação. Sobre o tema, o entrevistado D mencionou que foi muito produtivo trabalhar em grupo em um dos simulados do qual participou. Nas palavras do estudante C “ Eu achei interessante[participar do simulado em grupos], porque assim, mostra que a outra pessoa tem formas diferentes de resolver a questão, da uma forma que [eu] responderia. Sim, era assim que funcionava. Eu via a questão de uma forma e o [colega de time] via de outra. Ele via de outra forma como que ele queria fazer o código eu via que das duas formas davam certo.”, mencionando em seguida que programar com outras pessoas influenciou a forma de pensar em resolução de problemas.

Dentre os entrevistados, estavam 2 participantes do sexo feminino. As duas entrevistadas relataram que a competição feminina foi uma experiência interessante, mesmo tendo somente uma etapa. Um dos pontos levantados na entrevista por uma das participantes, foi que na prova da Competição Feminina da OBI – CF-OBI⁸, houve questões das quais as participantes não se sentiram prontas, por não terem praticado suficientemente ou por não terem visto o conteúdo de algumas questões, entretanto, segundo relato de uma delas, a CF-OBI foi um grande incentivo para o ingresso de meninas no contexto de TI, que ainda é um ambiente amplamente ocupado por pessoas do sexo masculino.

Por fim, foi perguntado sobre o impacto do curso nas vidas dos participantes. De 8 estudantes que participaram da entrevista, 7 afirmaram que o projeto descrito no presente trabalho teve impacto positivo na escolha de uma área profissional a ser seguida. Do total de 8 entrevistados, 5 estão matriculados em Instituição Superior em cursos de tecnologia, 1 está estudando para vestibulares de cursos de tecnologia e 1 está cursando o 3º ano do ensino médio, porém pensa em se aventurar na área de tecnologia um dia. Também vale destacar que 1 entrevistado optou por não cursar o ensino superior no momento, mas está trabalhando na área de redes de computadores.

4.2.1 Limitações do Estudo

Apesar dos resultados positivos observados com a implementação do curso preparatório para a Olimpíada Brasileira de Informática (OBI), este estudo apresenta limitações relevantes que devem ser consideradas para a interpretação dos dados e para o planejamento de futuras iniciativas semelhantes.

A principal limitação encontrada foi o **tempo reduzido** disponível para o desenvolvimento do curso. O treinamento teve início a menos de dois meses da aplicação da prova da primeira fase da OBI, o que impediu a abordagem de conteúdos mais avançados, como grafos, ordenação e análise de algoritmos. Essa restrição temporal exigiu uma priorização dos conteúdos mais fundamentais, como estruturas condicionais, laços de repetição e *arrays*. Outro fator importante a se levar em conta é a idade e maturidade dos alunos. O fato de estarem, em sua maioria, no terceiro ano do ensino médio – e também inscritos em uma disciplina que tinha como objetivo a preparação para olimpíadas de matemática– permitiu que os assuntos mais introdutórios fossem assimilados com certa facilidade, considerando o intervalo de tempo no qual as aulas foram ministradas.

Além disso, as aulas do curso preparatório foram ministradas de forma **concomitante às demais disciplinas da grade curricular dos estudantes**, o que impôs desafios à assiduidade e à dedicação contínua dos participantes. Em alguns casos, houve conflito

⁸https://olimpiada.ic.unicamp.br/info/competicao_feminina/

com demandas e outras atividades escolares, o que dificultou a participação plena dos estudantes em todas as etapas do curso.

Esses fatores evidenciam a necessidade de maior institucionalização do ensino de computação na educação básica. Projetos extracurriculares, embora valiosos, enfrentam barreiras estruturais que poderiam ser mitigadas com a incorporação da computação como componente curricular obrigatório, com tempo pedagógico adequado e infraestrutura mínima garantida.

Capítulo 5

Considerações Finais

O presente trabalho se propôs a investigar a viabilidade, o processo e o impacto de um treinamento preparatório para a Olimpíada Brasileira de Informática – OBI, realizado com estudantes do Ensino Médio de uma escola pública do Distrito Federal. Tendo como norte a metodologia de Aprendizagem Baseada em Problemas (PBL) e o uso de ferramentas de programação competitiva, o objetivo central foi não somente capacitar os estudantes para participar da competição, mas também promover o desenvolvimento das habilidades de decomposição de problemas, reconhecimento de padrões, abstração e criação de algoritmos, habilidades que são descritas pelo pensamento computacional. Além disso, o trabalho teve como objetivo analisar os efeitos da iniciativa nas trajetórias educacionais e pessoais dos participantes. Ao final desta jornada, é possível tecer as seguintes conclusões.

Primeiramente, o estudo demonstrou que a iniciativa foi viável e bem-sucedida, considerando os fatores observados de critério de seleção de participantes e contexto educacional da escola. Esses fatos ficaram evidentes pelo resultado quantitativo mais notável: a classificação inédita de nove estudantes da escola para a Fase 2 da OBI, um marco que, por si só, valida a eficácia da abordagem. Contudo, os impactos do projeto transcendem essa métrica. A análise qualitativa, realizada por meio de entrevistas com participantes, revelou que o maior ganho foi a promoção de uma aprendizagem significativa e o despertar do interesse pela área de tecnologia. Os estudantes, antes com pouco ou nenhum contato com programação, não apenas aprenderam conceitos fundamentais de lógica de programação através da linguagem C, mas o fizeram de forma engajadora e contextualizada, o que corrobora que há uma sinergia entre a metodologia da Aprendizagem Baseada em Problemas com o ensino de computação.

Em segundo lugar, pode-se concluir que a estratégia pedagógica adotada foi um dos fatores-chave para o engajamento dos estudantes. O treinamento teve como objetivo apresentar uma abordagem prática, o que foi combinado com os alunos desde o momento

inicial do curso. Além disso, a utilização da plataforma BeeCrowd¹, que conta com um sistema de *feedback* instantâneo, e a realização de simulados (Mini-OBIs) foram apontadas pelos participantes como experiências positivas ao longo do curso. Essas experiências promoveram a prática de competências socioemocionais como a resiliência ao lidar com erros, a colaboração e a gestão de tempo, como foi evidenciado nos relatos dos alunos ao sobre a resolução de problemas durante os simulados realizados. Neste contexto, o projeto proporcionou aos estudantes a oportunidade de praticar habilidades que podem ser úteis a eles em quesitos que vão além da olimpíada.

Adicionalmente, o trabalho teve um impacto decisivo na tomada de decisão acerca da trajetória educacional dos participantes. Para 6 dentre os 8 entrevistados, a participação no curso foi um fator determinante na escolha de perseguir uma formação e, posteriormente, o ingresso em uma carreira na área de Tecnologia da Informação. Em um contexto onde estudantes de escola pública lidam, muitas vezes, com um acesso limitado a experiências na área de Tecnologia, o curso pôde atuar como uma janela de oportunidades, mostrando aos alunos que, com estudo e dedicação, é possível expandir os horizontes e alcançar objetivos ambiciosos. A iniciativa reforça, assim, o papel do ensino de computação na emancipação do sujeito.

Entretanto, algumas dificuldades persistem e merecem atenção. A principal delas é que a computação ainda não está integrada à educação básica com a mesma prioridade das demais disciplinas, o que dificulta a continuidade e a sistematização do ensino. Essa limitação fica evidente na ausência de professores de computação ministrando disciplinas de computação em muitas escolas públicas. Além disso, o curso foi ministrado em espaço extracurricular, paralelo às demais matérias, o que reduziu a carga horária disponível e dificultou o aprofundamento de conteúdos mais avançados exigidos nas etapas seguintes da competição.

Apesar dessas limitações, o trabalho mostrou que, mesmo em um cenário de escassez de tempo, e com a falta de um currículo formal para computação, é possível implementar ações educativas eficazes. A partir dos resultados obtidos, recomenda-se a replicação e ampliação deste modelo de ensino, incluindo o planejamento de ações de longo prazo, formação continuada de professores e a articulação com políticas públicas que visem à inclusão da computação como área estruturante do currículo escolar. Somado a isso, observa-se que a formalização de parcerias entre universidades públicas e escolas da rede pública de ensino pode ser benéfica para todos os agentes envolvidos no cenário da educação. Essas parcerias podem assumir o formato de projetos de extensão, estágios supervisionados ou programas de iniciação à docência, promovendo a presença e a participação constante dos licenciandos e pesquisadores da área no ambiente escolar. Por meio de parcerias como

¹<https://judge.beecrowd.com>

essas, há a oportunidade de oferecer o pensamento computacional de forma continuada para os estudantes, fortalecendo, assim, a formação inicial de professores.

Este trabalho deixa caminhos abertos para futuras investigações e ações. Como sugestões técnicas para trabalhos futuros, recomenda-se:

- A realização de um estudo longitudinal para acompanhar a trajetória acadêmica dos participantes, a fim de mensurar o impacto no longo prazo da iniciativa;
- A implementação do projeto em um formato de curso de extensão anual, permitindo uma cobertura mais ampla dos conteúdos presentes no currículo da OBI, bem como a preparação de alunos dos anos iniciais do Ensino Médio, possibilitando uma capacitação contínua;
- A adoção da linguagem de programação C++, que tende a exigir menor tempo de adaptação por parte dos estudantes, especialmente aqueles com alguma familiaridade prévia com lógica estruturada. Além de dispor de uma biblioteca padrão robusta, a linguagem oferece estruturas consolidadas com métodos eficientes que podem otimizar o desenvolvimento das soluções. Entre eles, destacam-se funções como `sort()` para ordenação e métodos da classe `std::string`, como `find()` e `rfind()`, que favorecem a clareza do código e contribuem para o melhor aproveitamento do tempo disponível durante o treinamento.

Cumprе destacar, ainda, que os apêndices de 1A.1 a 7A.7 reúnem planos de aula que constituem um caderno técnico, elaborado como sugestão de sequência pedagógica para o ensino de lógica de programação no ensino médio. Esse material oferece um norte para a implementação do ensino sistemático de computação nas escolas públicas, servindo tanto como apoio para professores em formação quanto como recurso para futuras iniciativas de extensão ou inserção curricular. Dessa forma, o caderno técnico se apresenta como um produto educacional completo que pode ser adaptado e replicado em diferentes contextos escolares.

Por fim, espera-se que este estudo possa contribuir para a discussão sobre o papel da computação na escola pública, destacando a necessidade de sua inserção como disciplina regular e a importância de oferecer aos estudantes oportunidades concretas de desenvolver competências digitais e computacionais que serão decisivas em seu futuro acadêmico e profissional.

Referências Bibliográficas

- [1] Wing, Jeannette M: *Computational thinking*. Communications of the ACM, 49(3):33–35, 2006.
- [2] Campagnolo, Julio Cesar Neves: *O caráter incentivador das olimpíadas de conhecimento: uma análise sobre a visão dos alunos da olimpíada brasileira de astronomia e astronáutica sobre a olimpíada*. Revista Areté| Revista Amazônica de Ensino de Ciências, 11:31–41, 2011.
- [3] Brasscom: *Relatório Perspectivas do Mercado de Trabalho do Macrosetor de TIC*, 2025. <https://brasscom.org.br/pdfs/relatorio-perspectivas-do-mercado-de-trabalho-do-macrosetor-de-tic/>, Acesso em: 10 mar. 2025.
- [4] Google Cloud: *90 casos de IA na América Latina que estão moldando o futuro da inovação*, 2025. <http://bit.ly/457Zdn7>, Acesso em: 10 mar. 2025.
- [5] TAMAYO, Jorge, Leila DOUMI, Sagar GOEL, Orsolya KOVÁCS-ONDREJKOVIC e Raffaella SADUN: *Reskilling in the age of AI: five new paradigms for leaders—and employees*. Harvard Business Review. Disponível em: <https://hbr.org/2023/09/reskilling-in-the-age-of-ai>, set. 2023. Acesso em: 7 jul. 2025.
- [6] Ministério da Educação: *Complemento à base nacional comum curricular (bncc) de computação na educação básica*. Documento oficial do Ministério da Educação, Brasil, 2022. <http://portal.mec.gov.br/docman/fevereiro-2022-pdf/236791-anexo-ao-parecer-cneceb-n-2-2022-bncc-computacao/file>, Acessado em: 10 de março de 2025.
- [7] Masterson, Victoria: *Future of jobs 2023: These are the most in-demand skills now - and beyond*. World Economic Forum, May 2023. <https://www.weforum.org/stories/2023/05/future-of-jobs-2023-skills/>, Acessado em: 10 de março de 2025.
- [8] World Economic Forum: *Future of jobs report 2025*. Relatório Técnico, World Economic Forum, 2025. https://reports.weforum.org/docs/WEF_Future_of_Jobs_Report_2025.pdf, Acessado em: 10 de março de 2025.
- [9] Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq): *Olimpíadas científicas e suas contribuições para o ensino e a alfabetização científica*, 2024. <http://bit.ly/46xBT40>, Acessado em: 10 de março de 2025.

- [10] Secretaria de Estado de Educação do Distrito Federal: *Calendário das Olimpíadas Científicas 2025*, 2025. <https://www.educacao.df.gov.br/calendario-das-olimpiadas-cientificas-2025/>, Acesso em: 10 mar. 2025.
- [11] Universidade de São Paulo: *Edital n.º 1/2025 - Competições do Conhecimento 2025*, 2025. <https://doe.sp.gov.br/executivo/universidade-de-sao-paulo/edital-n-1-2025-competicoes-do-conhecimento-2025-2025010813411426811904>, Acesso em: 10 mar. 2025.
- [12] Instituto de Computação da Unicamp: *Ementa de Programação - Olimpíada Brasileira de Informática*, 2025. https://olimpiada.ic.unicamp.br/prepare/ementas/ementa_prog/, Acesso em: 10 mar. 2025.
- [13] WALTON, H. J. e M. B. MATTHEWS: *Essentials of problem-based learning*. Medical Education, 23(6):542–558, 1989. <https://asmepublications.onlinelibrary.wiley.com/doi/abs/10.1111/j.1365-2923.1989.tb01581.x>.
- [14] Rossi, Leila Lisiane, Dyeizon Procopiuk da Silva, Kennedy Ferreira Araujo, Fábio José Rodrigues Pinheiro, Wagner Carlos Mariani e Angela Maria Crotti Rosa: *Olimpíada brasileira de informática – obi 2020*. Extensão Tecnológica: Revista de Extensão do Instituto Federal Catarinense, 8(16):141–147, dez. 2021. <https://publicacoes.ifc.edu.br/index.php/RevExt/article/view/2121>.
- [15] Ramos, Thaís, Leonardo Vidal Batista, José Alves Monteiro Neto, Aellison Santos, Karla Machado e Paulo Branco: *Ensino de programação para olimpíada brasileira de informática*. Em *Anais do Workshop de Informática na Escola*, volume 21, páginas 122–126, 2015.
- [16] Gonçalves Fernandes, Flávia: *A importância de treinamentos para olimpíadas brasileiras de informática*. Revista Ciência em Extensão, 17, 2021.
- [17] Santos, Erika Raquel Silva dos, Fábio Cristiano e Ivaldo Barbosa Da Mota Neto: *Raciocínio lógico e computação: Descobrendo estratégias de ensino por meio da olimpíada brasileira de informática*. Em *Anais do Workshop de Informática na Escola*, volume 21, páginas 266–270, 2015.
- [18] Lopes, Alexandre Ferreira, TS Santana e AB Braga: *Uma análise da influência da aplicação de curso preparatório de olimpíadas científicas de informática com estudantes de ensino médio*. Fórum Goiano de Software Livre, 2018.
- [19] Moreira, Thiago Rafael Félix: *Proposta de ensino de programação para estudantes do ensino médio visando a participação na olimpíada brasileira*

de informática, 2019. https://bdm.unb.br/bitstream/10483/23051/1/2019_ThiagoRafaelFelixMoreira_tcc.pdf, Acesso em: 5 ago. 2025.

- [20] Triviños, Antonio Carlos Gil: *Introdução à pesquisa em ciências sociais: a pesquisa qualitativa em educação*. Atlas, São Paulo, 1987.
- [21] Bardin, Laurence: *Análise de Conteúdo*. Edições 70, São Paulo, 2011.

Apêndice A

Planos de Ensino

A.1 Plano 1 - Impressão no terminal

A.1.1 Princípios Educacionais

Para esta aula, planejamos utilizar uma abordagem construtivista, onde o aprendizado é ativo e centrado no aluno, usando também a visão de aprendizagem baseada em problemas – PBL [1], [2]. Segundo Piaget e Vygotsky, o conhecimento é construído à medida que os alunos interagem com o ambiente e entre si. Portanto, a aula será prática e interativa [3], com ênfase em atividades que permitam aos alunos experimentar e resolver problemas reais, como o uso da função `printf()` em C e a familiarização com a plataforma BeeCrowd¹.

Competência Específica de Computação (BNCC)

“**Competência 6** – Expressar e compartilhar informações, ideias, sentimentos e soluções computacionais utilizando diferentes plataformas, ferramentas, linguagens e tecnologias da Computação de forma fluente, criativa, crítica, significativa, reflexiva e ética.”

A.1.2 Objetivos Educacionais

Objetivos de ensino

- (a) Preparar os alunos para compreender o uso e a importância da função `printf()` na linguagem C [4], estabelecendo as bases para a formatação de saídas em programas computacionais;

¹<https://judge.beecrowd.com/pt/>

- (b) desenvolver a percepção dos tipos de dados em C e como eles são armazenados em variáveis, integrando esses conceitos com a prática de programação;
- (c) proporcionar uma introdução ao ambiente da plataforma BeeCrowd, promovendo o registro e a submissão de soluções de problemas para avaliação automatizada;
- (d) familiarizar os alunos com o contexto de competições como a Olimpíada Brasileira de Informática², enfatizando a importância da prática contínua em programação; e
- (e) promover a colaboração entre os alunos, criando um ambiente de aprendizado ativo, com o uso de problemas práticos e interativos para fortalecer o desenvolvimento de habilidades computacionais.

Objetivos de aprendizagem

- (a) Aplicar a função `printf()` em C para exibir saídas formatadas, utilizando corretamente os especificadores de formato adequados para cada tipo de dado;
- (b) Identificar e distinguir diferentes tipos de dados em C (como `int`, `float`, `char`), compreendendo como armazená-los e manipulá-los em variáveis;
- (c) Desenvolver habilidades práticas na formatação de saídas de texto, garantindo que o programa apresente informações claras e organizadas ao usuário;
- (d) Navegar pela plataforma BeeCrowd, realizando o cadastro, submissão de problemas e leitura do *feedback* do compilador, com confiança e autonomia;
- (e) Reconhecer o valor de competições de programação como uma ferramenta de aprendizagem, participando ativamente e praticando a resolução de problemas.

A.1.3 Conceitos a serem desenvolvidos

- (i) Introdução à plataforma BeeCrowd: cadastro, submissão de problemas e leitura de *feedback* do compilador;
- (ii) Armazenamentos de dados em variáveis;
- (iii) Tipos de dados;
- (iv) Função `printf()` em C: sintaxe e uso básico;
- (v) Uso de especificadores de formato (`%d`, `%s`, `%f`, etc.) na função `printf()`;
- (vi) Prática de resolução de problemas simples no BeeCrowd.

²<http://olimpiada.ic.unicamp.br/>

A.1.4 Público-Alvo

Alunos do segundo e terceiro ano do Ensino Médio do CED 04 de Taguatinga, inscritos no curso de preparação para a OBI. Para essa primeira aula prática, os alunos não precisam ter nenhum conhecimento prévio sobre programação.

A.1.5 Tema ou situação de contexto (organizadores prévios)

A primeira impressão é a que fica: a importância da saída de dados formatada em C. Como imprimir mensagens, valores e resultados de um programa na tela, e como submeter o primeiro programa na plataforma do BeeCrowd.

Contexto

Os alunos serão apresentados a cenários onde a saída de dados precisa ser bem formatada, como em relatórios, variáveis com valores pré-definidos ou interfaces de usuário, para mostrar a importância de entender e usar a função `printf()` corretamente.

A.1.6 Níveis Cognitivos

Os níveis cognitivos (básico, intermediário e avançado) e os respectivos tópicos a serem trabalhados estão descritos na Tabela 11.

A.1.7 Descrição da Atividade

As atividades a serem desenvolvidas estão descritas no Quadro 12.

A.1.8 Recursos Materiais e Tecnológicos

- (i) Computadores com acesso à internet;
- (ii) Conta na plataforma BeeCrowd;
- (iii) Editor de código (preferencialmente IDE para C, como Code::Blocks³ ou VSCode⁴, ou compilador online);
- (iv) Projetor para demonstração.

³<https://www.codeblocks.org/>

⁴<https://code.visualstudio.com/>

Quadro 11: Níveis cognitivos e tópicos a serem trabalhados

Nível	Tópicos
Básico	★ Introdução à função <code>printf()</code> e seus especificadores de formato. ★ Introdução ao conceito de variáveis. ★ Exercícios de formatação simples com <code>printf()</code>.
Intermediário	★ Cadastro e navegação na plataforma BeeCrowd. ★ Armazenamento de valores e tipos de valores em variáveis em C. ★ Submissão de problemas básicos e interpretação de <i>feedback</i>.
Avançado	★ Cadastro e navegação na plataforma BeeCrowd. ★ Discussão sobre a importância da formatação de saída em programas mais complexos. ★ Reflexão: tipos de variáveis podem guardar determinados valores. ★ Análise de problemas de formatação em competições anteriores da OBI.
Meta Cognitivo	★ Discussão sobre a importância da formatação de saída em programas mais complexos. ★ Análise de problemas de formatação em competições anteriores da OBI. ★ Abstração e contextualização do conceito de variáveis.

Fonte: o Autor.

A.1.9 Avaliação

Avaliação de Aprendizagem

- Realização correta dos exercícios de formatação com `printf()`;
- Sucesso no cadastro e submissão de problemas na plataforma BeeCrowd.

Avaliação da Atividade

- *Feedback* dos alunos sobre a clareza das instruções e a utilidade da atividade;
- Autoavaliação do professor sobre o andamento da aula e a participação dos alunos.

Avaliação dos Alunos

- Conversa sobre satisfação e sugestões de melhoria;
- Observação direta do engajamento e compreensão dos conceitos durante a aula.

Quadro 12: Atividades a serem desenvolvidas

Etapa	Duração (em min)	Atividades
Introdução	10	→ Apresentação da função <code>printf()</code> e sua importância. → Discussão sobre a relevância da saída de dados formatada.
Demonstração Prática	20	→ Exemplo de código em C utilizando <code>printf()</code> com diferentes especificadores de formato. → Exercícios práticos em pares para formatar saídas de texto.
Cadastro na Plataforma BeeCrowd	20	→ Orientação passo-a-passo para cadastro na plataforma. → Navegação inicial e apresentação da interface do BeeCrowd.
Atividade Prática	40	→ Submissão de problemas básicos na plataforma BeeCrowd. → Exercícios práticos em pares para formatar saídas de texto.
Conclusão e Reflexão	10	→ Recapitulação dos conceitos abordados. → Discussão sobre a importância da prática e da participação em competições. → Reflexão individual sobre o aprendizado da aula.

Fonte: o Autor.

Referências Bibliográficas

- [1] Sousa, Sidinei de Oliveira: *Aprendizagem baseada em problemas (pbl – problem-based learning): Estratégia para o ensino e aprendizagem de algoritmos e conteúdos computacionais*. Nuances: Estudos sobre Educação, 18(19):168, jun. 2011. <https://revista.fct.unesp.br/index.php/Nuances/article/view/363>.
- [2] Affeldt, Fabrício Sobrosa e Fábio Ferraz Fernandez: *Problem-based learning: um modelo para o ensino e a aprendizagem de teorias organizacionais*. Revista Ibero-Americana de Estudos em Educação, 13(esp.1):436–450, abr. 2018. <https://periodicos.fclar.unesp.br/iberoamericana/article/view/11437>.
- [3] Witt, Diego Teixeira e Avanilde Kemczinski: *Metodologias de aprendizagem ativa aplicadas à computação: Uma revisão da literatura*. Informática na educação: te-

oria e prática, 23(1 Jan/Abr), maio 2020. <https://seer.ufrgs.br/index.php/InfEducTeoriaPratica/article/view/90319>.

- [4] Kernighan, Brian W. e Dennis M. Ritchie: *The C Programming Language*. Prentice Hall Professional Technical Reference, 2nd edição, 1988, ISBN 0131103709.

A.2 Plano 2 - Leitura de dados e operadores aritméticos

A.2.1 Princípios Educacionais

Para esta aula, planejamos utilizar uma abordagem construtivista, onde o aprendizado é ativo e centrado no aluno, usando também a visão de aprendizagem baseada em problemas – PBL [1], [2]. Segundo Piaget e Vygotsky, o conhecimento é construído à medida que os alunos interagem com o ambiente e entre si. Portanto, a aula será prática e interativa [3], com ênfase em atividades que permitam aos alunos experimentar e resolver problemas reais, como o uso da função `scanf()` em C e a familiarização com a plataforma BeeCrowd⁵.

Competência Específica de Computação (BNCC)

“**Competência 6** – Expressar e partilhar informações, ideias, sentimentos e soluções computacionais utilizando diferentes plataformas, ferramentas, linguagens e tecnologias da Computação de forma fluente, criativa, crítica, significativa, reflexiva e ética.”

A.2.2 Objetivos Educacionais

Objetivos de ensino

- (a) Auxiliar os alunos a compreender o uso e a importância da função `scanf()` da linguagem C [4], compreendendo sua importância para leitura de dados de entrada fornecidos por usuários;
- (b) Desenvolver a capacidade de aplicar operadores aritméticos (soma, subtração, multiplicação, divisão e resto) em problemas computacionais, integrando lógica matemática à programação;
- (c) Facilitar a compreensão de operadores aritméticas em C, promovendo o uso desses operadores em programas que envolvam cálculos básicos;
- (d) Familiarizar os alunos com o contexto de competições, como a Olimpíada Brasileira de Informática⁶, enfatizando a importância da prática contínua em programação;
- (e) Propor a análise de diferentes tipos de variáveis e o processo de atribuição de valores, diferenciando entre variáveis pré-definidas e variáveis que recebem valores de entrada do usuário.

⁵<https://judge.beecrowd.com/pt/>

⁶<http://olimpiada.ic.unicamp.br/>

Objetivos de aprendizagem

- (a) Chamar a função `scanf()` em C para ler e armazenar dados de entrada fornecidos por usuários, de forma eficiente e segura;
- (b) Utilizar operadores aritméticos para resolver problemas envolvendo cálculos básicos, desenvolvendo soluções programáticas que incorporem operações de soma, subtração, multiplicação e divisão;
- (c) Avaliar o uso de operadores lógicos e aritméticos na resolução de problemas contextualizados, compreendendo sua importância no controle e verificação de condições;
- (d) Distinguir entre variáveis que são inicializadas com valores fixos e variáveis que recebem dados de entrada dos usuários, integrando essa prática na resolução de problemas.

A.2.3 Conceitos a serem desenvolvidos

- (i) O uso da função `scanf()`;
- (ii) Atribuição de valores a variáveis de diferentes tipos;
- (iii) Leitura de dados do usuário em um programa;
- (iv) Resolução de problemas usando operadores lógicos (+, -, *, /).

A.2.4 Público-Alvo

Alunos do segundo e terceiro ano do Ensino Médio do CED 04 de Taguatinga, inscritos no curso de preparação da OBI. Para essa aula prática, os alunos precisarão ter o conhecimento prévio sobre a função `printf()`, que será abordada na primeira aula. Como material de apoio, será disponibilizado um material complementar em formato de vídeo, com o objetivo de criar um repositório para esclarecimento de dúvidas.

A.2.5 Tema ou situação de contexto (organizadores prévios)

A primeira impressão é a que fica: a importância da saída de dados formatada em C. Como imprimir mensagens, valores e resultados de um programa na tela, e como submeter o primeiro programa na plataforma do BeeCrowd.

Contexto

Os alunos serão apresentados a cenários onde a entrada de dados precisa ser recebida e atribuída a diversas variáveis, e nesse cenário de armazenamento de valores, eles terão que manipular os dados, às vezes guardando o valor dessas operações em outras variáveis, e outras vezes imprimindo os valores usando as operações pedidas pelo problema proposto.

A.2.6 Níveis Cognitivos

Os níveis cognitivos (básico, intermediário e avançado) e os respectivos tópicos a serem trabalhados estão descritos na Quadro 13.

Quadro 13: Níveis cognitivos e tópicos a serem trabalhados

Nível	Tópicos
Básico	★ Introdução à função <code>scanf()</code> e seus especificadores de leitura
	★ Introdução ao conceito de atribuição de valores a variáveis, a partir de uma entrada de usuário
	★ Exercícios de leitura simples de valores com <code>scanf()</code> e impressão de valores manipulados pelos programas
Intermediário	★ Submissão dos problemas na plataforma BeeCrowd.
	★ Interpretação de erros apresentados pelo compilador
Avançado	★ Manipulação e impressão de valores e seus tipos em C
	★ Compreensão de ordem de operadores lógicos
	★ Compreender que tipos diferentes não são manipulados da mesma forma em C
Meta Cognitivo	★ Discussão sobre a diferença de resolução de problemas em C e no papel, prova

Fonte: o Autor

A.2.7 Descrição da Atividade

As atividades a serem desenvolvidas estão descritas na Quadro 14.

A.2.8 Recursos Materiais e tecnológicos

- (i) Computadores com acesso à internet;
- (ii) Conta na plataforma BeeCrowd;

Quadro 14: Atividades a serem desenvolvidas

Etapa	Duração (em min)	Atividades
Introdução	10	→ Apresentação da função <code>scanf()</code> e sua importância → Apresentação dos operadores (+, -, *, /)
Demonstração Prática	20	→ Resolução de um problema básico por parte dos alunos (discussão do passo a passo) → Contextualização da resolução feita oralmente a nível de implementação
Atividade Prática	40	→ Submissão de problemas de operadores lógicos, impressão e leitura de valores, previamente selecionados na plataforma Bee-Crowd. → Resolução de problemas em grupo e discussão dos feedbacks recebidos.
Conclusão e Reflexão	10	→ Recapitulação dos conceitos abordados. → Discussão sobre a importância da prática e da participação em competições. → Reflexão individual sobre o aprendizado da aula

Fonte: o Autor.

(iii) Editor de código (preferencialmente IDE para C, como Code::Blocks⁷ ou VSCode⁸, ou compilador online);

(iv) Projetor para demonstração.

A.2.9 Avaliação

Avaliação da Aprendizagem

- Realização correta dos exercícios de formatação com `scanf()`;

⁷<https://www.codeblocks.org/>

⁸<https://code.visualstudio.com/>

- Sucesso no cadastro e submissão de problemas na plataforma BeeCrowd.

Avaliação da Atividade

- *Feedback* dos alunos sobre a clareza das instruções e a utilidade da atividade;
- Autoavaliação do professor sobre o andamento da aula e a participação dos alunos.

Avaliação dos Alunos

- Conversa sobre satisfação e sugestões de melhoria;
- Observação direta do engajamento e compreensão dos conceitos durante a aula.

Referências Bibliográficas

- [1] Sousa, Sidinei de Oliveira: *Aprendizagem baseada em problemas (pbl – problem-based learning): Estratégia para o ensino e aprendizagem de algoritmos e conteúdos computacionais*. Nuances: Estudos sobre Educação, 18(19):168, jun. 2011. <https://revista.fct.unesp.br/index.php/Nuances/article/view/363>.
- [2] Affeldt, Fabrício Sobrosa e Fábio Ferraz Fernandez: *Problem-based learning: um modelo para o ensino e a aprendizagem de teorias organizacionais*. Revista Ibero-Americana de Estudos em Educação, 13(esp.1):436–450, abr. 2018. <https://periodicos.fclar.unesp.br/iberoamericana/article/view/11437>.
- [3] Witt, Diego Teixeira e Avanilde Kemczinski: *Metodologias de aprendizagem ativa aplicadas à computação: Uma revisão da literatura*. Informática na educação: teoria e prática, 23(1 Jan/Abr), maio 2020. <https://seer.ufrgs.br/index.php/InfEducTeoriaPratica/article/view/90319>.
- [4] Kernighan, Brian W. e Dennis M. Ritchie: *The C Programming Language*. Prentice Hall Professional Technical Reference, 2nd edição, 1988, ISBN 0131103709.

A.3 Plano 3 - Controle de fluxo/condicionais

A.3.1 Princípios Educacionais

Para esta aula, planejamos utilizar uma abordagem construtivista, onde o aprendizado é ativo e centrado no aluno, usando também a visão de aprendizagem baseada em problemas – PBL [1], [2]. Segundo Piaget e Vygotsky, o conhecimento é construído à medida que os alunos interagem com o ambiente e entre si. Portanto, a aula será prática e interativa [3], com ênfase em atividades que permitam aos alunos experimentar e resolver problemas reais, como o uso das estruturas de controle de fluxo e de tomadas de decisão, usando os comandos **if/else** e os operadores lógicos em C, continuando a adaptação com a plataforma BeeCrowd⁹.

Competência Específica de Computação (BNCC)

“**Competência 6** – Expressar e compartilhar informações, ideias, sentimentos e soluções computacionais utilizando diferentes plataformas, ferramentas, linguagens e tecnologias da Computação de forma fluente, criativa, crítica, significativa, reflexiva e ética.”

A.3.2 Objetivos Educacionais

Objetivos de ensino

- (a) Preparar os alunos para compreender o fluxo de execução de um programa na linguagem C [4], destacando a importância da ordem das instruções;
- (b) Desenvolver a capacidade de utilizar estruturas condicionais (**if/else**) para tomada de decisões em programas, promovendo o raciocínio lógico;
- (c) Capacitar os alunos a avaliar entradas e condições em um programa para determinar diferentes fluxos de execução, com base em operadores lógicos e estruturas condicionais;
- (d) Proporcionar uma compreensão clara do uso dos operadores lógicos (E/OU) e seu papel na avaliação de expressões booleanas em C.

Objetivos de aprendizagem

- (a) Compreender como as condições são avaliadas em programas escritos em C, utilizando operadores lógicos e estruturas de controle (**if/else**);

⁹<https://judge.beecrowd.com/pt/>

- (b) Resolver problemas que envolvam estruturas condicionais simples e aninhadas, aplicando o raciocínio lógico para determinar o fluxo correto de execução;
- (c) Avaliar o uso de condicionais aninhados (**if/else**) para resolver problemas mais complexos que demandam múltiplas verificações de condições;
- (d) Compreender como o computador interpreta expressões lógicas, determinando o resultado como verdadeiro (**true**) ou falso (**false**);
- (e) Aplicar os operadores lógicos E (**&&**) e OU (**||**) para construir expressões condicionais que dependam de múltiplas variáveis e condições.

A.3.3 Conceitos a serem desenvolvidos

- (i) Condicionais;
- (ii) Delimitadores de bloco de código '**{**' e '**}**';
- (iii) Noções de escopo;
- (iv) Operadores Lógicos.

A.3.4 Público-Alvo

Alunos do segundo e terceiro ano do Ensino Médio do CED 04 de Taguatinga, inscritos no curso de preparação da OBI. Para essa aula prática, os alunos precisarão ter conhecimento prévio sobre as funções `printf()` e `scanf()`, além de noções sobre variáveis. Esses conteúdos serão abordados nos primeiros encontros, e todos contarão com vídeos gravados para compor um repositório de apoio e esclarecimento de dúvidas.

A.3.5 Tema ou situação de contexto (organizadores prévios)

Contexto

Os alunos serão expostos a cenários onde o fluxo do programa terá que ser definido ao decorrer do problema, o que os levará a usar estruturas de controle de fluxo, avaliando condições, usando operadores lógicos quando necessário e tomando decisões com base nas avaliações de condicionais, usando a linguagem de programação C.

A.3.6 Níveis Cognitivos

Os níveis cognitivos (básico, intermediário e avançado) e os respectivos tópicos a serem trabalhados estão descritos na Quadro 15.

Quadro 15: Níveis cognitivos e tópicos a serem trabalhados

Nível	Tópicos
Básico	★ Introdução às estruturas <code>if/else</code>
	★ Introdução ao conceito de controle de fluxo
	★ Exercícios de tomadas de decisões baseadas em condições
Intermediário	★ Submissão dos problemas na plataforma BeeCrowd
	★ Interpretação dos problemas e análise da satisfatibilidade de condições usando operadores lógicos
Avançado	★ Compreensão de como um programa avalia uma condição e decide por um fluxo ou outro
	★ Compreensão dos operadores OU e E
Meta Cognitivo	★ Discussão de possíveis erros de interpretação e falhas na conversão de problemas reais para códigos na linguagem C

Fonte: o Autor.

A.3.7 Descrição da Atividade

As atividades a serem desenvolvidas estão descritas na Quadro 16.

A.3.8 Recursos Materiais e Tecnológicos

- (i) Computadores com acesso à internet;
- (ii) Conta na plataforma BeeCrowd;
- (iii) Editor de código (preferencialmente IDE para C, como Code::Blocks¹⁰ ou VSCode¹¹, ou compilador online);
- (iv) Projetor para demonstração.

A.3.9 Avaliação

Avaliação da Aprendizagem

- Realização correta dos exercícios de decisão com `if/else`;
- Sucesso no cadastro e submissão de problemas na plataforma BeeCrowd.

Avaliação da Atividade

- *Feedback* dos alunos sobre a clareza das instruções e a utilidade da atividade;

¹⁰<https://www.codeblocks.org/>

¹¹<https://code.visualstudio.com/>

Quadro 16: Atividades a serem desenvolvidas

Etapa	Duração (em min)	Atividades
Introdução	20	→ Apresentação dos comandos <code>if</code> e <code>else</code> , bem como de suas respectivas importâncias → Exemplos de fluxos de programa e como o computador avalia condições e toma decisões → Apresentação dos operadores lógicos E (<code>&&</code>) e OU (<code> </code>)
Atividade Prática	60	→ Submissão de problemas de estruturas de controle de fluxo e de condicionais, previamente selecionados na plataforma BeeCrowd → Resolução de problemas em grupo e discussão dos <i>feedbacks</i> recebidos
Conclusão e Reflexão	20	→ Recapitulação dos conceitos abordados → Discussão sobre a importância da prática e da participação em competições → Reflexão individual sobre o aprendizado da aula

Fonte: o Autor.

- Autoavaliação do professor sobre o andamento da aula e a participação dos alunos.

Avaliação dos Alunos

- Conversa sobre satisfação e sugestões de melhoria;
- Observação direta do engajamento e compreensão dos conceitos durante a aula.

Referências Bibliográficas

- [1] Sousa, Sidinei de Oliveira: *Aprendizagem baseada em problemas (pbl – problem-based learning): Estratégia para o ensino e aprendizagem de algoritmos e conteúdos computacionais*. Nuances: Estudos sobre Educação, 18(19):168, jun. 2011. <https://revista.fct.unesp.br/index.php/Nuances/article/view/363>.

- [2] Affeldt, Fabrício Sobrosa e Fábio Ferraz Fernandez: *Problem-based learning: um modelo para o ensino e a aprendizagem de teorias organizacionais*. Revista Ibero-Americana de Estudos em Educação, 13(esp.1):436–450, abr. 2018. <https://periodicos.fclar.unesp.br/iberoamericana/article/view/11437>.
- [3] Witt, Diego Teixeira e Avanilde Kemczinski: *Metodologias de aprendizagem ativa aplicadas à computação: Uma revisão da literatura*. Informática na educação: teoria e prática, 23(1 Jan/Abr), maio 2020. <https://seer.ufrgs.br/index.php/InfEducTeoriaPratica/article/view/90319>.
- [4] Kernighan, Brian W. e Dennis M. Ritchie: *The C Programming Language*. Prentice Hall Professional Technical Reference, 2nd edição, 1988, ISBN 0131103709.

A.4 Plano 4 - Laços de repetição - **while**

A.4.1 Princípios Educacionais

Para esta aula, planejamos utilizar uma abordagem construtivista, onde o aprendizado é ativo e centrado no aluno, usando também a visão de aprendizagem baseada em problemas – PBL [1], [2]. Segundo Piaget e Vygotsky, o conhecimento é construído à medida que os alunos interagem com o ambiente e entre si. Portanto, a aula será prática e interativa [3], com ênfase em atividades que permitam aos alunos experimentar e resolver problemas reais, como o uso de laços de repetição em C (mais precisamente o comando **while**), continuando a adaptação com a plataforma BeeCrowd¹².

Competência Específica de Computação (BNCC)

“**Competência 6** – Expressar e partilhar informações, ideias, sentimentos e soluções computacionais utilizando diferentes plataformas, ferramentas, linguagens e tecnologias da Computação de forma fluente, criativa, crítica, significativa, reflexiva e ética.”

A.4.2 Objetivos Educacionais

Objetivos de ensino

- (a) Preparar os alunos para compreender o conceito de laço de repetição, com foco no comando **while** e sua aplicação prática em programas na linguagem C [4];
- (b) Desenvolver a capacidade de identificar e definir condições de parada adequadas em um *loop*, garantindo que o programa funcione corretamente e evite *loops* infinitos;
- (c) Promover o entendimento sobre como automatizar código por meio da repetição, economizando esforços manuais e evitando redundâncias no código.

Objetivos de aprendizagem

- (a) Compreender como o laço de repetição **while** funciona, analisando as condições que determinam se um *loop* continua ou para;
- (b) Desenvolver a habilidade de usar variáveis iteradoras para controlar e gerenciar o fluxo de repetição em um *loop*;
- (c) Avaliar e tomar decisões sobre as condições de parada mais apropriadas para cada problema, garantindo a eficiência e execução correta do código;

¹²<https://judge.beecrowd.com/pt/>

- (d) Compreender em quais situações o uso de laços de repetição, especialmente o **while**, é recomendado ou necessário, e aplicar essa estrutura para resolver problemas de programação repetitivos.

A.4.3 Conceitos a serem desenvolvidos

- (i) Laços de repetição;
- (ii) Comando **while**;
- (iii) Noções de escopo;
- (iv) Condições de parada.

A.4.4 Público-Alvo

Alunos do segundo e terceiro ano do Ensino Médio do CED 04 de Taguatinga, inscritos no curso de preparação da OBI. Para essa aula prática, os alunos precisarão ter algum conhecimento prévio sobre as funções **printf()** e **scanf()**, noções sobre variáveis, estruturas de controle de fluxo e condicionais. Esses conceitos serão ministrados nos primeiros encontros e todos contarão com material de apoio em vídeo, gravado com o intuito de compor um repositório de esclarecimento de dúvidas.

A.4.5 Tema ou situação de contexto (organizadores prévios)

Contexto

Os alunos serão expostos a cenários nos quais, dependendo da estrutura do problema, precisarão decidir sobre o uso de laços de repetição (**while**) e suas condições de parada, automatizando o programa e evitando o uso repetitivo e desnecessário de comandos.

A.4.6 Níveis Cognitivos

Os níveis cognitivos (básico, intermediário e avançado) e os respectivos tópicos a serem trabalhados estão descritos na Quadro 17.

A.4.7 Descrição da Atividade

As atividades a serem desenvolvidas estão descritas na Quadro 18.

Quadro 17: Níveis cognitivos e tópicos a serem trabalhados

Nível	Tópicos
Básico	★ Introdução ao comando while ★ Introdução ao conceito de condições de parada ★ Exercícios de <i>loops</i>
Intermediário	★ Submissão dos problemas na plataforma BeeCrowd ★ Interpretação dos problemas e análise da satisfatibilidade de condições usando operadores lógicos
Avançado	★ Compreensão de como um programa avalia uma condição de parada e como uma programa decide por executar ou não um laço de repetição ★ Compreensão dos operadores OU e E
Meta Cognitivo	★ Discussão de possíveis erros de interpretação e falhas na conversão de problemas reais para códigos na linguagem C

Fonte: o Autor.

Quadro 18: Atividades a serem desenvolvidas

Etapa	Duração (em min)	Atividades
Introdução	20	→ Apresentação do comando while e de sua importância → Exemplo de programas que se repetem e discussão de como automatizá-los → Reforço dos operadores usados para comparação de condições (<code>==</code>, <code>!=</code>, <code><=</code>, <code>>=</code>, <code><</code> e <code>></code>)
Atividade Prática	60	→ Submissão de problemas de estruturas de controle de fluxo e de condicionais, previamente selecionados na plataforma BeeCrowd → Resolução de problemas em grupo e discussão dos <i>feedbacks</i> recebidos
Conclusão e Reflexão	20	→ Recapitulação dos conceitos abordados → Reflexão individual sobre o aprendizado da aula

Fonte: o Autor.

A.4.8 Recursos Materiais e Tecnológicos

- (i) Computadores com acesso à internet;

- (ii) Conta na plataforma BeeCrowd;
- (iii) Editor de código (preferencialmente IDE para C, como Code::Blocks¹³ ou VSCode¹⁴, ou compilador online);
- (iv) Projetor para demonstração.

A.4.9 Avaliação

Avaliação de Aprendizagem

- Realização correta dos exercícios de laços de repetição usando `while`;
- Sucesso na submissão de problemas na plataforma BeeCrowd.

Avaliação de Aprendizagem

- *Feedback* dos alunos sobre a clareza das instruções e a utilidade da atividade;
- Autoavaliação do professor sobre o andamento da aula e a participação dos alunos.

Avaliação de Aprendizagem

- Conversa sobre satisfação e sugestões de melhoria;
- Observação direta do engajamento e compreensão dos conceitos durante a aula.

Referências Bibliográficas

- [1] Sousa, Sidinei de Oliveira: *Aprendizagem baseada em problemas (pbl – problem-based learning): Estratégia para o ensino e aprendizagem de algoritmos e conteúdos computacionais*. Nuances: Estudos sobre Educação, 18(19):168, jun. 2011. <https://revista.fct.unesp.br/index.php/Nuances/article/view/363>.
- [2] Affeldt, Fabrício Sobrosa e Fábio Ferraz Fernandez: *Problem-based learning: um modelo para o ensino e a aprendizagem de teorias organizacionais*. Revista Ibero-Americana de Estudos em Educação, 13(esp.1):436–450, abr. 2018. <https://periodicos.fclar.unesp.br/iberoamericana/article/view/11437>.

¹³<https://www.codeblocks.org/>

¹⁴<https://code.visualstudio.com/>

- [3] Witt, Diego Teixeira e Avamilde Kemczinski: *Metodologias de aprendizagem ativa aplicadas à computação: Uma revisão da literatura*. Informática na educação: teoria e prática, 23(1 Jan/Abr), maio 2020. <https://seer.ufrgs.br/index.php/InfEducTeoriaPratica/article/view/90319>.
- [4] Kernighan, Brian W. e Dennis M. Ritchie: *The C Programming Language*. Prentice Hall Professional Technical Reference, 2nd edição, 1988, ISBN 0131103709.

A.5 Plano 5 - Laços de repetição - **for**

A.5.1 Princípios Educacionais

Para esta aula, planejamos utilizar uma abordagem construtivista, onde o aprendizado é ativo e centrado no aluno, usando também a visão de aprendizagem baseada em problemas – PBL [1], [2]. Segundo Piaget e Vygotsky, o conhecimento é construído à medida que os alunos interagem com o ambiente e entre si. Portanto, a aula será prática e interativa [3], com ênfase em atividades que permitam aos alunos experimentar e resolver problemas reais, como o uso de laços de repetição em C (mais precisamente o comando **for**), continuando a adaptação à plataforma BeeCrowd¹⁵.

Competência Específica de Computação (BNCC)

“**Competência 6** – Expressar e partilhar informações, ideias, sentimentos e soluções computacionais utilizando diferentes plataformas, ferramentas, linguagens e tecnologias da Computação de forma fluente, criativa, crítica, significativa, reflexiva e ética.”

A.5.2 Objetivos Educacionais

Objetivos de ensino

- (a) Preparar os alunos para compreender o conceito de laço de repetição, com foco no comando **for** e sua aplicação prática em programas na linguagem C [4];
- (b) Desenvolver a capacidade de identificar e definir condições de parada adequadas em um *loop*, garantindo que o programa funcione corretamente e evite *loops* infinitos;
- (c) Promover o entendimento sobre como automatizar código por meio da repetição, economizando esforços manuais e evitando redundâncias no código.

Objetivos de aprendizagem

- (a) Compreender como o laço de repetição **for** funciona, analisando as condições que determinam se um *loop* continua ou para;
- (b) Desenvolver a habilidade de usar variáveis iteradoras para controlar e gerenciar o fluxo de repetição em um *loop*;
- (c) Avaliar e tomar decisões sobre as condições de parada mais apropriadas para cada problema, garantindo a eficiência e execução correta do código;

¹⁵<https://judge.beecrowd.com/pt/>

- (d) Compreender em quais situações o uso de laços de repetição, especialmente o **for**, é recomendado ou necessário, e aplicar esse comando para resolver problemas de programação repetitivos.

A.5.3 Conceitos a serem desenvolvidos

- (i) Laços de repetição;
- (ii) Comando **for**;
- (iii) Noções de escopo;
- (iv) Condições de parada.

A.5.4 Público-Alvo

Alunos do segundo e terceiro ano do Ensino Médio do CED 04 de Taguatinga, inscritos no curso de preparação da OBI. Para essa aula prática, os alunos precisarão ter algum conhecimento prévio sobre as funções `printf()` e `scanf()`, noções sobre variáveis, estruturas de controle de fluxo e condicionais. Esses conceitos serão ministrados nos primeiros encontros e todos contarão com material de apoio em vídeo, gravado com o intuito de compor um repositório de esclarecimento de dúvidas.

A.5.5 Tema ou situação de contexto (organizadores prévios)

Contexto

Os alunos serão expostos a cenários nos quais, dependendo da estrutura do problema, precisarão decidir sobre o uso de laços de repetição (**for**) e suas condições de parada, automatizando o programa e evitando o uso repetitivo e desnecessário de comandos.

A.5.6 Níveis Cognitivos

Os níveis cognitivos (básico, intermediário e avançado) e os respectivos tópicos a serem trabalhados estão descritos na Quadro 19.

A.5.7 Descrição da Atividade

As atividades a serem desenvolvidas estão descritas na Quadro 20.

Quadro 19: Níveis cognitivos e tópicos a serem trabalhados

Nível	Tópicos
Básico	★ Introdução ao comando for ★ Introdução ao conceito de condições de parada ★ Exercícios de <i>loops</i>
Intermediário	★ Submissão dos problemas na plataforma BeeCrowd ★ Interpretação dos problemas e análise da satisfatibilidade de condições usando operadores lógicos
Avançado	★ Compreensão de como um programa avalia uma condição de parada e como uma programa decide por executar ou não um laço de repetição ★ Compreensão dos operadores OU e E
Meta Cognitivo	★ Discussão de possíveis erros de interpretação e falhas na conversão de problemas reais para códigos na linguagem C

Fonte: o Autor.

Quadro 20: Atividades a serem desenvolvidas

Etapa	Duração (em min)	Atividades
Introdução	20	→ Apresentação do comando for e de sua importância → Exemplo de programas que se repetem e discussão de como automatizá-los → Reforço dos operadores usados para comparação de condições (<code>==</code>, <code>!=</code>, <code><=</code>, <code>>=</code>, <code><</code> e <code>></code>)
Atividade Prática	60	→ Submissão de problemas de estruturas de controle de fluxo e de condicionais, previamente selecionados na plataforma BeeCrowd → Resolução de problemas em grupo e discussão dos <i>feedbacks</i> recebidos
Conclusão e Reflexão	20	→ Recapitulação dos conceitos abordados → Reflexão individual sobre o aprendizado da aula

Fonte: o Autor.

A.5.8 Recursos Materiais e Tecnológicos

- (i) Computadores com acesso à internet;

- (ii) Conta na plataforma BeeCrowd;
- (iii) Editor de código (preferencialmente IDE para C, como Code::Blocks¹⁶ ou VSCode¹⁷, ou compilador online);
- (iv) Projetor para demonstração.

A.5.9 Avaliação

Avaliação de Aprendizagem

- Realização correta dos exercícios de laços de repetição usando `for`;
- Sucesso na submissão de problemas na plataforma BeeCrowd.

Avaliação de Aprendizagem

- *Feedback* dos alunos sobre a clareza das instruções e a utilidade da atividade;
- Autoavaliação do professor sobre o andamento da aula e a participação dos alunos.

Avaliação de Aprendizagem

- Conversa sobre satisfação e sugestões de melhoria;
- Observação direta do engajamento e compreensão dos conceitos durante a aula.

Referências Bibliográficas

- [1] Sousa, Sidinei de Oliveira: *Aprendizagem baseada em problemas (pbl – problem-based learning): Estratégia para o ensino e aprendizagem de algoritmos e conteúdos computacionais*. Nuances: Estudos sobre Educação, 18(19):168, jun. 2011. <https://revista.fct.unesp.br/index.php/Nuances/article/view/363>.
- [2] Affeldt, Fabrício Sobrosa e Fábio Ferraz Fernandez: *Problem-based learning: um modelo para o ensino e a aprendizagem de teorias organizacionais*. Revista Ibero-Americana de Estudos em Educação, 13(esp.1):436–450, abr. 2018. <https://periodicos.fclar.unesp.br/iberoamericana/article/view/11437>.

¹⁶<https://www.codeblocks.org/>

¹⁷<https://code.visualstudio.com/>

- [3] Witt, Diego Teixeira e Avaniilde Kemczinski: *Metodologias de aprendizagem ativa aplicadas à computação: Uma revisão da literatura*. Informática na educação: teoria e prática, 23(1 Jan/Abr), maio 2020. <https://seer.ufrgs.br/index.php/InfEducTeoriaPratica/article/view/90319>.
- [4] Kernighan, Brian W. e Dennis M. Ritchie: *The C Programming Language*. Prentice Hall Professional Technical Reference, 2nd edição, 1988, ISBN 0131103709.

A.6 Plano 6 - Mini OBI

A.6.1 Princípios Educacionais

Para esta aula planejamos fazer uma simulação de uma competição de programação. A ideia é ambientar os alunos com experiências que podem se tornar realidade nas rotinas deles, que acontecem no dia a dia dessa área (na UnB, por exemplo, existem maratonas de programação competitiva) e que vão prepará-los para a prova da OBI.

A.6.2 Objetivos Educacionais

Objetivos de ensino

- (a) Familiarizar os alunos com o formato de maratonas competitivas de programação, como a Olimpíada Brasileira de Informática (OBI)¹⁸;
- (b) Capacitar os alunos a identificar e aplicar as estruturas adequadas para resolver problemas em tempo limitado, como em competições;
- (c) Auxiliar os alunos a compreender o estilo e o formato de prova da OBI, incluindo o gerenciamento de tempo e estratégia para resolver problemas de forma eficiente.

Objetivos de aprendizagem

- (a) Chamar a função `printf()` em C para exibir saídas formatadas, utilizando corretamente os especificadores de formato adequados para cada tipo de dado;
- (b) Desenvolver a habilidade de utilizar corretamente as estruturas aprendidas (laços, condicionais, etc.) para resolver problemas de competição;
- (c) Analisar e escolher as melhores estruturas de controle e decisão para resolver problemas específicos, levando em conta a eficiência;
- (d) Avaliar e gerenciar o tempo disponível para resolução de questões, aprendendo a otimizar o tempo disponível para resolver o máximo de questões possível.

A.6.3 Conceitos a serem desenvolvidos

- (i) Impressão e leitura de dados em variáveis;
- (ii) Tipos de dados;

¹⁸<https://olimpiada.ic.unicamp.br/>

- (iii) Noções de escopo;
- (iv) Laços de repetição;
- (v) Estruturas de controle de fluxo e decisão.

A.6.4 Público-Alvo

Alunos do segundo e terceiro ano do Ensino Médio do CED 04 de Taguatinga, inscritos no curso de preparação da OBI. Para essa aula prática, os alunos precisarão ter algum conhecimento prévio sobre as funções `printf()` e `scanf()`, noções sobre variáveis, estruturas de controle de fluxo e condicionais. Esses conceitos serão ministrados nos primeiros encontros e todos contarão com material de apoio em vídeo, gravado com o intuito de compor um repositório de esclarecimento de dúvidas.

A.6.5 Tema ou situação de contexto (organizadores prévios)

Contexto

Os alunos serão expostos a cenários de competição, onde poderão usar todo o conteúdo aprendido e aplicar na simulação de uma competição de programação, com aproximadamente 2 horas para a realização de até 6 problemas, de dificuldades diversas.

A.6.6 Níveis Cognitivos

Os níveis cognitivos (básico, intermediário e avançado) e os respectivos tópicos a serem trabalhados estão descritos na Tabela A.1.

Tabela A.1: Níveis cognitivos e tópicos a serem trabalhados

Nível	Tópicos
Básico	★ Resolução de problemas básicos (soma, subtração, etc.), usando variáveis, impressão e leitura de valores
Intermediário	★ Resolução de problemas mais elaborados, envolvendo manipulação de valores, impressão formatada de mensagens e resultados e estruturas de decisão
Avançado	★ Resolução de problemas usando laços de repetição e estruturas de controle de fluxo/decisão, concomitantemente
Meta Cognitivo	★ Percepção do próprio aprendizado, entendimento da estratégia de prova, controle de ansiedade/nervosismo e assimilação do conteúdo proposto

Fonte: o Autor.

A.6.7 Descrição da Atividade

As atividades a serem desenvolvidas estão descritas na Tabela A.2.

Tabela A.2: Atividades a serem desenvolvidas

Etapa	Duração (em min)	Atividades
Introdução	10	→ Apresentação do modelo de maratonas competitivas e explicação das regras da prova
Atividade Prática	120	→ Realização dos exercícios propostos na plataforma MOJ ¹⁹ e submissão dos problemas com <i>feedback</i> imediato

Fonte: o Autor.

A.6.8 Recursos Materiais e Tecnológicos

- (i) Computadores com acesso à internet;
- (ii) Acesso à plataforma MOJ;
- (iii) Editor de código (preferencialmente IDE para C, como Code::Blocks²⁰ ou VSCode²¹, ou compilador online);
- (iv) Projetor para demonstração.

A.6.9 Avaliação

Avaliação de Aprendizagem

- Desempenho e comportamento dos alunos ao longo da competição.

Avaliação da Atividade

- *Feedback* dos alunos sobre a percepção de aprendizado e expectativa para a prova da OBI.

Avaliação dos Alunos

- Conversa sobre satisfação e sugestões de melhoria.

²⁰<https://www.codeblocks.org/>

²¹<https://code.visualstudio.com/>

A.7 Plano 7 - Laços de repetição - *Arrays*

A.7.1 Princípios Educacionais

Para esta aula planejamos utilizar uma abordagem construtivista, onde o aprendizado é ativo e centrado no aluno, usando também a visão de aprendizagem baseada em problemas – PBL [1], [2]. Segundo Piaget e Vygotsky, o conhecimento é construído à medida que os alunos interagem com o ambiente e entre si. Portanto, a aula será prática e interativa [3], com ênfase em atividades que permitam aos alunos experimentar e resolver problemas reais, usando conceitos de estruturas de dados como *arrays*, sempre continuando a adaptação à plataforma BeeCrowd²².

Competência Específica de Computação (BNCC)

“**Competência 6** – Expressar e partilhar informações, ideias, sentimentos e soluções computacionais utilizando diferentes plataformas, ferramentas, linguagens e tecnologias da Computação de forma fluente, criativa, crítica, significativa, reflexiva e ética.”

A.7.2 Objetivos Educacionais

Objetivos de ensino

- (a) Compreender o conceito de uma estrutura de dados capaz de armazenar múltiplos valores de um mesmo tipo;
- (b) Avaliar o uso de *arrays* na resolução de problemas;
- (c) Identificar a semelhança da resolução de problemas que usam os laços **while** e **for**, e compreender como esses mesmos problemas podem ser resolvidos usando *arrays*.

Objetivos de aprendizagem

- (a) Compreender como inicializar e atribuir valores a um *array*, utilizando laços de repetição ou atribuição direta;
- (b) Compreender como percorrer os elementos de um *array*;
- (c) Decidir quando usar *arrays* e avaliar o uso de laços de repetição para percorrer e interagir com valores armazenados no *array*.

²²<https://judge.beecrowd.com/pt/>

A.7.3 Conceitos a serem desenvolvidos

- (i) *Arrays*;
- (ii) Estruturas de Dados;
- (iii) Noções de memória;
- (iv) Acesso aos índices de um *array*.

A.7.4 Público-Alvo

Alunos do segundo e terceiro ano do Ensino Médio do CED 04 de Taguatinga, inscritos no curso de preparação da OBI. Para essa aula prática, os alunos precisarão ter algum conhecimento prévio sobre as funções `printf()` e `scanf()`, noções sobre variáveis, estruturas de controle de fluxo e condicionais. Esses conceitos serão ministrados nos primeiros encontros e todos contarão com material de apoio em vídeo, gravado com o propósito de esclarecimento de dúvidas.

A.7.5 Tema ou situação de contexto (organizadores prévios)

Contexto

Os alunos serão expostos a cenários onde precisarão usar estruturas de dados como *arrays* para armazenar valores do mesmo tipo e resolver problemas propostos sobre esse tema. Esses problemas envolvem acessar e manipular valores dentro de um *array*, além de preencher e imprimir *arrays*.

A.7.6 Níveis Cognitivos

Os níveis cognitivos (básico, intermediário e avançado) e os respectivos tópicos a serem trabalhados estão descritos na Quadro 21.

A.7.7 Descrição da Atividade

As atividades a serem desenvolvidas estão descritas na Quadro 22.

A.7.8 Recursos Materiais e Tecnológicos

- (i) Computadores com acesso à internet;
- (ii) Conta na plataforma BeeCrowd;

Quadro 21: Níveis cognitivos e tópicos a serem trabalhados

Nível	Tópicos
Básico	★ Introdução à estrutura de dados <i>array</i> ★ Armazenamento de múltiplos valores do mesmo tipo em uma só variável ★ Exercícios de <i>arrays</i>
Intermediário	★ Submissão dos problemas na plataforma BeeCrowd ★ Exercícios envolvendo o uso de laços de repetição para preenchimento e acesso aos índices de um array
Avançado	★ Compreender como fazer uso de <i>arrays</i> para resolver problemas complexos
Meta Cognitivo	★ Discussão de sobre o uso de <i>arrays</i> e laços de repetição: quando armazenar dados?

Fonte: o Autor.

Quadro 22: Atividades a serem desenvolvidas

Etapa	Duração (em min)	Atividades
Introdução	20	→ Apresentação da estrutura de dados <i>array</i> e um breve exemplo de como acessar um elemento preenchido no momento de sua declaração
Atividade Prática	60	→ Submissão de problemas de estruturas de controle de fluxo, condicionais e laços de repetição, com o acréscimo de estruturas de dados, previamente selecionados na plataforma BeeCrowd → Resolução de problemas em grupo e discussão dos <i>feedbacks</i> recebidos
Conclusão e Reflexão	20	→ Recapitulação dos conceitos abordados → Reflexão individual sobre o aprendizado da aula

Fonte: o Autor.

- (iii) Editor de código (preferencialmente IDE para C, como Code::Blocks²³ ou VSCode²⁴, ou compilador online);
- (iv) Projetor para demonstração.

A.7.9 Avaliação

Avaliação da Aprendizagem

- Realização correta dos exercícios de *arrays*;
- Sucesso na submissão de problemas na plataforma BeeCrowd.

Avaliação da Atividade

- *Feedback* dos alunos sobre a clareza das instruções e a utilidade da atividade;
- Autoavaliação do professor sobre o andamento da aula e a participação dos alunos.

Avaliação dos Alunos

- Conversa sobre satisfação e sugestões de melhoria;
- Observação direta do engajamento e compreensão dos conceitos durante a aula.

Referências Bibliográficas

- [1] Sousa, Sidinei de Oliveira: *Aprendizagem baseada em problemas (pbl – problem-based learning): Estratégia para o ensino e aprendizagem de algoritmos e conteúdos computacionais*. Nuances: Estudos sobre Educação, 18(19):168, jun. 2011. <https://revista.fct.unesp.br/index.php/Nuances/article/view/363>.
- [2] Affeldt, Fabrício Sobrosa e Fábio Ferraz Fernandez: *Problem-based learning: um modelo para o ensino e a aprendizagem de teorias organizacionais*. Revista Ibero-Americana de Estudos em Educação, 13(esp.1):436–450, abr. 2018. <https://periodicos.fclar.unesp.br/iberoamericana/article/view/11437>.
- [3] Witt, Diego Teixeira e Avanilde Kemczinski: *Metodologias de aprendizagem ativa aplicadas à computação: Uma revisão da literatura*. Informática na educação: teoria e prática, 23(1 Jan/Abr), maio 2020. <https://seer.ufrgs.br/index.php/InfEducTeoriaPratica/article/view/90319>.

²³<https://www.codeblocks.org/>

²⁴<https://code.visualstudio.com/>