

Uso da Programação em Pares como Estratégia para o Ensino de Fundamentos de Programação com Python

Maria Julia S. de Souza¹, Paulo Arthur F. Lemes¹, Gabriel Lucas S. Rodrigues¹, Antônio Kawan F. D. Pereira¹, Patrícia F. C. de Vasconcelos¹, Edvan C. de Miranda¹

¹Universidade Federal do Ceará (UFC) - Campus Russas
62900-420 – Russas – CE – Brasil

{mariajulia44, arthurp1910, gabriellucassr, kawanfreitas05}@alu.ufc.br
{edvan, patricia.vasconcelos}@ufc.br

Abstract. *This paper presents an experience report on the use of the pair learning methodology in teaching Fundamentals of Programming with Python, applied in a mini-course aimed at beginner students. The approach sought to promote active learning through collaboration, verbalization of reasoning, and joint review of solutions. Quantitative and qualitative data obtained from pre- and post-class forms and participant feedback were analyzed. The results indicate an average improvement of 20% in performance and broad acceptance of the methodology, associated with increased confidence, engagement, and conceptual understanding. The experience reinforces the potential of pair learning as an effective strategy for introductory programming instruction.*

Resumo. *Este trabalho apresenta um relato de experiência sobre o uso da programação em pares no ensino de Fundamentos de Programação com Python, aplicado em um minicurso voltado a estudantes iniciantes. A abordagem buscou promover o aprendizado ativo por meio da colaboração, verbalização de raciocínios e revisão conjunta de soluções. Foram analisados dados quantitativos e qualitativos obtidos de formulários pré e pós-aula e de feedback dos participantes. Os resultados indicam evolução média de 20% no desempenho e ampla aceitação da metodologia, associada ao aumento da segurança, engajamento e compreensão conceitual. A experiência reforça o potencial do trabalho em duplas como estratégia efetiva para o ensino introdutório de programação.*

1. Introdução

O ensino introdutório de programação representa um desafio recorrente em cursos de graduação, especialmente para estudantes com pouca ou nenhuma experiência prévia. Dificuldades relacionadas à lógica, abstração e resolução de problemas tendem a gerar frustração e insegurança, contribuindo para baixos índices de desempenho e persistência, assim como descreve Lima e Menezes (2024). Nesse contexto, metodologias ativas têm ganhado destaque por favorecerem o engajamento, a autonomia e a construção coletiva do conhecimento.

Entre essas abordagens, a programação em pares (*pair learning / pair programming*) destaca-se por reduzir a carga cognitiva dos iniciantes, promover a verbalização do raciocínio e estimular a análise crítica de soluções entre colegas. Segundo McDowell et al.

(2006), duplas colaborativas tendem a apresentar melhor desempenho, maior motivação, persistência e qualidade nos programas produzidos, especialmente em ambientes introdutórios de programação.

Embora a programação em pares seja amplamente investigada na literatura, a presente experiência apresenta características particulares que a distinguem de abordagens tradicionais. Diferentemente de estudos conduzidos em disciplinas regulares, este trabalho descreve a aplicação da metodologia em um contexto de extensão universitária, estruturado em formato de minicurso intensivo, com participação voluntária e público heterogêneo, composto por estudantes de diferentes áreas. Além disso, a investigação adota uma estratégia longitudinal de avaliação, combinando análises quantitativas e qualitativas ao longo de todos os encontros, permitindo observar não apenas variações de desempenho, mas também aspectos perceptivos e socioemocionais associados à aprendizagem colaborativa.

Nesse contexto, o relato descreve a aplicação da programação em pares em um minicurso de Fundamentos de Programação com Python, ofertado a estudantes de cinco cursos da Universidade Federal do Ceará, campus de Russas. A escolha de Python fundamenta-se em sua legibilidade e simplicidade, adequadas a iniciantes, conforme destaca Downey (2016). Além de amplamente utilizada em cursos introdutórios, a linguagem tem sido adotada em disciplinas de Engenharia, favorecendo abordagens transdisciplinares e a articulação entre teoria e prática. Lima et al. (2025).

O objetivo deste trabalho é apresentar e discutir evidências quantitativas e qualitativas obtidas ao longo do minicurso, destacando os efeitos da programação em pares em um contexto de extensão universitária com público heterogêneo e avaliação longitudinal. As análises quantitativas, baseadas em respostas em escala Likert, foram complementadas por boxplots para mostrar a evolução conceitual nos formulários diagnósticos. As análises qualitativas, por meio de análise temática do formulário de feedback, identificaram padrões nas respostas abertas, evidenciando efeitos socioemocionais da colaboração. Em conjunto, os resultados oferecem subsídios para o uso da programação em pares em contextos semelhantes de ensino.

Este artigo está organizado da seguinte forma: a Seção 2 apresenta o contexto e a motivação do minicurso; a Seção 3 discute trabalhos relacionados à aprendizagem em pares no ensino de programação; a Seção 4 descreve a metodologia adotada, incluindo público-alvo, planejamento e condução das atividades; a Seção 5 apresenta e analisa os resultados quantitativos e qualitativos; a Seção 6 discute as ameaças à validade; a Seção 7 traz as considerações finais e trabalhos futuros; e a Seção 8 apresenta os agradecimentos.

2. Contextualização

O Programa de Educação Tutorial (PET) da Universidade Federal do Ceará desenvolve ações integradas de ensino, pesquisa e extensão, e a oferta de minicursos como ação de extensão contribui para a integração entre ensino, pesquisa e prática, favorecendo a aprendizagem contextualizada, como observado por Freire (1996). Entre as ações de extensão, destaca-se o projeto Capacite-se, voltado à oferta de minicursos que atendem às demandas formativas da comunidade acadêmica. Como relatado por Araújo et al. (2023), o projeto Capacite-se tem como objetivo complementar a formação dos estudantes, proporcionando habilidades práticas e teóricas que fortalecem seu aprendizado dentro e fora da

sala de aula.

Anualmente, cerca de 200 estudantes ingressam nos cursos de Ciência da Computação e Engenharia de Software do campus. Ainda no primeiro semestre, entram em contato com a disciplina de Fundamentos de Programação, ministrada em Python. Nesse momento inicial, muitos alunos apresentam dificuldades conceituais e de abstração, o que pode desencadear pânico, sobrecarga cognitiva e, conseqüentemente, comprometer o desempenho acadêmico, podendo até levar à desistência, conforme evidenciado por Rahman et al. (2020).

Diante desse cenário, torna-se necessária a adoção de estratégias pedagógicas mais eficazes. Assim, o PET estruturou o minicurso Fundamentos de Programação em Python, adotando a programação em pares como forma de promover a aprendizagem colaborativa, engajamento dos ingressantes e redução da sobrecarga inicial da graduação.

3. Trabalhos Relacionados

As investigações existentes sobre aprendizagem em pares e ensino introdutório de programação convergem em três fundamentos centrais: (1) o apoio mútuo da dupla reduz a carga cognitiva de iniciantes; (2) a verbalização constante favorece a consolidação conceitual; e (3) a interação social aumenta o engajamento, a persistência e a autoconfiança.

Korber et al. (2021) analisaram o uso da programação em pares em cursos introdutórios e observaram que os estudantes apresentam melhor desempenho e maior motivação quando trabalham em duplas, especialmente em linguagens textuais como Python. Esses achados reforçam a opção metodológica adotada no minicurso aqui descrito, para o qual a prática de programação em pares foi estruturada como elemento central de apoio ao processo de aprendizagem.

No contexto brasileiro, Pereira et al. (2021) demonstraram que oficinas colaborativas de Python, baseadas em aprendizagem entre pares, aumentam a segurança dos estudantes e reduzem as barreiras iniciais ao ingresso na programação. Assim como no presente minicurso, os autores destacam que a troca contínua entre colegas facilita a compreensão de estruturas básicas da linguagem e estimula a construção conjunta de soluções.

Colin et al. (2025) abordam a formação das duplas na programação em pares, evidenciando que diferenças de autoeficácia influenciam diretamente a colaboração e o progresso das duplas. Essa discussão dialoga com o presente relato, já que as duplas foram formadas espontaneamente, o que pode explicar parte da variabilidade inicial observada nos pré-testes.

Por fim, Alves et al. (2019) investigaram o uso de práticas colaborativas como *Coding Dojo* e mostraram que tais abordagens promovem participação ativa, apoio mútuo e maior compreensão de conceitos introdutórios. Esses resultados reforçam o potencial de ambientes colaborativos, como o adotado no minicurso, para favorecer a aprendizagem de iniciantes em programação.

Em síntese, a literatura aponta que a programação em pares é especialmente eficaz em cenários de introdução à programação, proporcionando benefícios cognitivos, afetivos e sociais. Diante desses resultados, o minicurso relatado neste trabalho alinha-se às evidências existentes, buscando potencializar a aprendizagem de fundamentos de programação em Python por meio de práticas colaborativas estruturadas.

4. Metodologia

A pesquisa caracteriza-se como um estudo de caso exploratório, conforme Creswell (2017), conduzido durante o minicurso com abordagem mista. A coleta quantitativa incluiu formulários pré e pós-aula em escala Likert, e a qualitativa contemplou respostas abertas do formulário de feedback. O minicurso ocorreu em três encontros presenciais, totalizando seis horas, com momentos expositivos e práticas guiadas em Python. Os participantes foram organizados em duplas durante todo o curso, seguindo a técnica de Programação em Pares. A cada dois tópicos, realizavam breves explicações mútuas antes da prática, retomando os conceitos apresentados. Essa dinâmica estimulou a verbalização, o apoio entre colegas e a consolidação imediata dos conteúdos.

A avaliação da aprendizagem e da percepção dos estudantes foi estruturada de forma longitudinal, com formulários pré e pós-aula em todos os encontros. Esses instrumentos continham questões objetivas para identificar mudanças conceituais imediatas e a autopercepção após cada tópico. Ao final do minicurso, aplicou-se um formulário de feedback com itens em escala Likert sobre clareza do curso, efetividade da abordagem colaborativa e aceitação da Programação em Pares. Como síntese, os estudantes realizaram um desafio individual, sem apoio da dupla, para verificar a autonomia e a internalização dos conceitos básicos de Python. Todos os procedimentos seguiram princípios éticos de pesquisa com seres humanos. O Termo de Consentimento Livre e Esclarecido (TCLE) foi apresentado e assinado antes das atividades, garantindo confidencialidade, participação voluntária e desistência sem prejuízo acadêmico. O relato restringiu-se à coleta de dados pedagógicos, com finalidade exclusivamente educacional e de pesquisa.

O minicurso foi estruturado em seis etapas (Seções 4.1 a 4.6): definição do público-alvo e perfil dos inscritos (4.1); planejamento pedagógico, com conteúdos, encontros e instrumentos avaliativos (4.2); preparação da execução e dinâmica das aulas (4.3); composição da equipe (4.4); aplicação da Programação em Pares nos três dias de curso (4.5); e avaliação, com instrumentos para mensurar desenvolvimento conceitual, percepção dos estudantes e aceitação da abordagem (4.6).

4.1. Definição do Público-alvo

O minicurso teve como objetivo apoiar estudantes ingressantes na programação. Por isso, foi direcionado não apenas aos alunos dos cursos de Tecnologia da Informação (Ciência da Computação e Engenharia de Software), mas também aos alunos dos cursos de Engenharia Civil, Engenharia Mecânica, Engenharia de Produção.

O processo de inscrição ocorreu por meio de um formulário online amplamente divulgado nas redes sociais do PET, além de folhetos e cartazes distribuídos na Universidade. Essa estratégia permitiu alcançar um público mais diverso.

Ao todo, 85 estudantes se inscreveram no curso, sendo 2 de Engenharia Civil, 10 de Engenharia Mecânica, 3 de Engenharia de Produção, 36 de Ciência da Computação e 34 de Engenharia de Software. Apesar do elevado número de inscrições, apenas 50 participantes compareceram aos dias de realização do curso, e 32 responderam aos formulários diagnóstico e de *feedback*.

Para identificar o nível de familiaridade dos participantes com programação, o formulário de inscrição incluiu três perguntas-chave: “Você já teve algum contato com

programação?”, “Qual é o seu nível de familiaridade com computadores e lógica de programação?”, “Você já programou em alguma linguagem? Se sim, qual?” As respostas a essas questões permitiram caracterizar o perfil dos inscitos e orientar o planejamento pedagógico do minicurso. O questionário aplicado encontra-se disponível para consulta no link de rodapé.

As respostas evidenciaram um público heterogêneo. Na primeira pergunta, 41,2% dos inscitos indicaram que o minicurso seria seu primeiro contato com programação. Quanto à familiaridade com computadores e lógica, 47,1% declararam-se iniciantes. Em relação às linguagens de programação, observou-se grande diversidade de experiências; contudo, entre aqueles que já haviam programado, Python foi mencionada com maior frequência como primeira linguagem utilizada. Diante desse perfil e visando uma abordagem inclusiva, a adoção da linguagem Python e da metodologia de programação em pares mostrou-se particularmente adequada para favorecer a aprendizagem dos participantes.

4.2. Construção do curso

O minicurso foi planejado e desenvolvido por dois integrantes do PET UFC Russas, seguindo princípios de progressão pedagógica e aprendizagem ativa. A estrutura do minicurso foi fundamentada na programação em pares, uma prática colaborativa que favorece a aprendizagem ativa, a troca de conhecimentos e a assimilação gradual de conceitos, especialmente em contextos introdutórios de programação, como mostra o livro de Williams et al. (2003) que apresenta os fundamentos teóricos e práticos da programação em pares.

A carga horária total contemplou seis horas em sala de aula, distribuídas em três encontros de duas horas cada, além de duas horas adicionais destinadas ao desenvolvimento do desafio final proposto no último dia. Os conteúdos foram organizados conforme apresentado a seguir:

- **Dia 1:** instalação de ambientes, sintaxe básica, tipos de dados e operadores;
- **Dia 2:** estruturas condicionais, laços de repetição e entrada de dados;
- **Dia 3:** funções, listas e boas práticas de organização de código.

Para monitorar o progresso dos participantes e avaliar a efetividade da proposta, foram aplicados questionários pré e pós-aula em todos os encontros, possibilitando a análise da evolução do aprendizado ao longo do minicurso.

4.3. Organização da execução

A condução das aulas foi planejada para alternar, de forma equilibrada, entre momentos de exposição teórica e atividades práticas colaborativas. No início do curso, os participantes foram organizados em duplas, de modo a favorecer a interação e o apoio mútuo. A cada dois tópicos apresentados, os estudantes eram convidados a explicar ao colega os conceitos recém-aprendidos. Essa estratégia permitiu identificar rapidamente possíveis dificuldades: quando grande parte da turma demonstrava dúvidas, o conteúdo era retomado coletivamente; quando as dificuldades eram pontuais, os instrutores auxiliares realizavam intervenções direcionadas. Em todos os encontros foram realizados exercícios práticos, que abrangeram desde a tradução de fluxogramas para código em Python até a utilização de estruturas condicionais, laços e listas. Ao final do minicurso, foi proposto um desafio

⁰Dataset disponível em: [doi:10.5281/zenodo.17849210](https://doi.org/10.5281/zenodo.17849210)

individual no formato de um CRUD simples para consolidar os conhecimentos trabalhados. O CRUD consistiu em um sistema básico de cadastro de usuários, permitindo criar, visualizar, editar e excluir registros. O documento completo com os requisitos detalhados encontra-se disponível no link indicado na nota de rodapé.

4.4. Equipe responsável

O minicurso foi ministrado por dois estudantes de Ciência da Computação, com experiência em programação e domínio de Python. A proposta, incluindo conteúdos, metodologia e organização das atividades, foi validada pelos professores tutores do PET UFC Russas, assegurando alinhamento pedagógico e rigor metodológico. A equipe contou ainda com dois monitores, responsáveis por apoiar as atividades práticas e esclarecer dúvidas. As turmas foram divididas em duas salas, cada uma com um ministrante e um monitor, para garantir acompanhamento mais próximo. Nos momentos em que não conduziam o conteúdo, os ministrantes também atuavam como suporte às práticas.

4.5. Aplicação prática da metodologia em pares

A metodologia em pares foi utilizada de forma sistemática ao longo dos três dias. Durante os momentos de prática, cada dupla era incentivada a discutir suas ideias, verbalizar raciocínios e revisar o código do colega. Esse processo aumentou o engajamento e promoveu um ambiente de apoio mútuo. Dificuldades pontuais foram observadas no início, principalmente quanto à comunicação entre parceiros que não se conheciam. Entretanto, ao longo do curso, houve melhora perceptível na interação e na autonomia dos estudantes, que passaram a explicar suas soluções com maior clareza e segurança.

4.6. Avaliação dos participantes

A avaliação dos participantes foi conduzida de forma contínua ao longo dos três encontros do minicurso, por meio da aplicação de formulários diagnósticos pré e pós-aula contendo as mesmas questões. Essa estratégia teve como objetivo mensurar a evolução conceitual dos estudantes em cada etapa, permitindo analisar tanto o desempenho imediato quanto a progressão do aprendizado ao longo do curso. A média de participação nesses instrumentos foi de aproximadamente 46 respostas por encontro, variando conforme a disponibilidade dos alunos.

Os formulários foram elaborados sem a apresentação das respostas corretas, com o intuito de minimizar vieses de memorização e garantir que os resultados refletissem o nível real de compreensão dos participantes. A comparação entre os dados pré e pós-aula possibilitou identificar avanços no entendimento dos conteúdos e mapear tópicos que demandaram maior atenção pedagógica, prática amplamente adotada em contextos educacionais para avaliar ganhos de aprendizagem imediatos, conforme discutido por Black e Wiliam (1998).

Ao término do minicurso, foi aplicado um formulário geral de *feedback*, voltado à avaliação da metodologia adotada e da experiência global dos participantes. Esse instrumento combinou questões fechadas em escala *Likert*, contemplando aspectos como clareza dos conteúdos, efetividade da programação em pares, organização das aulas e percepção de aprendizagem, com questões abertas que permitiram aos estudantes expressar sugestões, críticas e impressões livres sobre a dinâmica do minicurso.

⁰Dataset disponível em: [doi:10.5281/zenodo.17849210](https://doi.org/10.5281/zenodo.17849210)

5. Resultados

Nesta seção, são apresentados os resultados do minicurso, reunindo análises quantitativas e qualitativas sobre o desempenho dos participantes e suas percepções acerca da metodologia adotada. Primeiro, são examinados os formulários pré e pós-aula dos três dias, evidenciando a evolução do aprendizado. Em seguida, são discutidas as impressões gerais sobre a aprendizagem em pares e seu impacto no engajamento e na compreensão dos conteúdos. Por fim, a análise temática das respostas abertas aprofunda a interpretação das experiências, desafios e benefícios relatados pelos estudantes.

5.1. Análise Quantitativa

As respostas fechadas dos formulários pré e pós-aula, incluindo os dados em escala *Likert* coletados ao longo dos três dias do minicurso, foram analisadas por meio de estatísticas descritivas. Esse tipo de abordagem é amplamente recomendado para dados oriundos de escalas *Likert*, pois o uso de frequências, porcentagens, médias e da distribuição dos padrões de resposta permite sintetizar as percepções dos participantes e identificar tendências gerais, sem a necessidade de recorrer a procedimentos inferenciais mais complexos, conforme discutido por Boone (2012). Além disso, visualizações gráficas, como *boxplots*, são indicadas para representar a distribuição e a variabilidade dos dados educacionais, possibilitando comparações claras entre grupos e momentos distintos de avaliação, como mostra Tukey (1977).

5.1.1. Análise dos Formulários Pré e Pós-aula

A Figura 1 apresenta a evolução do desempenho, comparando os resultados dos formulários pré e pós-aula ao longo dos três dias de minicurso.

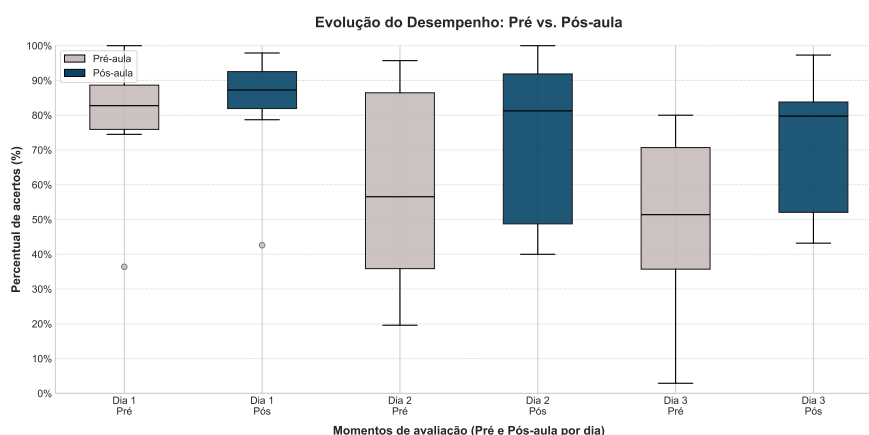


Figura 1. Evolução do desempenho Pré vs. Pós-aula

A análise quantitativa foi realizada a partir da porcentagem média de acertos por questão, permitindo identificar tanto os conceitos de maior domínio quanto aqueles que exigiram maior atenção dos estudantes. Os *boxplots* produzidos para os três dias evidenciam a distribuição dos acertos, suas medianas, dispersão e valores atípicos, oferecendo uma visão clara da evolução ao longo do minicurso.

No primeiro dia, o percentual de acertos foi elevado tanto no pré-aula quanto no pós-aula, possivelmente em função do caráter introdutório do conteúdo. A presença de *outliers* indica lacunas pontuais de conhecimento, enquanto a caixa compacta sugere compreensão relativamente homogênea. A elevação da mediana no pós-aula confirma o ganho imediato após as atividades em pares.

No segundo dia, a maior variabilidade no pré-aula reflete o aumento da complexidade dos tópicos abordados. Ainda assim, observa-se elevação de aproximadamente 30% na mediana e redução da dispersão no pós-aula, indicando melhor equilíbrio entre os participantes e consolidação dos conteúdos, mesmo com menor número de respondentes.

O terceiro dia apresenta o comportamento mais marcante: o *boxplot* do pós-aula é mais alto e menos disperso, refletindo avanços tanto de alunos com baixo desempenho inicial quanto daqueles com maior facilidade. A elevação simultânea dos limites inferior e superior indica um avanço mais uniforme do grupo.

O padrão observado nos três dias não é casual. A literatura indica que, conforme discutido por Williams et al. (2003), a alternância de papéis na metodologia em pares reduz a carga cognitiva, enquanto a verbalização do raciocínio diante do colega contribui para reorganizar e consolidar o entendimento. O suporte imediato entre os pares reduz a frustração típica de iniciantes, favorecendo resultados mais consistentes. Assim, a redução progressiva da variabilidade e a elevação das medianas observadas são coerentes com os mecanismos de aprendizagem colaborativa descritos por estudos prévios.

De forma geral, os três pares de *boxplots* demonstram melhora consistente, redução de dispersão e avanço coletivo ao longo do minicurso, evidenciando o impacto positivo da abordagem em pares na compreensão de fundamentos de programação.

A Figura 2 apresenta a média total de acertos e erros obtida em ambos os momentos avaliativos, permitindo uma comparação direta entre o nível de conhecimento prévio e o desempenho após a ministração dos conteúdos.

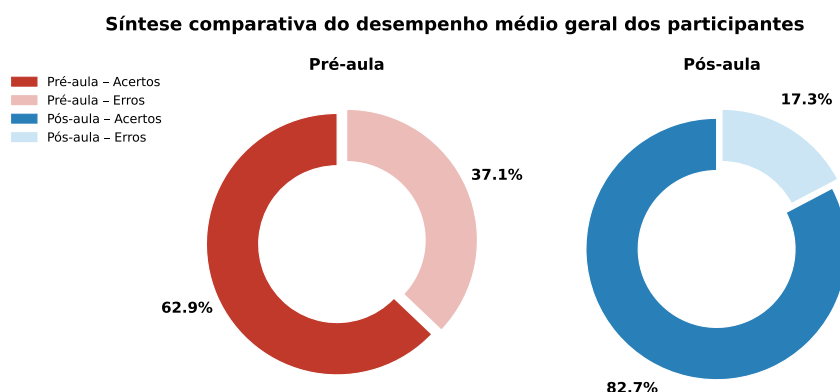


Figura 2. Comparativo geral de acertos e erros nos formulários pré e pós-aula.

Os resultados quantitativos mostram um ganho médio de cerca de 20 pontos percentuais na taxa de acertos (de 63% no pré-teste para 83% no pós-teste), além de uma redução progressiva da dispersão nos *boxplots* ao longo dos três dias. Esses achados confirmam os resultados clássicos de McDowell et al. (2006), que apontaram maior retenção de conteúdo, melhor qualidade de programas e maior confiança entre estudantes que utilizaram programação em pares em disciplinas introdutórias. A equalização do desempenho

observada, com elevação simultânea dos limites inferior e superior no terceiro dia, dialoga diretamente com Korber et al. (2021), reforçando que a colaboração em duplas não só diminui a média de erros, mas também reduz a variabilidade entre alunos com níveis iniciais diferentes. Este trabalho confirma e amplia as evidências da literatura.

5.1.2. Percepções dos Estudantes sobre a Metodologia em Pares

No último dia de minicurso, os discentes responderam um formulário geral de feedback. Esses dados permitem avaliar de forma direta a percepção dos participantes sobre a metodologia em pares adotada. Entre os 32 respondentes, cerca de 75% avaliaram positivamente a experiência de aprender em duplas (53,1% “concordo totalmente” e 21,9% “concordo parcialmente”), enquanto aproximadamente 12,5% mantiveram posição neutra e menos de 13% declararam discordância. Esse padrão evidencia uma aceitação majoritariamente favorável à dinâmica colaborativa.

Além disso, quando questionados especificamente se a metodologia em pares contribuiu para o próprio aprendizado, 78% dos estudantes indicaram concordância, sendo 46,9% em total acordo e 31,2% em concordância parcial. As médias dessas avaliações permaneceram acima de 4,0 em uma escala de 1 a 5, sugerindo que a prática de discutir soluções, validar raciocínios e explicar conceitos ao parceiro funcionou como um mecanismo direto de consolidação do conteúdo.

Esses dados convergem com os resultados obtidos nos formulários pré e pós-aula, reforçando que a evolução no desempenho não se limitou à exposição aos conteúdos, mas esteve associada à interação constante entre pares, particularmente na formulação de hipóteses, na depuração de erros e na verbalização dos passos da solução.

Os altos índices de aceitação da metodologia, 75% dos participantes avaliaram positivamente a experiência de aprender em duplas e 78% consideraram que ela contribuiu para seu aprendizado, corroboram os achados de Pereira et al. (2021) que observaram um aumento da segurança e redução das barreiras iniciais em oficinas colaborativas de Python no Brasil. A média superior a 4,0 (em escala de 1 a 5) nos itens sobre discussão de soluções e validação de raciocínios confirma o papel central da verbalização e da alternância de papéis, mecanismos destacados por Williams et al. (2003) e Colin et al. (2025) como fundamentais para a consolidação conceitual e o desenvolvimento da autoeficácia. Esses dados de percepção convergem com os resultados dos formulários pré e pós-aula, indicando que a evolução no desempenho não resultou apenas da exposição aos conteúdos, mas esteve diretamente associada à interação contínua entre os pares.

5.2. Análise Qualitativa

A análise das respostas abertas do formulário de feedback foi conduzida por meio de análise temática, seguindo o procedimento proposto por Braun e Clarke (2006). Inicialmente, realizou-se a leitura recorrente das respostas para familiarização com os dados, seguida da codificação de unidades de significado relevantes. Esses códigos foram posteriormente agrupados por similaridade, originando temas que representam aspectos centrais das percepções dos participantes.

A Figura 3 mostra o processo após a revisão e o refinamento, os temas finais definidos e organizados em uma estrutura hierárquica, cuja representação visual considera

a frequência das respostas associadas, permitindo identificar a predominância de aspectos cognitivos e socioemocionais da experiência do minicurso.



Figura 3. Representação hierárquica dos temas e códigos identificados na análise temática (tamanho proporcional à frequência das respostas)

O tema “Consolidação conceitual por meio da colaboração” (em lilás) reúne os códigos “melhor entendimento”, “troca de conhecimentos” e “ensinar os amigos”, correspondendo a 47% das respostas analisadas. Exemplos como R7 — “Explicar para o colega ajuda a fixar o conteúdo aprendido” e R12 — “Trocar ideias facilitou muito o entendimento” reforçam que o diálogo e a verbalização do raciocínio atuaram como mecanismos de aprendizagem ativa, conforme descrito por Williams et al. (2003).

O tema “Suporte emocional e ajuda mútua” (em azul-esverdeado) reflete o papel efetivo da colaboração, com códigos como “ajuda mútua” e “resolver dúvidas rapidamente”. Relato como R10 — “Um ajuda o outro quando trava, e isso deixa o curso mais leve” evidenciam a criação de um ambiente de apoio e segurança, reduzindo a ansiedade típica dos iniciantes, um efeito relatado também por Korber et al. (2021).

O tema “Motivação e engajamento” (em laranja) aparece como um dos mais expressivos, com códigos como “vontade de continuar estudando programação” e “mais dias de minicurso”. Afirmações como R22 — “O minicurso me motivou a continuar estudando programação” ilustram o impacto da programação em pares na manutenção do interesse e na percepção positiva da própria aprendizagem.

O tema “Clareza e mediação docente” (em verde) reúne elogios à atuação dos ministrantes e monitores, com códigos como “esclareceu bem as dúvidas” e “boa didática”, reforçam o papel do professor como mediador, assim como mostrado nas falas dos alunos R7 - “Professora teve boa didática e muito atenciosa para tirar dúvidas” e R2 - “Ajudou muito, principalmente quando tava na dúvida nos exercícios”.

O tema “Aprendizado técnico” (em rosa) evidencia o domínio de conceitos específicos da linguagem Python. Códigos como “estruturas condicionais”, “funções (def)” e “listas e laços de repetição” em resposta a pergunta “Cite um aprendizado ou experiência marcante que você levará deste minicurso” mostram a consolidação de conteúdos práticos, conforme ilustrado por R12 — “a utilização do for, If, Else, elif, while, range” e R15 — “Muito aprendizado sobre comandos de repetição e listas”.

Por fim, o tema “Desafios pontuais” (em vermelho) agrupa ocorrências menos frequentes, como “ficar sem par” ou “diferenças de ritmo”. Embora representem apenas 7% das respostas, esses relatos — como R9: “Fiquei sem par” — indicam limitações logísticas comuns em atividades colaborativas.

6. Ameaças a Validade

A pesquisa apresenta algumas limitações metodológicas que podem afetar a validade dos resultados. A seguir, são descritas as principais ameaças e as estratégias adotadas para mitigá-las, com base em categorias clássicas de validade propostas por Creswell (2017).

Validade interna. Parte dos participantes já havia tido contato prévio com programação durante o semestre letivo, o que pode ter gerado diferenças iniciais entre as duplas. Além disso, a divisão da turma em dois laboratórios resultou em condições físicas e organizacionais distintas. Como mitigação, adotou-se uma abordagem prática padronizada em ambos os grupos e garantiu-se o acompanhamento contínuo dos monitores para reduzir variações no suporte oferecido.

Validade externa. A generalização é limitada pelo contexto: minicurso de curta duração, participação voluntária e realização em uma universidade pública. Para ampliar o potencial de generalização, recomenda-se replicar o relato em instituições com diferentes perfis, em formatos mais longos e com amostras não voluntárias ou mais heterogêneas.

Validade de construção. A interpretação dos construtos pode ter sido afetada pela simplicidade dos instrumentos de coleta (formulários curtos e medidas autorrelatadas), bem como pela possível variação na compreensão das escalas Likert entre os participantes. Para mitigar essas limitações, os instrumentos foram previamente revisados pelos professores tutores do PET, e as instruções foram esclarecidas oralmente no início de cada aplicação. Futuras versões podem adotar instrumentos validados e medidas adicionais de desempenho. Ainda assim, a abordagem prática e o suporte contínuo dos monitores contribuíram para reduzir o impacto dessas variações na experiência formativa, fortalecendo a consistência dos resultados observados.

7. Considerações Finais

Os resultados indicam que a aprendizagem em pares teve impacto positivo no desempenho e na percepção dos participantes. A comparação entre pré e pós-aula mostrou ganhos consistentes nos três dias, com aumento das medianas, menor dispersão e evolução mais uniforme. A média geral de acertos cresceu cerca de 20 pontos percentuais (de 63% para 83%), reforçando a efetividade da abordagem, mesmo diante do caráter introdutório dos conteúdos e da heterogeneidade inicial da turma.

As percepções dos estudantes corroboram esses resultados: a maioria reconheceu que a dinâmica em pares ampliou a compreensão dos conceitos, favoreceu a troca

de conhecimentos, facilitou a validação de raciocínios e ajudou na identificação de erros (elementos centrais para o aprendizado em programação). Também foram relatados efeitos socioemocionais relevantes, como redução da ansiedade, sensação de apoio mútuo e maior motivação para continuar estudando. Esses achados são compatíveis com a literatura, segundo a qual a alternância de papéis, a verbalização do pensamento e a comparação de soluções contribuem para reduzir carga cognitiva e promover a construção ativa do conhecimento.

A análise temática das respostas abertas reforçou a importância do ambiente colaborativo para compreensão e engajamento, embora tenham surgido desafios de pareamento, diferenças de ritmo e timidez, aspectos recorrentes na literatura sobre aprendizagem colaborativa. O professor-mediador foi apontado como essencial para garantir equilíbrio, acolhimento e orientação. Em síntese, a aprendizagem em pares mostrou-se eficaz na introdução de fundamentos de programação, promovendo ganhos cognitivos e benefícios socioemocionais para iniciantes. Os resultados evidenciam seu potencial em contextos formativos semelhantes e indicam a necessidade de novas investigações em conteúdos mais avançados ou com maior diversidade de perfis. Como trabalhos futuros, sugere-se aplicar a proposta em turmas regulares de Introdução à Programação e investigar diferentes estratégias de formação de duplas, analisando efeitos sobre autoeficácia, engajamento e desempenho. A iniciativa contribuiu para práticas mais colaborativas, inclusivas e alinhadas às necessidades de estudantes iniciantes.

8. Agradecimentos

Os autores agradecem ao Programa de Educação Tutorial da UFC Russas pelo apoio institucional e pelo suporte financeiro que viabilizou a realização deste minicurso e desta pesquisa.

Uso de Inteligência Artificial

Ferramentas de Inteligência Artificial Generativa foram utilizadas como apoio à escrita e organização deste manuscrito, sem substituir o julgamento crítico ou decisões metodológicas. Elas foram empregadas para revisão linguística (incluindo correções ortográficas, melhorias de fluidez, redução de redundâncias e ajustes de concisão), apoio à formatação e uso de comandos no \LaTeX , inserção de figuras, referências e links externos. Também auxiliaram na indicação de termos e referências adicionais, na identificação de lacunas da revisão, na elaboração e refinamento de prompts para geração de gráficos quantitativos e na sugestão de trechos de código para análise qualitativa. As ferramentas utilizadas foram ChatGPT (OpenAI) e Grok (xAI), e todo conteúdo gerado foi revisado e validado pelos autores antes de sua incorporação ao texto final.

Referências

- Allen, D. (2015). *Think Python: How to think like a computer scientist*. Green Tea Press.
- Alves, G., Rebouças, A., and Scaico, P. (2019). Coding dojo como prática de aprendizagem colaborativa para apoiar o ensino introdutório de programação: Um estudo de caso. In *Workshop sobre Educação em Computação (WEI)*, pages 276–290. SBC.
- Araújo, T. O., da Silva, S. N. C., de Vasconcelos, P. F. C., and de Miranda, E. C. (2023). Pet capacite-se: Um projeto para apoiar o ensino dos cursos de ciência da computação

- e engenharia de software na ufc campus de russas. In *Workshop sobre Educação em Computação (WEI)*, pages 156–166. SBC.
- Black, P. and Wiliam, D. (1998). Assessment and classroom learning. *Assessment in Education: principles, policy & practice*, 5(1):7–74.
- Boone Jr, H. N. and Boone, D. A. (2012). Analyzing likert data. *The Journal of extension*, 50(2):48.
- Braun, V. and Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative research in psychology*, 3(2):77–101.
- Colin, J., Hoarau, S., and Broisin, J. (2025). Evaluation of different group formation methods in the context of distributed pair programming: Design and experiment in higher education. In *Proceedings of the 30th ACM Conference on Innovation and Technology in Computer Science Education V. 1*, pages 569–575.
- Creswell, J. W. and Creswell, J. D. (2017). *Research design: Qualitative, quantitative, and mixed methods approaches*. Sage publications.
- de Lima, J. R. and de Menezes, C. S. (2024). As dificuldades enfrentadas pelos estudantes na aprendizagem de programação de computadores: Uma revisão sistemática da literatura. *Revista Novas Tecnologias na Educação*, 22(1):130–140.
- Freire, P. (1996). *Pedagogia da autonomia*.
- Korber, P. and Motschnig, R. (2021). The effects of pair-programming in introductory programming courses with visual and text-based languages. In *2021 IEEE frontiers in education conference (FIE)*, pages 1–9. IEEE.
- Lima, G. N., Teles, T. M., Rodrigues, G. L., Girao, A. A., Silva, D. C., de Vasconcelos, P. F., and de Miranda, E. C. (2025). Integração de python no ensino de resistência dos materiais: uma abordagem transdisciplinar para potencializar a aprendizagem em engenharia. In *Workshop sobre Educação em Computação (WEI)*, pages 240–250. SBC.
- McDowell, C., Werner, L., Bullock, H. E., and Fernald, J. (2006). Pair programming improves student retention, confidence, and program quality. *Communications of the ACM*, 49(8):90–95.
- Pereira, F. T., Rosa, N. S., Silva, D. C., Pereira, C. P., and Bittencourt, R. A. (2021). A remote cs0 workshop based on peer learning: Motivation, engagement and self-regulation of novice programmers. In *2021 IEEE Global Engineering Education Conference (EDUCON)*, pages 821–830. IEEE.
- Rahman, M. M., Sharker, M. H., and Paudel, R. (2020). Active and collaborative learning based dynamic instructional approach in teaching introductory computer science course with python programming. In *2020 IEEE Integrated STEM Education Conference (ISEC)*, pages 1–7. IEEE.
- Williams, L. and Kessler, R. R. (2003). *Pair programming illuminated*. Addison-Wesley Professional.