



DOI: <https://doi.org/10.5281/zenodo.17408101>

**APRENDIZAGEM BASEADA EM PROBLEMAS (ABP) E PENSAMENTO
COMPUTACIONAL: UMA APLICAÇÃO COM PYTHON E GOOGLE COLAB EM
DISCIPLINAS DE INTRODUÇÃO À PROGRAMAÇÃO**

**PROBLEM-BASED LEARNING (PBL) AND COMPUTATIONAL THINKING: AN
APPLICATION WITH PYTHON AND GOOGLE COLAB IN INTRODUCTORY
PROGRAMMING COURSES**

Nisstton Moraes Tavares de Melo¹

Orcid: <https://orcid.org/0000-0003-3850-1786>

Karoline Lira Dantas da Costa²

Orcid: <https://orcid.org/0009-0005-5856-7530>

RESUMO

A Aprendizagem Baseada em Problemas (ABP) configura-se como uma metodologia ativa eficaz para promover a autonomia e o pensamento crítico dos estudantes. Este artigo analisa a aplicação da ABP integrada às ferramentas Python e Google Colab em uma disciplina introdutória de programação – Introdução à Ciência da Computação (ICC) da Universidade Federal de Campina Grande (UFCG). O estudo, de natureza aplicada e com abordagem qualitativa, envolveu 182 alunos, utilizando um ciclo de ABP que incluiu a apresentação de problemas contextualizados, trabalho em grupo, pesquisa e desenvolvimento de soluções no Google Colab. A coleta de dados combinou pré e pós-testes, análise de notebooks e questionários de percepção. Os resultados demonstram um aumento significativo na proficiência em programação, com a média de acertos evoluindo de 58% para 76%, além de maior engajamento, desenvolvimento do pensamento computacional e valorização do aprendizado colaborativo. A experiência evidenciou a efetividade da ABP, apoiada por ferramentas interativas, para criar uma aprendizagem significativa e alinhada às demandas da formação em computação, ainda que desafios como a adaptação inicial ao método

¹Análise e Desenvolvimento de Sistema - Doutor. Professor Aux. I Faculdade Estácio da Paraíba. E-mail: 80462570444@professores.estacio.br

²Mestrado em Engenharia Biomédica. Graduada em Processamento de Dados. Docente da Faculdade Estácio da Paraíba. E-mail: karoline.costa@estacio.br

e à ferramenta tenham sido identificados. Conclui-se que a abordagem é uma estratégia pedagógica promissora para o ensino de programação.

Palavras-chave: Pensamento Computacional; Programação; Ensino Superior; Aprendizagem Baseada em Problemas.

ABSTRACT

Problem-Based Learning (PBL) is an effective active methodology for promoting student autonomy and critical thinking. This article analyzes the application of PBL integrated with Python and Google Colab tools in an introductory programming course (Introduction to Computer Science) at the Federal University of Campina Grande (UFCG). The applied study, with a qualitative approach, involved 182 students and utilized a PBL cycle that included the presentation of contextualized problems, group work, research, and solution development in Google Colab. Data collection combined pre- and post-tests, notebook analysis, and perception questionnaires. The results show a significant increase in programming proficiency, with the average score rising from 58% to 76%, in addition to greater engagement, development of computational thinking, and appreciation for collaborative learning. The experience demonstrated the effectiveness of PBL, supported by interactive tools, in creating meaningful learning aligned with the demands of computer education, although challenges such as initial adaptation to the method and the tool were identified. It is concluded that the approach is a promising pedagogical strategy for teaching programming.

Keywords: Computational Thinking, Programming, Higher Education, Problem-Based Learning.

INTRODUÇÃO

A Aprendizagem Baseada em Problemas (ABP) é uma metodologia ativa de ensino que coloca o estudante no centro do processo de aprendizagem, promovendo a construção do conhecimento a partir da resolução de problemas reais ou simulados. Diferente do modelo tradicional, centrado na transmissão de conteúdo pelo professor, a ABP estimula a autonomia, o pensamento crítico e a aprendizagem colaborativa, ao desafiar os alunos a investigar, analisar e propor soluções fundamentadas.

Originada na década de 1960, na Faculdade de Medicina da Universidade de McMaster (Canadá), a ABP foi inicialmente concebida para aprimorar o raciocínio clínico dos estudantes, mas ao longo do tempo passou a ser aplicada em diversas áreas do conhecimento, incluindo engenharias, ciências da computação e educação, pela sua capacidade de integrar teoria e prática (Savery; Duffy, 1995; Barrows, 1996).

Segundo Barrows (1996), a ABP baseia-se em um ciclo que começa com a apresentação de um problema desafiador e contextualizado, seguido por discussões em grupo, levantamento de hipóteses, pesquisa orientada e síntese de soluções. Nesse processo, o papel do professor se transforma: ele atua como facilitador e mediador da aprendizagem, estimulando a reflexão e o desenvolvimento de competências cognitivas, sociais e metacognitivas.

Já para o autor Abdull Jabbar e Abd Halim (2024), apresentam em seus estudos “o entendimento dos estudantes em programação Python aumentou significativamente, de média de 57 % para 74,5 %” após a intervenção com ABP integrada à tecnologia. A abordagem de aprendizagem baseada em problema “pode melhorar o engajamento dos estudantes com o material e a aprendizagem ativa em comparação com métodos tradicionais centrados no docente”. Este estudo investiga os efeitos da aprendizagem baseada em problemas (ABP) integrada com tecnologia em disciplinas de ciência da computação, focando especialmente no aprendizado de programação em Python por estudantes de ensino médio em uma escola da Malásia. Utilizando o modelo “*PBL Gold Standard*” com sete componentes (conteúdo significativo, necessidade de saber, pergunta condutora, voz e escolha dos estudantes, habilidades do século XXI, criatividade, reflexão), os autores aplicaram um pré-teste e pós-teste com 30 alunos.

Os resultados demonstraram um aumento significativo na média de desempenho dos estudantes (de 57 % para 74,5 %) após a intervenção, o que indica que a abordagem ABP com tecnologia teve efeito positivo na aquisição de habilidades de programação. Os autores também destacam que o envolvimento ativo e a aprendizagem centrada no aluno aparecem como fatores importantes para o sucesso, embora reconheçam limitações quanto à generalização dos resultados devido à amostra única.

A partir dessa evidência, os autores sugerem que a ABP não apenas contribui para a aquisição de habilidades técnicas, mas também promove a autonomia dos estudantes, ao fazê-los escolher e conduzir projetos reais, o que reforça sua motivação e capacidade para resolver problemas práticos (Abdull Jabbar; Abd Halim, 2024).

Em ambientes de ensino de programação e pensamento computacional, a ABP tem se mostrado eficaz por favorecer a aprendizagem significativa (Ausubel, 2003) e a aplicação prática de conceitos abstratos. Ao resolver problemas autênticos utilizando linguagens como Python³ e

³ <https://www.python.org>

plataformas colaborativas como o Google Colab⁴, os estudantes desenvolvem não apenas habilidades técnicas, mas também competências de análise, modelagem, trabalho em equipe e comunicação — fundamentais para o perfil do profissional contemporâneo da área de tecnologia. A ABP é uma abordagem pedagógica que promove a construção ativa do conhecimento, partindo de situações-problema que refletem desafios reais ou simulados (Barrows, 1996). Diferentemente do modelo tradicional de ensino, centrado no professor, a ABP enfatiza a autonomia do estudante, a investigação colaborativa e a aplicação prática de conceitos teóricos. O problema apresentado serve como gatilho para que o estudante identifique lacunas no seu conhecimento e busque informações, promovendo aprendizagem significativa (Ausubel, 2003).

Barrows (1996) descreve a ABP como um processo cíclico que envolve várias etapas: apresentação do problema, discussão em grupo, levantamento de hipóteses, pesquisa autônoma, síntese e apresentação da solução. O papel do docente, neste contexto, é atuar como facilitador, orientando a construção do conhecimento e incentivando a reflexão crítica. Já Savery e Duffy (1995) enfatizam que a ABP integra competências cognitivas, sociais e metacognitivas, permitindo que os alunos desenvolvam habilidades de resolução de problemas, pensamento crítico e colaboração, fundamentais para sua formação acadêmica e profissional.

Na área de computação e ensino de programação, a ABP tem se mostrado particularmente eficaz. Estudos recentes mostram que a aplicação de ABP em cursos de Python ou pensamento computacional permite que os estudantes enfrentem problemas reais, desenvolvam algoritmos e testem soluções de forma colaborativa (Abdull Jabbar; Abd Halim, 2024). Além disso, a utilização de ferramentas digitais como Google Colab e Jupyter Notebooks potencializa a aprendizagem ao possibilitar ambientes interativos de programação, compartilhamento de código e experimentação em tempo real, reforçando a relação entre teoria e prática.

Autores contemporâneos também destacam que a ABP favorece a motivação intrínseca e a construção de competências do século XXI, como comunicação, criatividade e pensamento crítico (Boud; Feletti, 1997; Kyungbin *et al.*, 2021). Assim, a implementação da ABP em disciplinas introdutórias de programação não apenas melhora a compreensão conceitual, mas também prepara os estudantes para enfrentar problemas complexos de maneira estruturada, colaborativa e ética, aspectos essenciais para o desenvolvimento profissional na era digital.

⁴ <https://colab.research.google.com>

Este estudo tem como objetivo principal analisar os impactos da aplicação da ABP com o uso do Python e do Google Colab no desenvolvimento do pensamento computacional e na aprendizagem de programação por estudantes de uma disciplina de ICC da UFCG. Como objetivos específicos, buscou-se: (1) avaliar a evolução do desempenho discente por meio de pré e pós-testes; (2) investigar o engajamento e a autonomia dos estudantes no ambiente colaborativo do Google Colab; e (3) identificar as contribuições e os desafios da ABP na formação inicial em programação.

METODOLOGIA

Este estudo adotou uma abordagem qualitativa e aplicada, utilizando a ABP como metodologia ativa para o ensino de programação em Python. A intervenção foi realizada com estudantes da disciplina de ICC da UFCG, durante o semestre letivo de 2024.2.

A pesquisa contou com a participação de 182 alunos de graduação, matriculados na disciplina de ICC, sem experiência prévia em programação. Os procedimentos foram aplicados seguindo o ciclo proposto por Barrows (1996), organizado em etapas:

- **Apresentação do Problema:** Os estudantes receberam problemas contextualizados e desafiadores, relacionados a aplicações reais da computação (ex.: análise de dados, automação de tarefas, resolução de problemas matemáticos).
- **Discussão em Grupo:** Formação de equipes de 3 a 4 alunos para discussão e levantamento de hipóteses iniciais.
- **Pesquisa e Investigação:** Utilização de recursos como Google Colab para experimentação com código Python, pesquisa em documentação e materiais de apoio.
- **Desenvolvimento da Solução:** Implementação colaborativa das soluções no Google Colab, com acompanhamento do professor como facilitador.
- **Apresentação e Reflexão:** Cada grupo apresentou sua solução, seguida de debate coletivo sobre as abordagens adotadas.

As ferramentas e ambiente utilizados foram: Linguagem: Python, Plataforma: Google Colab e Recursos Adicionais: Material de apoio, videoaulas curtas e fóruns de discussão no

ambiente virtual da disciplina. Na coleta de dados, foram utilizados os seguintes instrumentos para avaliação:

- Pré-teste e pós-teste sobre conceitos básicos de programação e pensamento computacional.
- Análise qualitativa dos notebooks produzidos no Google Colab.
- Questionário de percepção dos estudantes sobre a metodologia.

Os dados quantitativos foram analisados por meio de estatística descritiva, enquanto os qualitativos foram examinados com base na categorização de respostas e na observação do engajamento e da evolução dos estudantes.

RESULTADOS E DISCUSSÃO

A comparação entre pré-teste e pós-teste revelou uma evolução significativa na compreensão dos conceitos fundamentais de programação. A média geral de acertos aumentou de 58% para 76%, indicando que a abordagem de ABP com Python e Google Colab contribuiu substantivamente para a aquisição de habilidades de programação. Esse resultado corrobora os achados de Abdull Jabbar e Abd Halim (2024), que observaram melhora semelhante (de 57% para 74,5%) em contexto análogo.

Os registros de acesso e interação no Google Colab mostraram um aumento de 40% na taxa de submissão de atividades em comparação com semestres anteriores que utilizaram metodologias tradicionais. Além disso, os estudantes demonstraram maior autonomia na resolução de problemas, recorrendo a fóruns, documentação e experimentação prática para superar desafios. Esse comportamento reforça a premissa de que a ABP promove aprendizagem significativa (Ausubel, 2003) e motivação intrínseca.

A análise dos notebooks produzidos evidenciou progresso na aplicação de conceitos como decomposição de problemas, reconhecimento de padrões e abstração. Inicialmente, a maioria dos alunos apresentava dificuldades na estruturação lógica dos algoritmos. Ao final da intervenção, notou-se maior clareza na modelagem de soluções e no uso de estruturas como funções e laços — elementos centrais do pensamento computacional.

Os questionários de percepção apontaram que 85% dos estudantes consideraram as discussões em grupo e o uso do Google Colab como fatores determinantes para o aprendizado. A

possibilidade de compartilhar código, revisar soluções dos colegas e receber feedback imediato potencializou a experiência de aprendizagem colaborativa.

Apesar dos resultados positivos, identificou-se que alguns alunos tiveram dificuldades iniciais com a curva de aprendizado da ferramenta e com a transição para um modelo menos diretivo. Esse achado ressalta a importância do papel do professor como facilitador e da necessidade de scaffolding pedagógico, especialmente em fases introdutórias.

Os resultados obtidos estão em sintonia com a literatura sobre ABP em computação, reforçando sua eficácia para integrar teoria e prática, desenvolver competências socioemocionais e preparar os estudantes para cenários reais de atuação profissional. A combinação entre ABP, Python e Google Colab mostrou-se uma estratégia pedagógica promissora para disciplinas introdutórias de programação.

CONSIDERAÇÕES FINAIS

Este estudo investigou a aplicação da ABP com o uso do Python e do Google Colab em uma disciplina introdutória de programação. Os resultados demonstram que a abordagem contribuiu significativamente para a melhoria do desempenho acadêmico, com um aumento médio de 58% para 76% nos escores de avaliação, além de promover o engajamento ativo e o desenvolvimento do pensamento computacional entre os discentes.

A integração entre a ABP e ferramentas colaborativas como o Google Colab mostrou-se eficaz para aproximar a teoria da prática, permitindo que os estudantes vivenciem ciclos completos de resolução de problemas — da análise à implementação — em um ambiente interativo e acessível. A experiência reforçou a premissa de que metodologias ativas favorecem não apenas a aquisição de conhecimentos técnicos, mas também competências transversais como trabalho em equipe, autonomia e comunicação.

Como limitações, destaca-se a necessidade de um período de adaptação tanto dos discentes quanto do docente ao modelo de ABP, especialmente em turmas com pouca familiaridade com aprendizagem autodirigida. Além disso, a dependência de conexão estável à internet e o domínio inicial das ferramentas digitais representaram desafios adicionais.

Para trabalhos futuros, sugere-se a expansão da aplicação para outras disciplinas e instituições, bem como a realização de estudos longitudinais que avaliem a retenção de

aprendizagem e o impacto da ABP na formação profissional a longo prazo. A incorporação de inteligência artificial como recurso de apoio personalizado também se mostra como um caminho promissor a ser explorado.

Por fim, reafirma-se o potencial da ABP como estratégia pedagógica inovadora e alinhada às demandas da educação no século XXI, capaz de preparar os estudantes não apenas para programar, mas para pensar computacionalmente e resolver problemas de forma criativa e colaborativa.

REFERÊNCIAS

ABDULL JABBAR, R.; ABD HALIM, N. D. The impact of project-based learning through integrating the use of technology in computer science courses on students' acquisition of programming skills. **Innovative Teaching and Learning Journal**, v. 8, n. 1, p. 1-14, 2024. DOI: 10.11113/itlj.v8.152. Acesso em 23 de abril de 2025.

AUSUBEL, D. P. **The acquisition and retention of knowledge: A cognitive view**. Boston: Kluwer Academic, 2003.

BARROWS, H. S. Problem-based learning in medicine and beyond: A brief overview. **New Directions for Teaching and Learning**, v. 1996, n. 68, p. 3-12, 1996. Disponível em: <https://onlinelibrary.wiley.com/doi/epdf/10.1002/tl.37219966804> Acesso em 23 de abril de 2025.

BOUD, D.; FELETTI, G. **The challenge of problem-based learning**. London: Kogan Page, 1997.

KYUNGBIN, A. T. K.; OTTENBREIT-LEFTWICH, A.; BRUSH, J. M.; GE, Y. Integration of problem-based learning in elementary computer science education. **Educational Technology Research and Development**, v. 69, n. 5, p. 2761-2787, 2021. Disponível em: <https://www.jstor.org/stable/27285839> Acesso em 23 de abril de 2025.

SAVERY, J. R.; DUFFY, T. M. Problem-based learning: An instructional model and its constructivist framework. **Educational Technology**, v. 35, n. 5, p. 31-38, 1995. Disponível em: <https://www.semanticscholar.org/paper/Problem-Based-Learning%3A-An-instructional-model-and-Savery-Duffy/d5d150c7126c0585cebc6888b2a00aaa4f232bac> Acesso em 23 de abril de 2025.