

Aprendendo a Voar (e a Programar): Um Relato de Experiência sobre o Ensino de Programação Esportiva com o Uso de Metodologias Ativas e Gamificação

Matheus Rodrigues Alves¹, Thalia Santos de Santana¹, Adriano Honorato Braga¹

¹Instituto Federal Goiano (IF Goiano) – Campus Ceres – Ceres-GO – Brasil

matheus.alves2@estudante.ifgoiano.edu.br,
{thalia.santana, adriano.braga}@ifgoiano.edu.br

Abstract. *Sports programming or competition is associated with solving logical problems using codes within a time limit. However, programming teaching tends to adopt traditional methodologies, which do not always motivate students. Pedagogical Architecture (PA) can contribute to student-centered methodological organization. This study presents a PA focused on the context of the Brazilian Informatics Olympiad, with the application of ten lessons focused on Programming. For the PA, active methodologies such as collaborative practices and gamification elements were evaluated. As a result, these methodologies demonstrated positive factors for students.*

Resumo. *A programação esportiva ou competitiva está associada à resolução de problemas por meio de códigos em um tempo limite. No entanto, o ensino de programação esportiva tende a adotar metodologias tradicionais, que nem sempre motivam os discentes. As Arquiteturas Pedagógicas (AP) podem contribuir ao prover uma organização metodológica voltada ao estudante. O presente trabalho apresenta o relato de experiência de uma AP no contexto da Olimpíada Brasileira de Informática na Modalidade de Programação. Os resultados de uma instância com estudantes de ensino médio técnico, demonstram os benefícios de metodologias ativas como práticas colaborativas e elementos de gamificação para o engajamento e aprendizado dos estudantes.*

1. Introdução

A programação esportiva ou competitiva é um esporte intelectual que desafia os participantes a resolverem problemas lógicos e matemáticos por meio da codificação em linguagens de programação, dentro de um tempo limite [Nunes et al. 2024]. Ainda segundo os autores, essa modalidade favorece o desenvolvimento de diversas competências, entre elas o raciocínio lógico-matemático, a organização das ideias, a capacidade de identificar problemas do cotidiano e outras habilidades relevantes.

Atualmente, um dos principais exemplos de programação esportiva, é a Olimpíada Brasileira de Informática (OBI), realizada pela Sociedade Brasileira de Computação (SBC) e organizada pelo Instituto de Computação da Universidade Estadual de Campinas (Unicamp) desde 1999 [OBI 2025]. A OBI é voltada para estudantes do ensino fundamental, ensino médio e primeiro ano de graduação, separando em duas modalidades: Iniciação (para estudantes do ensino fundamental) e Programação (para estudantes do ensino médio e do primeiro ano de graduação).

Ainda assim, observa-se que a programação voltada à competição

frequentemente se apoia em métodos tradicionais de ensino [Garcia and Oliveira 2022], o que pode dificultar a aprendizagem ao reforçar o papel central do professor e privilegiar aulas expositivas. Neste cenário, como apresentado por [Irion et al. 2024], o ensino de programação esportiva é um objetivo complexo, envolvendo a necessidade de metodologias dinâmicas, tal como a proposta educacional de uma Arquitetura Pedagógica (AP) [Carvalho et al. 2005].

A AP parte do entendimento de que aprender é um movimento contínuo e dinâmico [Tavares et al. 2012], no qual os estudantes constroem sentido a partir de uma participação ativa no contato com a realidade. Essa concepção dialoga com a pedagogia da pergunta, ao valorizar o conhecimento que nasce da interação, da curiosidade e do ato de questionar nas relações entre professor e estudante [Freire 1975]. Do mesmo modo, aproxima-se dos fundamentos do construtivismo [Piaget 1985], ao reconhecer que novas compreensões podem emergir justamente da dúvida, da problematização e da contestação de verdades aparentemente consolidadas [Carvalho et al. 2005].

Segundo [Tavares et al. 2012], a AP é uma proposta educacional em sincronia com as inovações tecnológicas, rompendo com a concepção tradicional do espaço físico da sala de aula como o único ambiente de aprendizagem. Essa abordagem é definida pelo desenvolvimento de metodologias que baseiam-se em técnicas pedagógicas fundamentadas em ferramentas digitais [Mocelin and Fiuza 2021], permitindo assim integração com software, inteligência artificial, internet e educação à distância, etc.

Para [Carvalho et al. 2005], a AP também se orienta pela necessidade de estruturar e articular tarefas que conduzam à aprendizagem de um conteúdo específico, de modo a alcançar um objetivo educacional definido. Essa estruturação envolve a integração de componentes fundamentais do processo de aprendizagem, como a abordagem pedagógica adotada e os recursos digitais utilizados. Além disso, o modelo de organização proposto pela AP é aplicável a qualquer área do conhecimento [Tavares et al. 2012], incluindo, por exemplo, o ensino de programação esportiva.

De acordo com [Santos and Machado 2024] a gamificação, por sua vez, complementa a proposta da AP ao enriquecer o processo de ensino, promovendo a comunicação e a interação social. A gamificação, como é apresentado pelos autores, se destaca dentro da arquitetura como um meio promissor para criar contextos de aprendizagens mais envolventes e motivadoras aos discentes. Essa potencialidade torna o ambiente de aprendizagem mais dinâmico, pois a participação ativa por meio de jogos torna o estudante protagonista no processo de ensino, também permitindo a colaboração entre eles [de Farias et al. 2018].

Diante disso, o objetivo deste estudo é apresentar uma AP gamificada voltada ao ensino de programação esportiva aplicada à OBI Modalidade Programação, denominada “Libélulas”, bem como a percepção dos estudantes sobre a instância da arquitetura. A AP possui o intuito de aperfeiçoar a prática pedagógica para o ensino de programação competitiva, podendo ser aplicada em atividades de ensino e cursos preparatórios para olimpíadas. Sua instanciação ocorreu no ano de 2025 por meio de um projeto de ensino, visando fomentar a participação estudantil na OBI.

2. Trabalhos relacionados

A AP é uma estrutura de aprendizagem aplicada em diversas áreas de conhecimento, como em contextos voltados à Computação [Menezes et al. 2021]. Em específico para programação de computadores, [Portilho et al. 2024] apresentaram uma AP focada no ensino de lógica de programação, por meio de um projeto de extensão utilizando a linguagem de programação Python. A AP é aplicada com estudantes iniciantes em lógica, utilizando do *Google Colab* e um site próprio desenvolvido, com a disponibilização dos planos de aulas e questões, bem como de uma ferramenta de apoio ao docente. Os resultados da avaliação apontaram que a estrutura desenvolvida desencadeou a participação positiva do público-alvo na ação.

Em [Tavares et al. 2013], os autores implementaram uma AP direcionada ao ensino e à aprendizagem de programação com suporte de tecnologias educacionais. A proposta incorporou diferentes recursos, como *wiki*, chats, fórum de dúvidas e questionários, utilizados para apoiar tanto o acompanhamento quanto o desenvolvimento dos estudantes na disciplina de algoritmos. Ao final, apontam que a AP permitiu a cooperação e interação dos estudantes.

[Lima et al. 2024] apresentaram uma AP apoiada por Inteligência Artificial, aplicada em um curso de extensão para introdução da linguagem Python. Como parte do trabalho, foi realizada uma mineração de dados a partir da interação dos estudantes com a ferramenta Moodle. Os autores apontaram que os estudantes que usaram o recurso da AP tiveram melhor resultado no curso. Outrossim, o trabalho de [Rezende et al. 2024] relatou uma proposta de AP aplicada ao ensino de Pensamento Computacional (PC). Esse estudo desenvolveu a AP a partir de um jogo que apresenta os pilares de PC de forma desplugada, apresentando apenas a proposta da arquitetura, sem avaliação dos resultados alcançados.

Diante dos trabalhos citados, é perceptível a diversidade de estudos que empregam AP na Computação, com vistas a prover melhorias no ensino-aprendizagem de conceitos computacionais, especialmente, em programação. Todavia, observa-se a falta de pesquisas voltadas à AP especificamente para programação esportiva. Deste modo, este estudo visa colaborar com o rol de propostas de APs aplicadas na Computação, tratando em especial, da participação em olimpíadas de informática.

3. Arquitetura pedagógica “Libélulas”

A denominação da AP relatada neste artigo deriva das libélulas, insetos frequentemente associados à agilidade, leveza e adaptação a variados ambientes. Assim como sua capacidade de voar e mudar de direção com precisão, a AP busca promover abordagens flexíveis e dinâmicas para a resolução de problemas no âmbito da OBI. Essa proposta também considera experiências anteriores de treinamentos para olimpíadas de informática em uma instituição de ensino ao longo de mais dez anos. Com base nos elementos essenciais definidos por [Carvalho et al. 2005], a AP “Libélulas” [Alves et al. 2025] foi concebida nos seguintes termos:

1. **Domínio de conhecimento:** Habilidades de programação esportiva com Python, voltadas à Modalidade de Programação da OBI no Nível 1.
2. **Objetivos educacionais:** Desenvolvimento de conhecimentos em lógica e

programação esportiva, como forma de descoberta e atração de talentos para a Computação.

3. **Conhecimento prévio:** Conhecimento em Matemática Básica e, caso haja, PC. Visto que o público-alvo trata-se de estudantes do primeiro ano de um curso Técnico em Informática para Internet Integrado ao Ensino Médio, em que majoritariamente não possui conhecimento prévio em programação.
4. **Dinâmicas interacionista-problematizadoras:** As aulas de programação esportiva utilizaram desafios da OBI e exercícios do Beecrowd¹, organizados de forma cooperativa e sustentados por metodologias ativas, como o *Coding Dojo* e a Aprendizagem Baseada em Problemas (ABP). Essas dinâmicas envolveram o *Coding Dojo* nos formatos *Kata*, com resolução individual acompanhada de discussão coletiva, e *Randori*, realizado em duplas que alternavam os papéis de piloto e copiloto.
5. **Mediações pedagógicas distribuídas:** Resolução colaborativa de exercícios em duplas e mediação de conhecimento por meio de fóruns virtuais.
6. **Avaliação processual e cooperativa das aprendizagens:** Correção cooperativa de exercícios durante as aulas.
7. **Suporte da tecnologia digital:** O uso do Beecrowd é reconhecido como um juiz online de programação, com foco em competitividade e gamificação, facilitando a correção automática e o *feedback* da resolução de exercícios. Em paralelo ao uso da plataforma, também foi empregada a plataforma Moodle, como um suporte para disponibilização de materiais utilizados ao longo das aulas, bem como a criação de fóruns de dúvidas.

A Figura 1 apresenta um panorama geral das etapas da AP, evidenciando uma organização cíclica das etapas, na qual não se configuram totalmente como sequenciais.

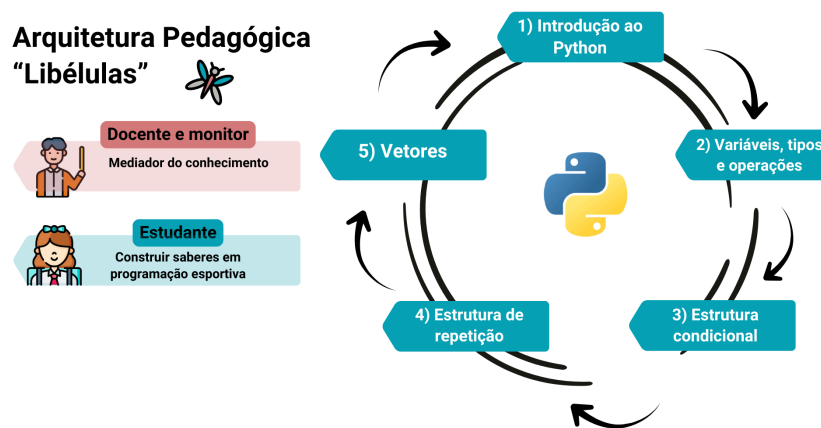


Figura 1. Etapas definidas para a AP. Fonte: Própria (2025).

Na **Etapa 1** (Introdução ao Python), apresentou-se o ambiente de desenvolvimento, a plataforma Beecrowd e conceitos iniciais de programação, como “Hello World”, além das estratégias de gamificação e do uso do Moodle como apoio às aulas. A **Etapa 2** (Variáveis, tipos e operações) abordou os tipos de dados, variáveis, operações aritméticas e uso de strings, destacando a relação entre os exercícios do

¹ <https://judge.beecrowd.com/>

Beecrowd e as questões da OBI. Na **Etapa 3** (Estruturas condicionais), foram estudados operadores lógicos e a estrutura *if/else*, com dedicação de mais aulas devido à complexidade desse conteúdo para iniciantes. A **Etapa 4** (Estruturas de repetição) introduziu os laços *for* e *while*, com ênfase na função *range()* e em aplicações práticas na resolução de problemas. Por fim, a **Etapa 5** (Vetores) apresentou essas estruturas de dados e sua implementação em Python, com atividades direcionadas à preparação para questões da OBI, incluindo simulados. Ao longo destas etapas, foram construídos artefatos educacionais para permitir a reprodutibilidade da AP desenvolvida com programação esportiva, sendo disponibilizados planos de aulas, slides, manual de um jogo conceitual e sala de aula virtual². Esse material foi publicizado para permitir a replicação da AP executada ao longo de um projeto de ensino, sobretudo com planos de aulas detalhados com objetivo, conteúdo e roteiro para cada aula.

A Tabela 1 apresenta os critérios de pontuação utilizados ao longo da gamificação da AP. Cada atividade realizada pelos estudantes gera uma quantidade específica de pontos, que varia conforme o desempenho: 100% (máxima), 50% ou 0%.

Atividade	Denominação	100%	50%	0%
1	Prova OBI (pontos/prova realizada)	5	0	0
2	Nota da Prova OBI (pontos obtidos)	15	7	0
3	Frequência (ponto/aula)	1	0	0
4	Lista de Exercícios (pontos/lista feita)	5	2,5	0
5	Desafios (pontos/desafio feito)	3	1	0
6	Dúvidas no fórum	1	0	0

Tabela 1. Critérios de pontuação no jogo conceitual. Fonte: Própria (2025).

Esses pontos são acumulados ao longo das aulas e organizados em um *ranking*, atualizado continuamente em uma plataforma online, a exemplo do ScoreLeader³. Esse *ranking* permite que os estudantes acompanhem sua própria evolução e seu desempenho, favorecendo o engajamento por meio de elementos gamificados como competição saudável. Idealmente, ao final da aplicação da AP, os melhores colocados devem ser premiados como forma de engajamento ao longo das aulas.

3.1. Coding Dojo e desafios de programação

Ao longo das etapas da AP, buscou-se realizar desafios de programação esportiva utilizando de metodologias colaborativas para o ensino de programação, como *Coding Dojo*. O *Coding Dojo* é uma metodologia ativa que permite o aperfeiçoamento em lógica de programação, possibilitando um aprendizado envolvendo imersão em desafios de forma colaborativa. Para isso, existem três formatos voltados à aplicação do método, sendo o *Randori*, *Kata* ou *Kake*, possuindo o mesmo objetivo final para aprendizagem colaborativa, mas com aplicabilidade diferente.

² <https://bit.ly/4ciYd4V>

³ <https://scoreleader.com/>

O modelo *Kata* acontece a partir de uma proposição de um desafio, que deve ser solucionado de forma individual a partir de uma plateia, com o aprimoramento do resultado de forma coletiva [Sousa et al., 2023]. O formato *Randori* apresenta a figura do piloto e copiloto para resolução de desafios de programação, sendo aplicado ao longo do processo de resolução. Já o método *Keke* ocorre por meio da divisão de grupos, a fim de resolver problemas, e, em seguida, o compartilhamento de respostas entre as equipes [Santos; Bezerra; Nunes, 2024].

Para a instância da AP, em cada uma das dez aulas de programação esportiva, foi selecionado um exercício em formato *Kata* ou *Randori*, de acordo com o repositório do Beecrowd e questões anteriores da OBI⁴. Para a realização desses desafios, também foi aplicada a regra de pontuação dos estudantes como parte dos elementos da gamificação, sendo uma estratégia para engajamento e motivação dos estudantes. Assim, ao longo das aulas, buscou-se realizar desafios imersivos em programação esportiva, com o uso de metodologias colaborativas e gamificação.

3.2. Contexto, planejamento e organização das aulas

O desenvolvimento de PC tem sido discutido como uma habilidade fundamental a ser desenvolvida na educação, sobretudo com a promulgação da Base Nacional Comum Curricular (BNCC) em 2022 evidenciando o ensino de Computação na Educação Básica [Junior and Rivera, 2024]. O PC é definido como competências para identificar e aplicar conceitos de Computação para resolução de problemas cotidianos, como o conceito de algoritmos, reconhecimento de padrões, decomposição e abstração [Wing, 2006].

Nesse contexto, o presente artigo apresenta uma AP voltada aos treinamentos para a Modalidade de Programação Nível 1 da OBI, por meio de um projeto de ensino executado no Campus Ceres do Instituto Federal Goiano (IF Goiano). A AP tem como público-alvo estudantes do primeiro ano de um curso Técnico de Informática para Internet Integrado ao Ensino Médio. A participação em aulas de programação esportiva envolve habilidades de PC, tal como a resolução de desafios com o uso de algoritmos em linguagens de programação, dividindo problemas complexos em etapas menores, identificando padrão e abstraindo os problemas de forma correta, favorecendo o desenvolvimento de competências ligadas ao PC [Oliveira et al. 2025].

Para a execução das etapas da AP, foram divididas as aulas correspondentes à dinâmica da arquitetura desenvolvida, sendo organizado a partir do conteúdo necessário para que os estudantes realizassem a prova da Modalidade de Programação, Nível 1. A divisão das aulas em relação às etapas foi considerando o nível de conhecimento dos estudantes e o conteúdo a ser abordado, sendo observado na Tabela 2.

As aulas da instância da AP foram organizadas com duração de 2h cada, em turno vespertino e em formato presencial, enquanto uma atividade extracurricular dos estudantes, voltada à edição de 2025 da OBI. Essas aulas foram conduzidas por monitores de graduação da área da Computação, dos períodos finais, com supervisão de docentes com experiência em olimpíadas de informática.

⁴ <https://olimpiada.ic.unicamp.br/passadas/>

Etapa	Descrição	Aula
1	Introdução ao Python	Aula 1
2	Variáveis, tipos e operações	Aula 2
3	Estrutura condicional e operações lógicas	Aula 3 e 4
4	Estrutura de repetição com For/While	Aula 5, 6 e 7
5	Vetores	Aula 8, 9 e 10

Tabela 2. Etapas da AP e as aulas. Fonte: Própria (2025).

Para a **Aula 1 (Hello World)**, foi um momento inaugural do projeto, apresentando a estrutura de organização e regras da OBI e o modo de condução das aulas, com o uso do Beecrowd e Moodle, bem como as regras do jogo conceitual. Para isso, dedicou um momento da aula para cadastro dos estudantes na plataforma do juiz online de programação, sendo uma ferramenta utilizada ao longo das aulas como um suporte de ensino. Além disso, também foi apresentado conceitos iniciais de programação, como variáveis, indentação, entradas/saídas e tipos de dados.

Durante a **Aula 2 (String, Split e Operações Lógicas)** apresentou o conceito de String e a manipulação por meio de funções do Python, como *split()*. Também foi apresentado o conceito de operações lógicas, com AND, OR e NOT. Ao longo da explicação, buscou-se estabelecer conexões com a realidade dos estudantes, utilizando elementos conhecidos da cultura *pop*, como *Harry Potter* e outras referências próximas ao seu cotidiano. A função *split()*, por exemplo, foi apresentada como uma “função mágica” que poderia separar textos, estratégia que ajudou a construir uma intuição mais leve e motivadora para os estudantes.

Enquanto que a **Aula 3 (Estruturas Condicionais - IF e ELSE - Parte 01)** introduziu o conceito de condicional por meio de abordagens motivadoras, também associando com elementos da cultura *pop*, como a construção do Chapéu Seletor em Python, em alusão ao mundo mágico de *Harry Potter*. Essas abordagens foram utilizadas para apresentar o conceito, posteriormente com a realização de exercícios do repositório do juiz online de programação (Beecrowd), como desafios de programação.

A **Aula 4 (Estruturas Condicionais - IF e ELSE - Parte 02)** aprofundou o conceito de estruturas de decisão por meio da ABP, com o uso de operadores lógicos e relacionais, na qual foram apresentadas a sintaxe e as funcionalidades, considerando a aplicação na resolução de problemas de programação. Para isso, foram utilizados exercícios do Beecrowd e o Moodle, em que se aplicou questões anteriores da OBI com apenas o conteúdo de estrutura condicional, tal qual o exercício “O Ogro” (OBI 2024). Esses exercícios foram desenvolvidos em formato imersivo de desafios, em que os estudantes deveriam resolver em tempo limitado, enquanto uma prática para olimpíadas.

Ao longo do início da **Aula 5 (Estruturas de Repetição - FOR e WHILE - Parte 01)**, foi proposta a apresentação do exercício “Var” (OBI 2023) enquanto uma atividade para a casa da Aula 4. Para a apresentação de conteúdo, foram introduzidas estruturas de controle para repetição, com o uso do *for* e *while*. Já a **Aula 06 (Estruturas de Repetição - Contadores, Acumuladores e Break - Parte 02)**

apresentou os conceitos de contadores e acumuladores, fundamentais na lógica de programação para auxiliar na resolução de problemas. Enquanto o contador é utilizado para registrar o número de repetições ou ocorrências, o acumulador serve para somar valores ao longo da execução do programa.

Para a **Aula 07 (Dinâmica com Prática de Programação Esportiva)** foi focado exclusivamente na resolução de problemas de forma imersiva e colaborativa, com a figura do piloto e copiloto. Diferentemente das aulas anteriores, que combinavam apresentação teórica seguida de exercícios no formato de *Coding Dojo*, esta aula foi dedicada exclusivamente à resolução de desafios, como “Torneio de Tênis” (OBI 2021), os problemas 1064 e 1045 (Beecrowd) e “Xadrez” (OBI 2018). Para isso, a turma foi dividida em duplas: com o piloto responsável por escrever o código de programa e outro colega por explicar (copiloto), sendo a explicação diante da sala de aula (plateia). Além disso, essa aula tratou majoritariamente de conteúdos relativos à estrutura de repetição.

A **Aula 08 (Vetores em Python)** introduziu o conceito das estruturas de dados vetores e as funções em Python para construção. Essa aula permitiu ao estudante compreender o conceito teórico e prático do assunto, a fim de facilitar a resolução da prova da OBI, com a apresentação de exercícios para resolução com os estudantes. Em relação à **Aula 09 (Aquecimento para Fase Local da OBI)** foi aplicado um simulado com duração de 2h com exercícios da OBI 2016, Nível 1. As questões contemplavam todos os conteúdos essenciais para a prova, incluindo tópicos como vetores e demais estruturas fundamentais da lógica de programação.

A **Aula 10 (Revisão para a OBI)** revisou todo o conteúdo aplicado ao longo das aulas, com o reforço do conteúdo apresentado anteriormente e a resolução de exercícios pelo docente e monitor, elencando elementos essenciais para a prova, como lista/vetores em Python. Tal como esperado, após a finalização das aulas planejadas, os estudantes encontravam-se aptos para participar da OBI, considerando os conceitos básicos necessários para uma possível classificação.

Embora as aulas da instância da AP seja em formato presencial, docente e monitores sempre destacaram o Moodle enquanto uma plataforma de suporte para o ensino, sendo publicados slides das aulas, desafios extras e videoaulas complementares para os estudantes. Outro fator da plataforma, foi o uso de fóruns de dúvidas para os estudantes, em que os próprios sanaram suas dúvidas, com a mediação do conhecimento por meio dessas ferramentas.

Além disso, também foram reservadas aulas extras conforme a classificação dos estudantes nas provas, sendo estas aulas não compreendidas como parte das etapas, mas sim do projeto de ensino, visto que o conteúdo mínimo para a resolução da prova foi finalizado ao longo das cinco etapas da AP. Essas aulas adicionais consistiram em revisar tópicos e contextualizar sobre conteúdos mais avançados, como Árvore, Algoritmos de Ordenação, Grafos e outros assuntos mais específicos da Fase Nacional [OBI 2025].

4. Cuidados éticos

O presente trabalho, embora não tenha sido submetido ao Comitê de Ética em Pesquisa (CEP), buscou ser construído a partir de princípios éticos. A participação de todos

estudantes foi voluntária, com a explicação dos objetivos e o direito de recusa ou desistência a qualquer momento, sem prejuízo. As informações obtidas foram coletadas e tratadas de forma anônima, garantindo a impossibilidade de identificação durante as etapas de pesquisa.

Ao longo da aplicação da pesquisa, sendo uma questão aberta para os estudantes participantes da AP, foi utilizado o *Google Forms* para a coleta de dados, com observação da política de privacidade da plataforma⁵. Além disso, durante o armazenamento e o tratamento dos dados, as respostas dos estudantes foram armazenadas considerando anonimização e confiabilidade dos dados, com o uso do *Google Drive* de contas institucionais.

5. Resultados e discussões

Durante as aulas, foi observado o engajamento dos estudantes por meio de dúvidas e o contato colaborativo entre os participantes do projeto de ensino, principalmente por meio das abordagens de desafios imersivos de programação esportiva, com o uso de elementos de gamificação. Esses desafios permitiam a discussão do conteúdo de maneira dinâmica, abrindo espaço para um ambiente de troca de conhecimentos em programação esportiva, com o desenvolvimento de pensamento algorítmico. Ademais, a participação das aulas de programação esportiva e a realização das Fases da OBI, permitiu a construção de habilidades interpessoais, como solução de problemas em tempo limitado, comunicação e divisão de tarefas.

Os estudantes que participaram da instância da AP realizaram as provas das Fases da OBI na edição de 2025, com desempenho e classificação satisfatória dos participantes. Ressalta-se, inclusive, que a presente instância da AP alcançou a maior classificação de estudantes na Fase Nacional entre os dez anos de treinamentos no Campus Ceres do IF Goiano, sugerindo uma possível associação com a proposta pedagógica da arquitetura. Com a perspectiva de avaliação sobre a instância da AP, foi desenvolvido uma questão aberta sobre momentos marcantes para que os discentes pudessem relatar sua visão sobre as aulas.

Para análise da percepção dos estudantes em relação à AP, foi aplicado um questionário com uma questão aberta com 63 estudantes participantes do projeto de ensino, com o intuito de permitir a avaliação da experiência. Para isso, as respostas abertas foram avaliadas por meio da classificação de sentimento e nuvens de palavras, diante da seguinte pergunta: “*O que foi mais marcante para você nas aulas do projeto?*”.

Para a questão aberta, foi aplicada a classificação de sentimento e nuvens de palavras em relação à percepção sobre momentos marcantes para uma instância da AP, considerando o uso de análise de sentimento para questionário em pesquisas voltadas à educação [Desidério et al., 2025]. A realização da classificação de sentimento, foi utilizado a abordagem desenvolvida por [Macedo et al, 2023] sendo as respostas classificadas automaticamente pelo ChatGPT4⁶ (plano gratuito em outubro de 2025, modelo GPT-4), utilizando de classificação por meio de pontos flutuantes. Assim, foram

⁵ <https://policies.google.com/privacy>

⁶ <https://openai.com/>

definidos valores que representam sentimentos positivos, negativos e neutros. Dessa forma, os comentários positivos como aqueles classificados entre 0.5 e 1, neutros aqueles classificados entre -0.5 e 0.5 e negativos aqueles classificados entre -1 e -0.5. O *prompt* pode ser observado na Figura 2.

prompt único para o GPT

prompt: Você é um classificador de texto, e está classificando os sentimentos contidos nesses textos entre positivo, negativo e neutro. Classifique os textos abaixo levando em consideração esses sentimentos (positivo, negativo e neutro) classificando entre -1 e 1 no formato de ponto flutuante. Onde -1 significa totalmente negativo e 1 completamente positivo. Retorne em formato de tabela.

Figura 2. *Prompt* único de classificação de sentimentos. Fonte: Própria (2025).

Após a aplicação desse *prompt* e a definição dos valores de sentimentos para as respostas da questão aberta, foi realizada a geração dos gráficos por meio do *Google Sheets* (Figura 3). A classificação de sentimentos em relação à questão sobre momentos marcantes dos estudantes ao participar das aulas, 88,9% (56 estudantes) dos comentários foram positivos, sendo 7,9% neutros (5 estudantes) e apenas 3,2% (2 estudantes) como negativos. Essas respostas classificaram os momentos marcantes enquanto positivo, destacando os slides produzidos, o uso do Beecrowd e as metodologias colaborativas aplicadas ao longo das aulas.

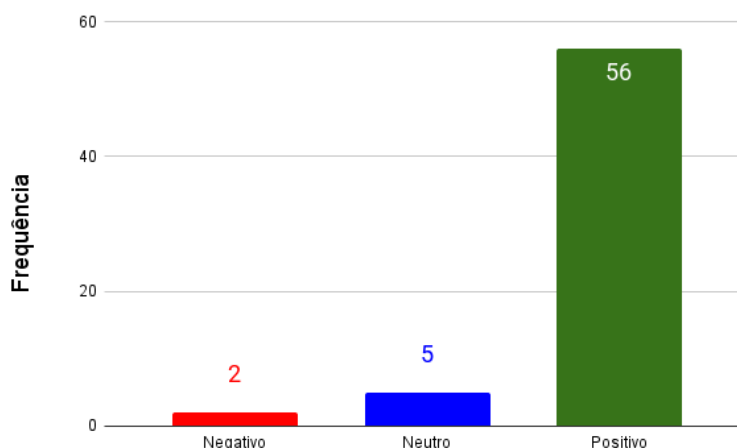


Figura 3. Classificação de sentimentos acerca de momentos marcantes. Fonte: Própria (2025).

Para a análise qualitativa das respostas abertas fornecidas pelos estudantes, foram construídas nuvens de palavras com o objetivo de identificar termos recorrentes e padrões lexicais relacionados às percepções sobre as aulas da instância da AP [Oliveira et al., 2023]. A geração das nuvens foi realizada com *Google Colab*, utilizando a linguagem Python e as bibliotecas *pandas*, *wordcloud*, *matplotlib* e *nlTK*. Sendo assim, observa-se a nuvem de palavras a seguir (Figura 4).

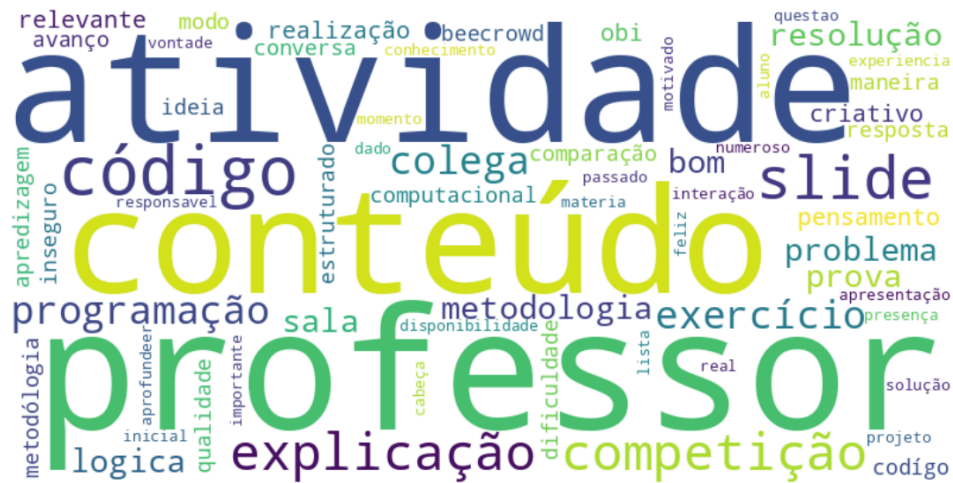


Figura 4. Nuvens de palavras de momentos marcantes. Fonte: Própria (2025).

Assim é possível observar a nuvem de palavras gerada, com intuito de descrever momentos marcantes durante as aulas da AP. Entre termos destaques para momentos marcantes, nota-se palavras como “Atividade”, “Conteúdo”, “Professor”, “Explicação” e “Slide”, em que os estudantes destacaram a importância das atividades e a forma de organização com metodologias ativas, com o uso de elementos do cotidiano apresentados anteriormente. Nesse sentido, observa-se a presença de elementos de gamificação e métodos colaborativos, sendo observado seu uso ao longo de todas as dez aulas, como a discussão de caminhos para a resolução de problemas lógicos-matemáticos de questões anteriores da OBI.

Além disso, é possível observar outras palavras em relação ao contexto das aulas, como “Criativo”, “Qualidade” e “Aprendizagem”, ressaltando o modo em que os treinamentos foram conduzidos, a exemplo da Aula 03 e 04. Essas aulas utilizaram abordagens para levar o ensino de condicionais por meio de elementos da cultura *pop*, tornando a aprendizagem mais próxima da realidade dos estudantes, motivando a presença e participação durante as aulas do projeto de ensino.

Outros termos destacados foram “Código”, “Competição” e “Beecrowd”, evidenciando a maneira que as aulas foram conduzidas, com a divisão de explicação de conteúdo e realização de exercícios no juiz online de programação (Beecrowd). Também pode ser colocado à tona a importância do jogo conceitual desenvolvido e aplicado por meio de desafios de programação. Esses desafios foram conduzidos por meio de metodologias colaborativas, como o *Coding Dojo*, em que os estudantes negociavam entre si, com uma discussão colaborativa para alternativas de resolução de problemas em programação esportiva.

Em relação às aulas, destaca-se a importância da Aula 07 enquanto uma imersão em programação esportiva, com a dedicação da aula exclusivamente para resolução de desafios com *Coding Dojo*, com formato *Randori*, sendo o papel de codificação (piloto) e explicação do código para a sala de aula (copiloto), com rotatividade das posições. Outrossim, observa-se outras dinâmicas para as aulas, como a aplicação de simulados das provas da OBI durante a Aula 09, como uma estratégia de treinamento.

6. Considerações finais

A AP “Libélulas” trouxe à tona uma proposta pedagógica para o ensino de programação esportiva, fundamentada no uso de metodologias ativas e estratégias de gamificação, com seus artefatos educacionais disponibilizados publicamente para acesso e reprodução. Para a organização das aulas, foram utilizadas abordagens colaborativas para a programação, como o *Coding Dojo*. Também foi adotado elementos cotidianos dos estudantes, sendo uma possível estratégia para tornar as aulas mais atrativas e motivadoras para os estudantes.

Diante da experiência de sua instanciamento, a avaliação de relatos dos participantes demonstrou um sentimento positivo em relação aos momentos marcantes, com desafios conduzidos por meio de gamificação e métodos ativos. Essa avaliação com o uso da análise de sentimento e nuvens de palavras sugeriu uma possível relação entre essa abordagem e momentos marcantes.

Apesar disso, a presente AP foi instanciada considerando apenas uma turma no ano de 2025 em uma instituição no âmbito de ensino de programação esportiva, sendo necessário mais instâncias para a generalização dos resultados. Ademais, o público-alvo da AP foram apenas em primeiro com programação e olimpíadas de informática, sendo a construção de proposta considerando esse contexto para o ensino de programação esportiva. Como trabalho futuro, espera-se ampliar a instância com outros Níveis da OBI, com envolvimento de mais turmas e instituições de ensino, avaliando a percepção dos estudantes, com mais questões abertas e o uso de instrumentos validados.

7. Agradecimentos

Agradecemos ao Núcleo de Estudos e Pesquisa em Tecnologia da Informação (NEPeTI) e ao Campus Ceres do IF Goiano pelo apoio na realização do trabalho.

Uso de Inteligência Artificial

Para a redação e organização do artigo, não foi utilizada nenhuma Inteligência Artificial Generativa, prezando pela autoria e originalidade das pessoas autoras da pesquisa. O uso de modelos generativos foi restrito à classificação de sentimentos da seção de Resultados e Discussões, com relato detalhado com base em outras pesquisas.

Referências

- Alves, M., Santana, T., & Braga, A. (2025). Libélulas: Arquitetura Pedagógica Gamificada para o Ensino de Programação Esportiva na Educação Básica. In Anais da XIII Escola Regional de Informática de Goiás, (pp. 234-243). Porto Alegre: SBC.
- Carvalho, M. J. S., Nevado, R. A., & Menezes, C. S. (2005). Arquiteturas pedagógicas para educação a distância: Concepções e suporte telemático. In Simpósio Brasileiro de Informática na Educação (SBIE) (pp. 351–360). SBC.
- Czmola, H., Grebogy, E., Petroski, F., & Marquette, F. (2025). A Computação como Componente Curricular nos anos iniciais do Ensino Fundamental: Discussões sobre a BNCC Computação e o Quadro Organizador Municipal. In Anais do XXXVI Simpósio Brasileiro de Informática na Educação, (pp. 773-784). Porto Alegre: SBC.

- Desidério, S., Azevedo, M., Viana, M., Nunes, A., Siqueira, M., Torres, Y., Cristina, J., Barros, J., Pinheiro, V., & Marques, A. (2025). Oficinas de Pensamento Computacional: explorando a experiência de estudantes sob a perspectiva de gênero. In *Anais do XIX Women in Information Technology*, (pp. 309-319). Porto Alegre: SBC.
- Farias, C. M. de, Azevedo, F. P., & Dias, J. E. de J. (2018). Uma abordagem gamificada para o ensino de lógica de programação: Relato de experiência. In *Anais do XXVI Workshop sobre Educação em Computação*. Porto Alegre, RS, Brasil: SBC.
- Freire, P. (1975). *Educação como prática da liberdade* (1. ed.). Rio de Janeiro: Paz e Terra.
- Garcia, F., & Oliveira, S. (2022). Aplicação de um plano de ensino para disciplina de algoritmos com metodologias ativas: Um relato de estudo de caso piloto. In *Anais do XXXIII Simpósio Brasileiro de Informática na Educação* (pp. 301–310). Porto Alegre, RS, Brasil: SBC.
- Gomes, A. S., & Gomes, C. R. A. (2020). Classificação dos tipos de pesquisa em informática na educação. In P. A. Jaques, M. Pimentel, S. Siqueira & I. Bittencourt (Eds.), *Metodologia de Pesquisa Científica em Informática na Educação: Concepção de Pesquisa* (Série Metodologia de Pesquisa em Informática na Educação, v. 1). Sociedade Brasileira de Computação.
- Junior, A., & Rivera, J. (2024). BNCC Computação: O que os acadêmicos de licenciatura precisam saber sobre o Pensamento Computacional?. In *Anais do XXXII Workshop sobre Educação em Computação*, (pp. 878-891). Porto Alegre: SBC.
- Irion, C., Santos, C., Araujo, R., & Pereira, J. H. (2024). Desenvolvimento de competências docentes em computação para fomento da programação competitiva: Um relato da disciplina de estágio em docência. In *Anais do XXX Workshop de Informática na Escola* (pp. 646–655). Porto Alegre, RS, Brasil: SBC.
- Lima, G., Costa, J., Araujo, R., & Dorça, F. (2024). Arquitetura pedagógica apoiada por IA para o desenvolvimento da aprendizagem autorregulada em estudantes. In *Anais do XXXV Simpósio Brasileiro de Informática na Educação* (pp. 2999–3008). Porto Alegre, RS, Brasil: SBC.
- Macedo, M., Moraes, B., Ribeiro, L., Costa, N., Santana, T., & Pereira Junior, C. (2023). Análise comparativa de vídeos do YouTube de canais de homens e mulheres na área de computação: uma investigação a partir dos comentários. In *Anais do XXXIV Simpósio Brasileiro de Informática na Educação*, (pp. 1862-1873). Porto Alegre: SBC.
- Menezes, C. S. de, Junior, A. N. C., & Aragón, R. (2021). Arquiteturas pedagógicas para aprendizagem em rede. In *Série de livros texto da CEIE-SBC* (pp. 7, 13, 14, 16, 34). [s.n.]. Acesso em: 16 abr. 2023.
- Mocelin, R. R., & Fiuza, P. J. (2021). Tecnologias digitais e arquiteturas pedagógicas na educação. In *Tecnologia da Informação e Comunicação: Pesquisas em Inovações Tecnológicas* (pp. 316–326). Guarujá, SP, Brasil: SBC.
- Nunes, J., Escalante, L., Silva, L., & Penze, L. (2024). A primeira maratona feminina

- de programação do Brasil: Motivações para o desenvolvimento do projeto e relatos da primeira edição do evento. In *Anais do XVIII Women in Information Technology*. Porto Alegre, RS, Brasil: SBC.
- OBI. (2025). Olimpíada Brasileira de Informática, <https://olimpiada.ic.unicamp.br/>.
- Oliveira, G., Ribeiro, R., Ferreira, N., Abreu, M., & Dusi, C. (2025). Ensinando Python com LEGO™ Spike Prime: Uma Abordagem Prática para Jovens de 13 a 18 Anos. In *Anais do V Simpósio Brasileiro de Educação em Computação*, (pp. 278-287). Porto Alegre: SBC.
- Piaget, J. (1985). *O possível e o necessário: evolução dos possíveis na criança* (1. ed.). Porto Alegre: Artes Médicas.
- Portilho, F., Amaral, J., Rodrigues, N., Junior, C. P., & Costa, N. (2024). Uma arquitetura pedagógica para o ensino de lógica de programação: Lições aprendidas a partir de um projeto de extensão. In *Anais do XXXII Workshop sobre Educação em Computação* (pp. 329–340). Porto Alegre, RS, Brasil: SBC.
- Rezende, F., Dullius, M., & Schorr, M. (2024). Arquitetura pedagógica Amarelinha para o desenvolvimento do pensamento computacional. In *Anais do I Simpósio Brasileiro de Computação na Educação Básica* (pp. 70–74). Porto Alegre, RS, Brasil: SBC.
- Santos, B., & Machado, L. (2024). Arquitetura pedagógica para alfabetização de crianças com transtorno do espectro autista: Uma proposta baseada na gamificação inclusiva. In *Anais do I Workshop de Informática na Educação Inclusiva* (pp. 146–155). Porto Alegre, RS, Brasil: SBC.
- Santos, J., Bezerra, F., & Nunes, I. (2024). Revisão Sistemática da Literatura sobre a Utilização do Coding Dojo como Prática de Ensino em Programação. In *Anais do XXXV Simpósio Brasileiro de Informática na Educação*, (pp. 2469-2482). Porto Alegre: SBC. doi:10.5753/sbie.2024.242555
- Sousa, J., Maranhão, D., Borges, P., & Neto, C. (2023). Uma análise da integração da metodologia ativa Coding Dojo a uma plataforma de ensino-aprendizagem de algoritmos. In *Anais do XXXIV Simpósio Brasileiro de Informática na Educação*, (pp. 127-138). Porto Alegre: SBC.
- Tavares, L. O., Menezes, C. S., & Nevado, R. A. (2012). Pedagogical architectures to support the process of teaching and learning of computer programming. *Education Conference Proceedings*, 1–6.
- Tavares, O., Menezes, C., Aragón, R., & Costa, L. (2013). Uma arquitetura pedagógica auxiliada por tecnologias para ensino e aprendizagem de programação. In *Anais do XXI Workshop sobre Educação em Computação* (pp. 778–787). Porto Alegre, RS, Brasil: SBC.
- Wing, Jeannette M. Computational thinking. *Communications of the ACM* 49.3 (2006): 33-35.