

EDITAL DE INSCRIÇÃO ( RESUMO EXPANDIDO) - ANÁLISE E  
DESENVOLVIMENTO DE SISTEMAS

**A EVOLUÇÃO DO ENSINO DE PROGRAMAÇÃO: UMA ANÁLISE  
COMPARATIVA ENTRE ABORDAGENS TEÓRICAS E PRÁTICAS**

*Lucas Oliveira Souza (lucas.olisouza03@gmail.com)*

*Caio Kenzo Mishima (caiok693@gmail.com)*

*Danilo Ferreira Simão (danilosimao68@gmail.com)*

*Tiago Misawa Calixto (tiago.calixto@metodista.br)*

O presente resumo expandido tem como objetivo sintetizar e elaborar sobre as análises contidas no projeto acadêmico intitulado “A Evolução do Ensino de Programação: Uma Análise Comparativa entre Abordagens Teóricas e Práticas”. O trabalho parte do pressuposto de que o ensino de programação, pilar fundamental na formação em Tecnologia da Informação e áreas afins, não é um campo monolítico. Pelo contrário, ele é composto por uma variedade de metodologias e abordagens pedagógicas que podem ser categorizadas em dois eixos principais: um de caráter científico e teórico, e outro de natureza prática e utilitária. O estudo empreende uma análise comparativa de quatro obras de referência—Sedgewick e Wayne (2010), Zelle (2016), Deitel e Deitel (2016) e Sweigart (2020)—com o intuito de demonstrar como cada autor, através de escolhas específicas de linguagem, estrutura e conteúdo, constrói uma percepção distinta sobre a arte e a ciência da programação.

A pesquisa caracterizou-se por ser de natureza qualitativa, bibliográfica e analítica. O corpus de análise foi rigidamente delimitado às quatro obras

mencionadas, que foram selecionadas por representarem correntes de pensamento distintas e influentes no ensino de programação. A análise focou em critérios específicos e comparáveis, incluindo a linguagem de programação eleita como veículo de ensino, o público-alvo declarado ou implícito na obra, a estrutura e a progressão dos capítulos, a complexidade e a origem dos exemplos e exercícios propostos e, por fim, o objetivo educacional último de cada texto. Através da justaposição metódica desses elementos, foi possível categorizar as obras e identificar suas contribuições únicas e complementares para o ecossistema educacional.

A análise permitiu a identificação de três abordagens pedagógicas claramente delineadas. A primeira, denominada Abordagem Científica e Interdisciplinar, é representada pelas obras de Sedgewick e Wayne (2010) e Zelle (2016). Esses autores transcendem o ensino de sintaxe, posicionando a programação como o cerne da Ciência da Computação. Utilizam a programação como um instrumento para modelar e resolver problemas complexos retirados de domínios como física, matemática e estatística. É notável que Zelle (2016) utilize Python, uma linguagem de sintaxe acessível, não para simplificar os conceitos, mas para permitir que o estudante se concentre na essência dos algoritmos e das estruturas de dados sem se perder em complexidades sintáticas excessivas.

A segunda abordagem identificada é a Abordagem Técnica e Compreensiva, exemplificada pela obra de Deitel e Deitel (2016). Este paradigma tem como foco o domínio profundo e abrangente de uma linguagem de programação específica—neste caso, Java. O texto serve como um compêndio técnico detalhado, cobrindo a linguagem, suas bibliotecas padrão e seus paradigmas (com forte ênfase em Programação Orientada a Objetos) de maneira exaustiva. Seu objetivo principal é formar programadores altamente proficientes em uma tecnologia de mercado, preparando-os para demandas industriais específicas através de uma didática baseada em exemplos concretos, diagramas e tabelas.

Por fim, a terceira abordagem, a Abordagem Prática e Imediatista, é encarnada pelo livro de Sweigart (2020). Esta perspectiva opera em um eixo completamente diferente, pois seu propósito declarado não é ensinar Ciência da Computação, mas sim capacitar o leitor a automatizar tarefas operacionais do cotidiano. A escolha do Python é estratégica, aproveitando-se de suas bibliotecas para realizar ações tangíveis como manipulação de arquivos, automação de planilhas e web scraping. Esta obra visa a um público amplo, incluindo não-programadores de carreira que buscam na programação um

meio para aumentar a produtividade e a eficiência em suas respectivas áreas de atuação.

Conclui-se que as diferentes abordagens analisadas não são excludentes, mas sim complementares. Cada uma delas atende a um objetivo específico e engage um público-alvo distinto, respondendo a diferentes necessidades no espectro do aprendizado de programação. A análise sugere que um currículo ideal poderia se beneficiar de uma integração sequencial dessas perspectivas. Propõe-se que se inicie o processo de aprendizagem com a abordagem prática de Sweigart (2020) para gerar engajamento e demonstrar aplicabilidade imediata. Em um segundo momento, consolidariam-se os fundamentos científicos com o rigor de Zelle (2016) ou Sedgewick e Wayne (2010). Por fim, aprofundar-se-ia o conhecimento técnico com a abordagem compreensiva de Deitel e Deitel (2016). Dessa forma, o discente seria exposto à programação em sua totalidade: como uma ferramenta utilitária, uma disciplina científica formal e uma competência técnica de alto nível, resultando na formação de um profissional mais completo, versátil e capacitado para os diversos desafios do mercado.

## REFERÊNCIAS

DEITEL, Paul; DEITEL, Harvey. Java: como programar. 10. ed. São Paulo: Pearson, 2016.

SEDGEWICK, Robert; WAYNE, Kevin. Introdução à computação com Java: uma abordagem interdisciplinar. São Paulo: Pearson, 2010.

SWEIGART, Al. Automatize tarefas maçantes com Python: programe soluções para problemas do dia a dia. 2. ed. São Paulo: Novatec, 2020.

ZELLE, John M. Python programming: an introduction to computer science. 3. ed. Franklin, Beedle & Associates, 2016.

Palavras-chave: metodologia de ensino; programação de computadores; abordagem pedagógica; linguagens de programação; análise comparativa.

