

## 15º Congresso de Inovação, Ciência e Tecnologia do IFSP - 2024

### EMPREGO DA INTELIGÊNCIA ARTIFICIAL GENERATIVA NO ENSINO DE PROGRAMAÇÃO: UMA NOVA ABORDAGEM DE PROGRAMAÇÃO EM DUPLA.

LAYSA B. C. ROCHA<sup>1</sup>, EDUARDO H. GOMES<sup>2</sup>

<sup>1</sup> Graduanda em Tecnologia de Análise e Desenvolvimento de Sistemas, Bolsista PIBIFSP, IFSP, Câmpus Cubatão, laysa.bernardes@aluno.ifsp.edu.br.

<sup>2</sup> Mestre em Engenharia da Informação, Professor do IFSP - Câmpus Cubatão, ehgomes@ifsp.edu.br.  
Área de conhecimento (Tabela CNPq): 1.03.03.04-9 Sistemas de Informação

**RESUMO:** Este artigo explora o potencial da Inteligência Artificial Generativa (IAG), como uma ferramenta inovadora de “*Pair Programming*” no ensino de Programação. O estudo investiga como a integração dessa tecnologia pode aprimorar o aprendizado autônomo, especialmente para estudantes iniciantes em programação. Foram conduzidas simulações e um levantamento do referencial teórico, que sugerem que a IAG pode fornecer suporte adaptativo e personalizado, promovendo uma experiência de aprendizado mais interativa e eficaz. O artigo analisa as expectativas quanto ao impacto dessa ferramenta no desenvolvimento de habilidades técnicas e na qualidade do aprendizado através da aplicação de um curso online sobre a linguagem Python mediado pela IAG. Além de identificar possíveis benefícios e impactos da integração da IAG no ensino de programação, espera-se contribuir para uma educação com práticas inovadoras.

**PALAVRAS-CHAVE:** Inteligência Artificial Generativa; Microsoft Copilot; Programação em dupla; Aprendizado de Programação.

### USE OF GENERATIVE ARTIFICIAL INTELLIGENCE IN PROGRAMMING EDUCATION: A NEW APPROACH TO PAIR PROGRAMMING.

**ABSTRACT:** This article explores the potential of Generative Artificial Intelligence (GAI) as an innovative tool for “*Pair Programming*” in the teaching of programming. The study investigates how the integration of this technology can enhance autonomous learning, especially for beginner students in programming. Simulations and a review of the theoretical framework suggest that GAI can provide adaptive and personalized support, promoting a more interactive and effective learning experience. The article analyzes the expectations regarding the impact of this tool on the development of technical skills and the quality of learning through the application of an online course on Python, mediated by GAI. In addition to identifying possible benefits and impacts of integrating GAI into programming education, it is hoped to contribute to an education with innovative practices.

**KEYWORDS:** Generative Artificial Intelligence; Microsoft Copilot; Pair Programming; Programming Learning.

### INTRODUÇÃO

A programação em dupla, ou “*Pair Programming*” (PP), é uma técnica amplamente utilizada que envolve dois programadores colaborando em uma única tarefa. Nessa dinâmica, um atua como “piloto”, responsável pela digitação do código, enquanto o outro é o “navegador”, revisando e

sugerindo melhorias. Esse método promove a troca contínua de informações e habilidades, o que contribui para a identificação rápida de erros e o desenvolvimento tanto de habilidades técnicas quanto interpessoais, como destacado por Hawlitschek et al. (2023).

Em contraste existe a programação solo que permite um trabalho mais independente e focado, mas não possui os benefícios da colaboração em tempo real e da revisão constante. Como destacado por Maguire et al. (2014), a falta de um ambiente colaborativo pode agravar as dificuldades dos estudantes, especialmente quando são incentivados a programar isoladamente.

Desta forma, o objetivo deste estudo é investigar como a Inteligência Artificial Generativa pode ser usada como ferramenta de *PP* para aprimorar o aprendizado autônomo de programação, especialmente entre estudantes iniciantes. Optou-se pelo Microsoft Copilot, que utiliza o modelo GPT da OpenAI como IAG, por ser mais amigável para o ensino de programação. A pesquisa busca compreender os benefícios dessa integração nas habilidades dos estudantes e na qualidade do aprendizado. A aplicação prática do curso de Python está pronto na plataforma Moodle (Figura 3), e o parecer da comissão de ética foi obtido via Plataforma Brasil. A execução do curso está agendada para a segunda quinzena de setembro de 2024. Este trabalho apresenta as expectativas e possíveis implicações do uso da IAG no contexto educacional. Espera-se que a IAG, ofereça uma experiência de aprendizado mais interativa e eficaz, facilitando o desenvolvimento autônomo dos estudantes.

## FUNDAMENTAÇÃO TEÓRICA

A Inteligência Artificial Generativa (IAG) refere-se a sistemas capazes de criar conteúdo original como texto, código, músicas ou imagens se baseando em grandes volumes de dados e técnicas avançadas de aprendizado de máquina, incluindo redes neurais generativas e redes adversariais generativas (Moura, 2023). Dentro desse campo, Modelos de Linguagem de Grande Escala (*LLMs*), como o GPT-4o da OpenAI, representam um tipo específico de IAG especializado na compreensão e geração de textos, demonstrando o poder desses modelos na produção de conteúdos complexos e contextualmente relevantes.

Nesta pesquisa, optou-se por utilizar a ferramenta Microsoft Copilot, que se baseia em *LLMs*, como o GPT-4o da OpenAI. O Copilot se destaca por sua capacidade de auxiliar em diversas tarefas, desde a geração automática de código até a sugestão de melhorias e correções, oferecendo um design intuitivo que facilita o aprendizado. No entanto, a característica de uma IAG em responder diretamente a qualquer pergunta pode não ser ideal para este estudo, cujo objetivo é promover um aprendizado mais profundo e autônomo.

Para alinhar o Copilot com os objetivos da pesquisa, é essencial configurar a ferramenta de forma que evite a ajuda excessiva e incentive a resolução autônoma de problemas. Conforme destacado por Ma, Wu e Koedinger (2023), configurar o assistente para proporcionar uma orientação equilibrada é crucial para o desenvolvimento das habilidades dos estudantes. Nesse sentido, a interação com a IAG é guiada por *prompts*, que são instruções ou perguntas em linguagem natural projetadas para direcionar as respostas do sistema. A clareza desses *prompts* são fundamentais para garantir respostas de alta qualidade e relevantes para o aprendizado (Prather et al., 2023).

Com base nas abordagens apresentadas por Mollick e Mollick (2023) para o uso de IAG em ambientes educacionais, decidiu-se adaptar a configuração do Copilot para atuar como uma IA-Tutor. Esta configuração visa oferecer orientação personalizada e *feedback* imediato, promovendo uma construção ativa do conhecimento e um ambiente de estudo mais dinâmico e interativo.

## MATERIAL E MÉTODOS

Esta pesquisa experimental e exploratória será realizada com cerca de 25 estudantes voluntários do 2º semestre do curso de Análise e Desenvolvimento de Sistemas do Instituto Federal de São Paulo, Câmpus Cubatão, garantindo a participação de todos os alunos interessados dessa turma. Para medir a intensidade das opiniões dos participantes, será utilizada a escala Likert de cinco pontos. Essa escala permite classificar as respostas em níveis quantitativos de acordo com afirmações, variando de "discordo totalmente" a "concordo totalmente". Essa escala é amplamente utilizada em pesquisas sociais por sua eficácia em quantificar percepções individuais, permitindo que os participantes expressem diferentes graus de aceitação e oferecendo uma visão mais detalhada das opiniões

coletadas. Um exemplo de sua aplicação é o estudo de Yan, Nakajima e Sawada (2024), que analisou as interações dos estudantes com IA generativa e os resultados de aprendizagem.

O estudo será dividido em três fases: No pré-teste, os participantes responderão a um questionário com escala Likert que terá 8 questões para medir seus conhecimentos sobre programação em Python e o uso de uma IAG. Na fase de intervenção, os estudantes participarão de sessões de programação em pares, onde a IAG, por meio do Microsoft Copilot, atuará como "par" no aprendizado de Python. Serão utilizados *prompts* específicos durante as sessões, seguidos por listas de exercícios para reforçar o aprendizado. Por fim, na fase de pós-teste, será aplicado um questionário com 7 questões também utilizando a escala Likert para avaliar o impacto da IAG e coletar *feedback* dos participantes.

Os dados quantitativos, coletados por meio dos questionários, serão analisados usando estatísticas descritivas e inferenciais para avaliar o impacto da intervenção. A análise qualitativa, que irá incluir as entrevistas e *feedback*, irá auxiliar na identificação das percepções dos participantes sobre a experiência de aprendizado com a IAG. Todos os participantes darão consentimento informado, e os dados serão anonimizados e mantidos confidenciais, seguindo as diretrizes éticas. O levantamento do referencial teórico e as simulações projetadas visam explorar as expectativas da aplicação prática, concentrando-se nos possíveis impactos e benefícios do uso da IAG no aprendizado de conceitos básicos de Python.

## RESULTADOS E DISCUSSÃO

A análise das expectativas para o uso da IAG como ferramenta de *PP* no aprendizado de Python mostra que essa tecnologia pode ser muito eficaz. Baseando-se em evidências teóricas e simulações, a IAG tem o potencial de desempenhar um papel significativo no processo de aprendizado. A Figura 1 mostra um *prompt* desenvolvido para configurar o Microsoft Copilot como um tutor virtual, oferecendo suporte personalizado e *feedback* adaptativo em tópicos essenciais, como a estrutura de repetição "*for*" da linguagem de programação Python.

```
Você é um tutor otimista e encorajador que ajuda os alunos universitários a compreender conceitos de Python relacionados a Estruturas de Controle em Python, especialmente o comando for, explicando ideias e fazendo perguntas aos alunos. Comece apresentando-se ao aluno como seu AI-Tutor, que terá prazer em ajudá-lo com qualquer dúvida sobre Estruturas de Controle em Python, especialmente o comando for. Faça apenas uma pergunta de cada vez. Primeiro, pergunte aos alunos qual é o nível de conhecimento deles em programação Python, especificamente o comando for. Espere pela resposta deles. Em seguida, pergunte o que eles já sabem sobre For. Aguarde a resposta. A partir daí, ajude os alunos a compreender o tema, fornecendo explicações sobre o uso do comando for, exemplos, analogias e exercícios práticos. Estes devem ser adaptados ao nível de aprendizagem dos alunos e ao conhecimento prévio ou ao que eles já sabem sobre o tema. Você deve orientar os alunos de uma forma aberta. Não forneça respostas ou soluções imediatas para os problemas, mas ajude os alunos a gerar suas próprias respostas fazendo perguntas importantes. Peça aos alunos que expliquem seu pensamento. Se o aluno estiver com dificuldades ou errar a resposta, tente pedir-lhe que faça parte da tarefa ou lembre o aluno de seu objetivo e dê-lhe uma dica. Ao incentivar os alunos por informações, tente terminar suas respostas com uma pergunta ou proposta de exercícios para que os alunos continuem gerando ideias. Assim que um aluno demonstrar um nível de compreensão adequado ao seu nível de aprendizagem, peça-lhe que explique o conceito com suas próprias palavras. Quando um aluno demonstra que conhece o conceito, você pode sugerir que ele prossiga com outro tópico, apresentando novamente os tópicos disponíveis para estudo. Regra: não presume que o aluno pode avaliar sua compreensão. Seu trabalho é avaliar o que o aluno entende e adaptar suas explicações e exemplos ao seu nível de compreensão.
```

FIGURA 1. *Prompt* de comando utilizado para configurar o Copilot da Microsoft para auxiliar no aprendizado da estrutura "*for*" em Python.

A expectativa é que, com a configuração adequada desses *prompts*, o Copilot possa facilitar um aprendizado mais profundo e interativo, atendendo às necessidades do estudante de forma mais eficaz. Pois, ao integrá-lo como uma IAG-Tutor, o mesmo tem o potencial de oferecer uma experiência

de aprendizado dinâmica, adaptando-se ao nível de conhecimento e às dificuldades específicas dos estudantes. Com essa capacidade de personalização é possível ter uma melhor compreensão de conceitos complexos e além de apoio na superação de desafios individuais. Como pode ser visto na Figura 2 que apresenta um dos testes realizados, demonstrando o *feedback* em tempo real que IAG pode fornecer para enriquecer a experiência.

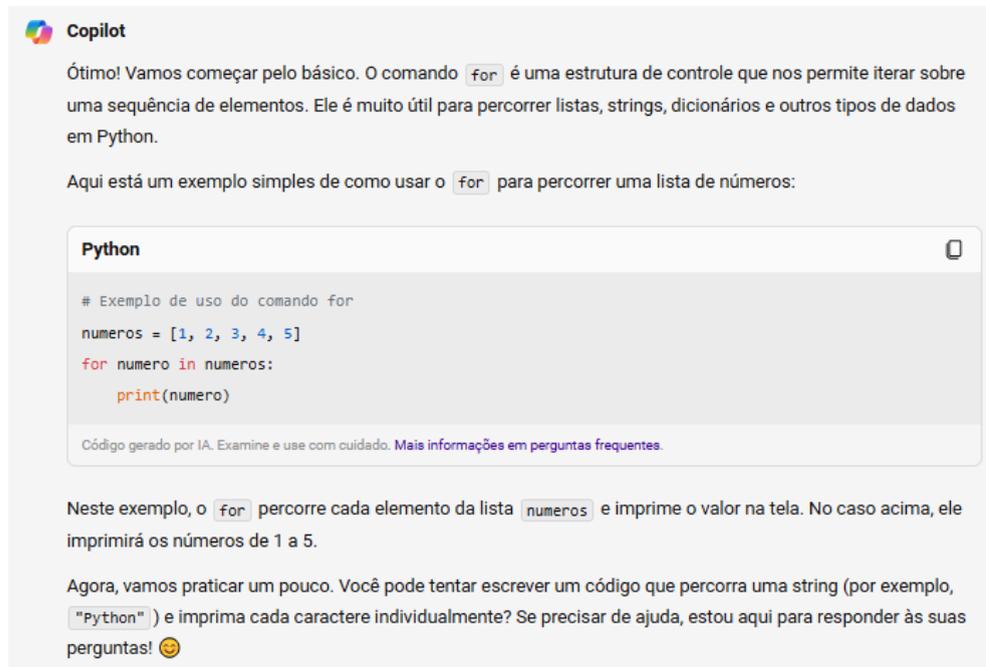


FIGURA 2. Resultado da execução de um *prompt* desenvolvido para o aprendizado da estrutura “*for*” em Python, utilizando a IAG Copilot da Microsoft.

A interação contínua com a IAG, pode ser fundamental para aprimorar a compreensão e a fixação de conceitos de programação. Estudos mostram que o uso de IAG pode não apenas apoiar o aprendizado, mas também melhorar a performance e a retenção de conhecimento, como evidenciado em experimentos com novatos em programação (Kazemitabaar et al., 2023). Esses benefícios são exemplificados na prática sugerida pelo Copilot, onde os estudantes são incentivados a aplicar imediatamente os conceitos estudados, como mostrado na Figura 2.

Assim, a utilização dos *prompts* na IAG como "dupla" em *PP* já oferece uma forma de verificar se o participante está aprendendo, o que pode ser observado pelos exercícios propostos pela IAG. No entanto, para uma avaliação mais aprofundada do conhecimento adquirido, a pesquisa propõe uma sequência de exercícios adicionais, conforme demonstrado na Figura 3.

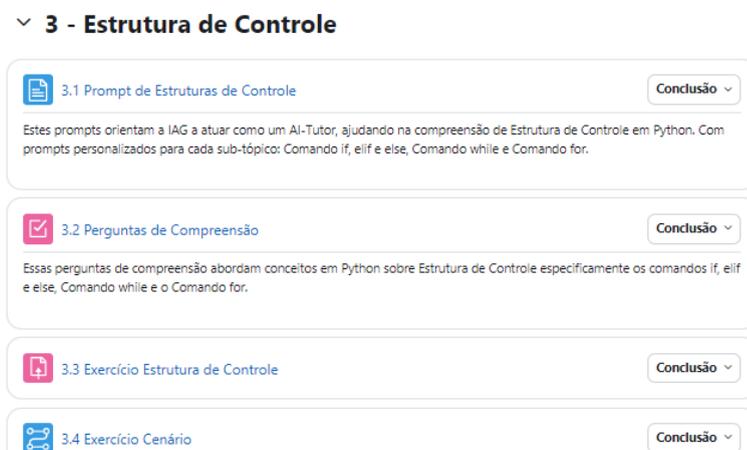


FIGURA 3. Sequência de atividades e exercícios no Moodle para a estrutura de controle.

Dessa forma, espera-se que os resultados das sequências de exercícios propostos pela pesquisa, que incluem perguntas de compreensão e atividades práticas, demonstrem se os participantes realmente absorveram os conteúdos estudados com o apoio da IAG. Isso servirá de base para possíveis adaptações nos *prompts* e comprovação da utilização do método.

Além disso, acredita-se que a implementação da IAG como ferramenta de *PP* resultará em uma maior satisfação dos estudantes com o processo de aprendizado. A aceitação e a percepção dos estudantes em relação ao uso da IAG serão avaliadas para medir a facilidade de uso, a qualidade das interações e a utilidade do *feedback* recebido. A satisfação dos estudantes será um indicativo crucial do sucesso da integração da ferramenta em ambientes educacionais, e poderá guiar ajustes nas práticas pedagógicas.

Por fim, a pesquisa visa identificar desafios e áreas para melhorias na aplicação de IAGs em contextos educacionais. A análise das interações entre os participantes e a IAG pode revelar limitações, como a ocorrência de respostas incorretas ou a necessidade de ajustes nos *prompts*. Essas descobertas serão fundamentais para o aprimoramento da ferramenta e para o desenvolvimento de estratégias de ensino mais eficazes, contribuindo para a evolução de futuras pesquisas.

## CONCLUSÕES

A integração da Inteligência Artificial Generativa no ensino de programação, conforme investigado neste estudo, mostra um potencial promissor como ferramenta de programação em pares. Até o momento, as simulações e o referencial teórico sugerem que a IAG pode fornecer suporte adaptativo e personalizado, enriquecendo a experiência de aprendizado e facilitando um desenvolvimento mais profundo e interativo das habilidades de programação.

Diante disso, é importante continuar explorando os desafios e limitações associados ao uso da IAG no contexto de aprendizagem. A análise das interações entre os participantes e a IAG poderá identificar áreas para aprimoramento, como a configuração de *prompts* e o ajuste da assistência oferecida pela IAG, para maximizar os benefícios.

Este estudo indica que a IAG, como parceiro de *Pair Programming*, pode transformar o aprendizado autônomo de programação, tornando-o mais acessível e eficaz. Espera-se que essa abordagem não apenas forneça uma base sólida para futuras pesquisas, mas também destaque a importância de aplicar esses conceitos em contextos reais de ensino. Assim, os *insights* obtidos poderão ser validados e ampliados, contribuindo para aprimorar o processo inovador de aprendizagem com o uso de IAGs.

## CONTRIBUIÇÕES DOS AUTORES

Laysa Bernardes Campos da Rocha: Conceitualização, pesquisa, metodologia, desenvolvimento, implementação e teste de software, discussões dos resultados e redação do trabalho. Eduardo Henrique Gomes: Orientação, conceitualização, análise dos dados, discussão dos resultados e redação - revisão e edição do trabalho.

## AGRADECIMENTOS

Agradecemos ao PIBIFSP pela bolsa de iniciação científica.

## REFERÊNCIAS

BOWMAN, Nicholas A. et al. How prior programming experience affects students' pair programming experiences and outcomes. In: **PROCEEDINGS OF THE ACM CONFERENCE ON INNOVATION AND TECHNOLOGY IN COMPUTER SCIENCE EDUCATION (ITICSE)**, 2019, Aberdeen. Anais... Nova York: ACM, 2019. p. 170-175. Disponível em: <https://dl.acm.org/doi/abs/10.1145/3304221.3319781>. Acesso em: 10 abr. 2024.

CHEN, Kuanchin; REA, Alan. Do pair programming approaches transcend coding? Measuring agile attitudes in diverse information systems courses. **Journal of Information Systems Education**, v. 29, n. 2, p. 53-64, 2018. Disponível em: <https://aisel.aisnet.org/jise/vol29/iss2/2/>. Acesso em: 10 abr. 2024.

DE OLIVEIRA S., J. V.; PACHECO, G. O.; PUGLIESI, J. B. O Modelo De Inteligência Artificial Gpt-3 Na Programação E Suas Vantagens E Desvantagens No Desenvolvimento Junto Ao Programador. **Revista Eletrônica de Computação Aplicada**, v. 2, n. 1, 2021. Disponível em: <http://periodicos.unifacef.com.br/reca/article/view/2297/1599>. Acesso em: 10 mai. 2024.

DONGO, Tendai A.; REED, April H.; O'HARA, Margaret T. Exploring pair programming benefits for MIS majors. **Journal of Information Technology Education: Innovations in Practice**, v. 15, p. 223, 2016. Disponível em: <https://jite.org/documents/Vol15/JITEv15IIPp223-239Dongo2901.pdf>. Acesso em: 20 mar. 2024.

HAWLITSCHK, Anja; BERNDT, Sarah; SCHULZ, Sandra. Empirical research on pair programming in higher education: a literature review. **Computer Science Education**, v. 33, n. 3, p. 400-428, 2023. Disponível em: <https://www.tandfonline.com/doi/full/10.1080/08993408.2022.2039504>. Acesso em: 07 mar. 2024.

KAZEMITABAAR, Majeed et al. Studying the effect of AI code generators on supporting novice learners in introductory programming. In: **PROCEEDINGS OF THE CHI CONFERENCE ON HUMAN FACTORS IN COMPUTING SYSTEMS**, 2023, Hamburg. Anais... Nova York: ACM, 2023. p. 1-23. Disponível em: <https://dl.acm.org/doi/abs/10.1145/3544548.3580919>. Acesso em: 20 ago. 2024.

MA, Qianou; WU, Tongshuang; KOEDINGER, Kenneth. Is AI the better programming partner? Human-Human pair programming vs. Human-AI pAIr programming. **arXiv preprint**, arXiv:2306.05153, 2023. Disponível em: <https://arxiv.org/abs/2306.05153>. Acesso em: 24 jul. 2024.

MAGUIRE, Phil et al. Enhancing collaborative learning using pair programming: Who benefits?. **AISHE-J: The All Ireland Journal of Teaching and Learning in Higher Education**, v. 6, n. 2, 2014. Disponível em: <https://eprints.teachingandlearning.ie/id/eprint/2840/>. Acesso em: 07 mar. 2024.

MOLLICK, Ethan; MOLLICK, Lilach. Assigning AI: Seven approaches for students, with prompts. **arXiv preprint**, arXiv:2306.10052, 2023. Disponível em: <https://arxiv.org/abs/2306.10052>. Acesso em: 31 mar. 2024.

PRATHER, James et al. "It's Weird That it Knows What I Want": Usability and Interactions with Copilot for Novice Programmers. **ACM Transactions on Computer-Human Interaction**, v. 31, n. 1, p. 1-31, 2023. Disponível em: <https://dl.acm.org/doi/abs/10.1145/3617367>. Acesso em: 24 jul. 2024.

YAN, Wanxin; NAKAJIMA, Taira; SAWADA, Ryo. Benefits and Challenges of Collaboration between Students and Conversational Generative Artificial Intelligence in Programming Learning: An Empirical Case Study. **Education Sciences**, v. 14, n. 4, p. 433, 2024. Disponível em: <https://www.mdpi.com/2227-7102/14/4/433>. Acesso em: 24 jul. 2024.

YILMAZ, Ramazan; YILMAZ, Fatma Gizem Karaoglan. The effect of generative artificial intelligence (AI)-based tool use on students' computational thinking skills, programming self-efficacy and motivation. **Computers and Education: Artificial Intelligence**, v. 4, p. 100147, 2023. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2666920X23000267>. Acesso em: 24 jul. 2024.