

Explorando a Robótica para mitigar Desafios Comportamentais e de Aprendizado em Programação na Graduação

João Pedro S. Arruda¹, Pedro Henrique D. M. Rocha¹,
Regiane Silva Kawasaki Francês¹, Victor Hugo S. C. Pinto¹

¹Faculdade de Computação (FACOMP)
Universidade Federal do Pará (UFPA) – Belém, PA – Brasil

joao.arruda,pedro.rocha{@icen.ufpa.br}, kawasaki,victor.santiago{@ufpa.br}

Abstract. *Computational thinking has been considered promising because it can foster creativity and problem-solving skills. However, students entering higher education computer science courses often encounter difficulties with abstractions and algorithms. In this study, we describe the use of introductory robotics and interaction with microcontrollers and sensors as tools to facilitate the understanding of programming concepts and exercise behavioral aspects. Eighteen computer science undergraduates participated in the study. As a result, the methodology promoted collaboration among the students and improved their understanding of basic programming concepts.*

Resumo. *O pensamento computacional é um caminho promissor por sua capacidade de fomentar a criatividade e habilidades de resolução de problemas. No entanto, ingressantes em cursos superiores de computação muitas vezes enfrentam dificuldades com abstrações e algoritmos. Neste estudo, descreve-se a utilização da introdução à robótica e a interação com microcontroladores e sensores como ferramentas para facilitar a compreensão de conceitos de programação e exercitar aspectos comportamentais. Dezoito estudantes da graduação em computação participaram do estudo. Como um dos resultados, a metodologia promoveu a colaboração entre os estudantes e melhor compreensão dos conceitos básicos de programação.*

1. Introdução

Atividades que fomentam o pensamento computacional e o aprendizado de programação, como a robótica, desempenham um papel fundamental no desenvolvimento de diversas habilidades, incluindo a capacidade de resolver problemas, comunicar-se efetivamente e colaborar com os outros. Além disso, tais atividades estão ligadas ao currículo de diversas áreas disciplinares [Junior et al. 2020]. A Norma sobre Computação na Educação Básica – Complemento à Base Nacional Comum Curricular (BNCC) foi inicialmente prevista nas Resoluções abrangendo todas as etapas de ensino, da educação infantil até o ensino médio do Brasil [Ministério da Educação (MEC) 2022].

Segundo o relatório recente do Google [Barrence 2023] sobre o panorama de talentos em tecnologia, o ensino de pensamento lógico é defasado nas escolas no Brasil. Isso influencia no afastamento entre os jovens brasileiros e bases de conhecimento para

tecnologia. O relatório da Brasscom [Brasscom 2021] aponta que, anualmente, 53 mil novos profissionais são formados no ensino superior entre 2021 e 2025, um número baixo quando comparado com a demanda por novos talentos neste período que somam aproximadamente 800 mil. Aliado a este cenário, é comum que muitos estudantes de graduação na área da computação se deparem com desafios significativos ao tentar desenvolver suas habilidades em disciplinas essenciais, como programação e algoritmos. Grande parte destes desafios está relacionada à dificuldade em compreender abstrações, fluxo de execução e interpretar a saída gerada a partir da execução de um código.

No estudo de [Medeiros 2019] sobre a aprendizagem de introdução à programação nas universidades, destaca-se a importância de desenvolver três habilidades-chave nos alunos da graduação: cognitiva (resolução de problemas, conclusões lógicas, planejamento e decisões), habilidades sociais (lidar com regras, cooperação e discussão em grupo) e habilidades emocionais (autoconfiança, autoestima e autoavaliação). A natureza colaborativa e interdisciplinar da computação requer que os alunos sejam capazes de expor suas ideias de forma clara, além de colaborar em equipe. Na vida profissional, além de necessitar ter um bom desempenho das habilidades, é necessário lidar com prazos apertados, gerenciar o estresse e manter um equilíbrio saudável.

A robótica surge como uma excelente alternativa, como tem sido considerada uma ferramenta relevante para auxiliar o trabalho colaborativo e o raciocínio lógico, como explorado em [Rubinacci et al. 2017]. Diante disso, uma metodologia envolvendo a introdução da robótica foi adotada, considerando uma nova abordagem prática e orientada a desafios na tentativa de auxiliar os alunos da graduação em desenvolver suas habilidades em programação. O objetivo principal deste estudo foi explorar como a interação com os sensores e placas poderia promover o engajamento dos alunos e facilitar a compreensão dos conceitos de programação. Para isso, adotou-se a plataforma de prototipagem Arduino que utiliza a linguagem C++ com pequenas modificações.

Dezoito graduandos dos primeiros semestres de Ciência da Computação e Sistemas de Informação que apresentavam dificuldades em disciplinas que envolviam o uso da programação, participaram de uma oficina introdutória à robótica, realizada ao longo de seis encontros totalizando dezesseis horas. Durante essas sessões, os alunos trabalharam em equipe para desenvolver algoritmos e criar soluções. Essa iniciativa fez parte de uma atividade piloto de um projeto mais amplo chamado *CombuMaker*, destinado a oferecer educação em programação e robótica para estudantes do ensino médio no Pará. A experiência prévia do projeto com os graduandos permitiu avaliar a proposta e identificar lições importantes, discutidas neste estudo.

2. Trabalhos relacionados

Em [Mota and Neves 2020] as autoras fazem o uso da robótica para fomentar os conhecimentos em lógica de programação e pensamento computacional para graduandos por meio de um projeto chamado REDES, fazendo uso de simuladores virtuais como Autodesk® e Tinkercad™ e também utilizando microcontroladores como Arduino UNO e outros meios *Open Source*, como o *Atmel AVR*, além disso, de forma semelhante, é realizada a coleta de dados e opinião dos alunos por meio de questionários. O estudo indica sucesso entre a união da teoria da programação e conhecimento com a prática aplicada nas soluções criadas pelos estudantes participantes por meio da robótica.

Em [Wiltgen 2022], o autor conduziu um estudo utilizando a robótica como um curso multidisciplinar na graduação de engenharia e no ensino médio, de forma a demonstrar que, utilizando a robótica, o interesse dos alunos pela área aumentaria. Ao término do curso, diversos alunos de graduação relataram interesse em prosseguir seus estudos ou em áreas correlatas, como automação e controle.

No contexto do pensamento computacional, [da Silva Correa et al. 2023] destaca a importância da utilização do Arduino¹ como plataforma de prototipagem e ensino na robótica pelo seu custo acessível e facilidade na incorporação de componentes. Além disso, pelo fato de ser uma plataforma *Open Source*, pode-se programar em vários ambientes uma vez que a linguagem utilizada é baseada em C/C++ e com grande compatibilidade de bibliotecas. Por essas razões, a plataforma Arduino foi escolhida em nosso estudo pelo seu baixo custo e também por ter uma vasta documentação. Porém, os trabalhos divergem no público alvo e no exercício de habilidades comportamentais e reflexões sobre a autoconfiança estimulada por desafios ao longo das interações com os estudantes.

3. Metodologia

Como metodologia, adotou-se elementos da pesquisa qualitativa [Leavy 2022]. Uma revisão da literatura foi conduzida para identificar os desafios de aprendizado enfrentados pelos estudantes no contexto de disciplinas de programação. Posteriormente, uma sequência de atividades baseada em práticas de ensino introdutório de robótica foi elaborada. A coleta de dados ocorreu por meio de observações diretas, questionários e análise de desempenho dos alunos antes e depois das intervenções. A pesquisa objetivou o aprofundamento teórico de situações que emergem espontaneamente na prática profissional, sem revelar dados que poderiam identificar os sujeitos.

O objetivo do estudo concentrou-se em verificar se práticas envolvendo a introdução da robótica computacional contribuem para que alunos da graduação superem dificuldades em programação e desenvolvam suas habilidades comportamentais. Tendo em vista essa meta, dezoito estudantes de semestres variados da graduação em cursos de computação participaram do estudo, com o auxílio de dois tutores e coautores do trabalho, que conduziram os encontros e ficaram disponíveis para esclarecer as dúvidas que eram manifestadas pelos alunos em relação ao conteúdo de programação.

3.1. Planejamento

O estudo foi planejado seguindo as diretrizes dos estudos de [Qian and Lehman 2017], no conteúdo programático do curso *Introduction to Robotics* do *STEM Institute at the City College of New York* [The CCNY STEM Institute 2013], adaptando e removendo tópicos voltados para engenharia para que o enfoque fosse na programação. Ademais, ao realizar questionários e entender a opinião dos alunos, e também unir aulas teóricas e atividades práticas, usou-se como base muito o estudo de [Mota and Neves 2020], de modo a tornar o aprendizado mais factível e lúdico e adaptando mais para o cenário computacional, tendo em vista que todos os participantes do estudo faziam parte de algum curso de computação. Por meio dessas adaptações, buscou-se responder as seguintes perguntas:

QP_1 : Como a utilização de plataformas de prototipagem eletrônica contribui para

¹<https://www.arduino.cc/>

superar as dificuldades dos estudantes com lógica e abstração durante o aprendizado de programação?

QP_2 : Quais são os efeitos da utilização dessas plataformas no comportamento dos estudantes, especialmente em relação à persistência, colaboração e autoconfiança?

Para investigar essas questões, foi desenvolvida uma oficina de introdução à robótica, com foco em dezoito alunos que precisaram manifestar interesse em participar do estudo por e-mail institucional. A oficina foi composta por cinco encontros presenciais e um encontro realizado de maneira remota. Nestas interações atentou-se para três dimensões de aprendizado em programação: conceitual, sintático e de estratégia [Qian and Lehman 2017]. Essas diretrizes contribuíram para a definição da estrutura de cada encontro. Com relação ao conhecimento conceitual, loops, funções e tópicos relacionados com a plataforma Arduino como resistores, jumpers, LEDs, etc foram explorados. Quanto ao conhecimento sintático, o enfoque foi na composição de estruturas de código e suas conexões com a parte física. Por fim, para o estímulo estratégico atentou-se a prática deliberada, em que os participantes ficavam livres para implementar desafios, assumindo que os tutores estariam a disposição para dúvidas.

Na Tabela 1 demonstra-se a estrutura dos encontros e o tempo estimado para cada parte. Na primeira, os instrutores tinham a responsabilidade de transmitir os conhecimentos fundamentais de eletrônica e programação, orientando a execução das atividades e até mesmo fazendo a comparação da linguagem arduino com outras linguagens, como C++ e Python. Na segunda parte, uma tarefa intermediária era atribuída aos alunos para que pudessem aplicar o que foi ensinado, como realizar a conexão e a ativação de um LED, gerando som com um buzzer, entre outras atividades similares. A parte final das aulas foi planejada para sempre possuir desafios usando todos os conhecimentos aprendidos. Assim, os participantes teriam a liberdade para discutir com outras duplas e esclarecer dúvidas com os instrutores. Ao longo desse processo os participantes exercitaram a manipulação de variáveis, laços de repetição e estruturas condicionais.

Tabela 1. Organização dos encontros

Agenda	Tempo estimado
1) Conceitos envolvendo eletrônica, mecânica e programação	60 min
2) Hands on com exercícios e desenvolvimento assistido	40 min
3) Prática deliberada e implementação de desafios	60 min

3.1.1. Preparação

Nesta fase, foi realizada uma avaliação de interesse na Oficina de Introdução à Robótica entre os alunos da Faculdade de Computação, focando principalmente aqueles com desempenho abaixo nas disciplinas de programação. Foram selecionados 24 participantes, que foram informados sobre o objetivo da oficina direcionada a estudantes com dificuldades em programação. Eles foram questionados sobre a aceitação das condições de confidencialidade e anonimidade dos dados fornecidos, que seriam usados exclusivamente para fins de investigação, e receberam a liberdade de se retirarem a qualquer momento. Para formalizar a participação, foi disponibilizado um Termo de Consentimento Livre

e Esclarecido (TCLE). Além disso, foram fornecidos diversos materiais, incluindo: (i) Formulário para coleta de dados e categorização de perfil; (ii) Materiais didáticos; (iii) Exercícios práticos e desafios; (iv) Tutoriais envolvendo testes com placas, circuitos e sensores; (v) Listas de exercícios; e (vi) Formulário de autoavaliação final.

A plataforma de prototipagem eletrônica adotada para realizar o estudo foi Arduino, por conta do baixo custo e a disponibilidade variada de sensores e dispositivos que poderiam ser usados para apresentar o básico da robótica e analisar como ela impactaria no desenvolvimento das habilidades em programação dos alunos. No total, o material que seria passado para cada aluno continha: uma placa Arduino UNO, uma protoboard de tamanho médio, um conjunto de jumpers machos de tamanho variados, um conjunto de leds de cores variadas, um conjunto de resistores de resistências variadas, um potenciômetro, um buzzer, um um conjunto de botões, um servo motor, um sensor infravermelho, uma ponte H, motores simples, um chassi de carrinho de seguidor de linha e um conjunto de pilhas e baterias para a alimentação de energia.

Materiais complementares, como slides e atividades foram desenvolvidos pensando em uma forma dos alunos treinarem seus conhecimentos mesmo fora do ambiente controlado da universidade. Adicionalmente, foi compartilhada uma plataforma online Wokwi², o qual permitia que os participantes fizessem o uso da placa Arduino UNO em casa, junto com seus componentes também foi fornecido um canal pelo WhatsApp para esclarecimento de dúvidas. Um Quiz foi elaborado utilizando a plataforma Quizizz³, que visava verificar o aprendizado dos participantes após os encontros. Essa ferramenta contribuiu para o engajamento dos participantes por meio da gamificação [de Sousa et al. 2022]. Além disso, foi realizado o plano de ensino como é possível visualizá-lo na Tabela 2, que representava como a oficina deveria ser apresentada aos participantes e qual assunto iria ser discutido em cada encontro, para assim desenvolve-los introduzindo assuntos da programação.

Tabela 2. Plano de Aulas

Aulas

- 1) Tipos de dados, Operadores, Funções, Const, Comentários;
 - 2) Introdução à Robótica, Protoboard, jumpers, linguagem Arduino, uso de Leds;
 - 3) Revisão, Resistores, sinal Pwm, Buzzer, Push Button, Programação de componentes;
 - 4) Revisão, Sensores de refletância, Uso de Switch Case, Servo Motor, programação de componentes;
 - 5) Revisão, Ponte H, Motores, Função Random, programação de componentes, desafio final;
 - 6) Encontro de desenvolvimento livre;
-

3.1.2. Categorização de perfil

A oficina iniciou com 24 participantes mas somente 18 permaneceram, isto é, com uma frequência mínima de 75% nos encontros executados. O perfil dos dezoito participantes do estudo foi definido a partir da coleta de dados por meio de um formulário⁴. A ideia dessa categorização foi entender a condição primária que os participantes iniciaram

²<https://wokwi.com/arduino>

³<https://quizizz.com/>

⁴<https://forms.gle/2GE3QUurmHPVVrn7>

a oficina e identificar as maiores dificuldades em relação às disciplinas de programação dos participantes e assim, elaborar aulas e materiais adequados para suprir suas dificuldades de aprendizado. Além disso, avaliou-se como importante definir uma base de como eles julgavam suas habilidades de programação antes do início da oficina. Na Tabela 3 descrevem-se as perguntas utilizadas neste formulário inicial.

Tabela 3. Questões utilizadas para a categorização de perfil

I_1) Até o momento, você tem encontrado facilidade nas disciplinas de programação?
I_2) Você compreendeu facilmente os conceitos de variáveis, condicionais e loops?
I_3) As aulas de programação envolveram principalmente atividades práticas?
I_4) A metodologia empregada nas disciplinas de programação era de fácil compreensão?
I_5) Assumindo que na programação, a abstração sugere a distinção entre “o que” o programa faz e “como” ele é implementado. O conceito de abstrações foram fáceis de compreender, conseguindo avançar sem dificuldades na implementação do código (ou pseudocódigo)?

Primeiramente, os participantes foram direcionados a questionamentos como: “Qual é seu curso?” e “Qual semestre está cursando no momento?”. Com estes dados pode-se inferir a maioria dos participantes (70,8%) pertenciam ao curso de Ciência da Computação. Ainda, a maior parte dos alunos encontravam-se no segundo semestre (58,3%), tendo então, contato com as disciplinas Algoritmos e Programação I. Adicionalmente, foram fornecidas as questões I_1 à I_5 (descritas na Tabela 3) para a categorização de perfil, as quais abordam as dificuldades encontradas em determinado aspecto em seus aprendizados na graduação. Vale ressaltar que tais perguntas foram respondidas por meio de uma escala Likert, comumente utilizada para medir percepções, variando de 1 (um) a 5 (cinco), para “discordo completamente” e “concordo completamente”.

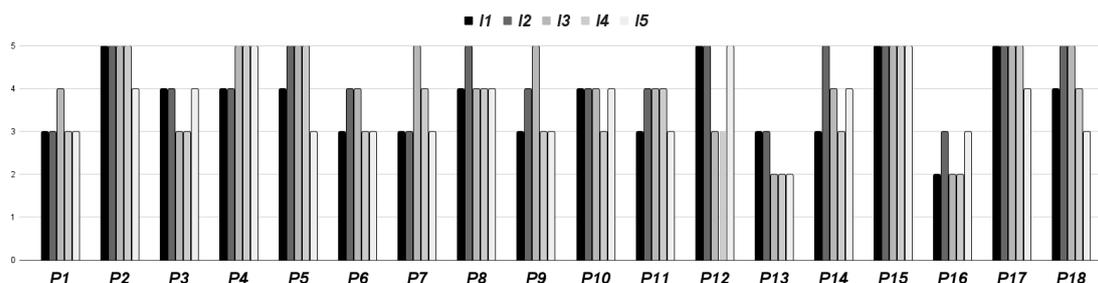


Figura 1. Categorização do perfil dos participantes

Na Figura 1 ilustra-se a sumarização das respostas dos participantes. Um dos pontos revelados por essa categorização é que poucos participantes admitiram ter dificuldades em assuntos que envolviam a parte da programação, fato que posteriormente se revelaria como uma superestimação de habilidades em programação, uma questão discutida com mais detalhes na seção 3.3. Contudo, outros participantes foram mais coerentes nas suas autoavaliações e revelaram que tinham dificuldade alta ou mediana em relação a compreensão dos conteúdos das disciplinas que envolviam o uso da programação e que a falta de aulas mais práticas dificultavam o aprendizado.

3.1.3. Formulário de autoavaliação final

Ao final da oficina, um questionário⁵ de autoavaliação foi elaborado para ser fornecido aos participantes. Na Tabela 4 descrevem-se as perguntas utilizadas (Q_1 a Q_{12} e D_1 a D_3). As questões de Q_1 a Q_{10} utilizam a escala Likert para que participantes atribuíssem valor numérico às suas respostas, de forma a representar uma taxa entre “*concordo completamente*” (variável máxima, no valor 1) e “*discordo completamente*” (variável mínima, no valor 5). As perguntas Q_{11} e Q_{12} utilizam a escala VAS (*Visual Analogue Scale*) para permitir maior precisão em relação autoavaliação dos participantes em relação as habilidades antes e depois da oficina, enquanto as questões de D_1 a D_3 eram discursivas.

As questões de Q_1 a Q_{12} buscaram entender como havia sido a experiência da oficina para os participantes e de acordo com suas percepções, a mesma contribuiu para superar suas dificuldades com a lógica de programação, como: loops, laços de repetição e variáveis. As questões (Q_{11} , Q_{12}) foram propostas para que os participantes realizassem uma auto-avaliação quanto as suas habilidades em programação antes e depois do estudo, a taxa de variação de tais questões representavam um valor de 1 a 10 para atribuir na resposta com maior nível de granularidade. Além disso, foram disponibilizadas questões discursivas (D_1 , D_2 e D_3) para que os participantes respondessem com suas palavras sobre suas experiências e opiniões sobre o estudo. As respostas destes questionamentos são mais detalhadas na subseção 3.3.

Tabela 4. Questões para autoavaliação após as intervenções

Q_1) A oficina de robótica contribuiu para entender melhor os conceitos de variáveis, condicionais e loops?
Q_2) As aulas da oficina foram bastante práticas?
Q_3) As partes práticas desafiaram suas habilidades como programador?
Q_4) A parte teórica foi de fácil entendimento?
Q_5) A quantidade de encontros poderia ter sido maior?
Q_6) Faltou mais clareza nas explicações dos instrutores?
Q_7) O material disponibilizada foi de fácil entendimento e explicava com clareza os componentes e habilidades necessárias para realizar os desafios propostos?
Q_8) Ao realizar as atividades em dupla/grupo durante a oficina, isso contribuiu para melhorar suas habilidades em termos comportamentais (trabalho em equipe, autoconfiança e liderança)?
Q_9) Ao trabalhar em equipe, você notou melhorias em termos de habilidades comportamentais, como programador, ao trocar ideias e opiniões com outras pessoas?
Q_{10}) Após a metodologia da oficina, você se sente mais capaz em resolver questões que exploram o raciocínio lógico e programação?
Q_{11}) Antes de participar dos encontros, qual nota você atribuiria às suas habilidades de programação?
Q_{12}) Após a oficina, como você avalia seu nível de habilidades em programação?
D_1) Você acha que materiais como Arduino, ESP-32, entre outras, deveriam ser utilizados nas disciplinas regulares de programação? Explique
D_2) Você acha que a oficina contribuiu positivamente para lhe proporcionar desafios como estudante de computação? Como?
D_3) Quais foram suas considerações finais sobre a oficina? (o que achou, o que pode melhorar, etc)

3.2. Execução

Para a condução do estudo, os alunos foram organizados em duplas, mantendo-se nessa configuração até o término da oficina. Essa abordagem foi adotada com o intuito de

⁵<https://forms.gle/tDLgyvTC7bwNW3rZ8>

otimizar a utilização dos recursos disponíveis e fomentar a interação entre os estudantes, visando o desenvolvimento das habilidades deles, tanto em programação, com a comunicação entre as duplas quanto as habilidades interpessoais e criativas, para estimular a capacidade de resolver problemas em equipe. Por meio dessa metodologia, notou-se que os alunos expressavam dúvidas mais elaboradas e precisas, enquanto aqueles com mais habilidades em programação auxiliavam os colegas com dificuldades mais básicas. Entretanto, houveram participantes que durante a execução da oficina, relataram preferência pelo trabalho individual e que não ficaram muito satisfeitos com o trabalho em grupo, parte que será mais detalhada na seção 3.3.

Nos encontros, os participantes tinham liberdade para pedir o suporte dos instrutores, tanto durante as aulas quanto nos desafios, para esclarecer dúvidas relacionadas ao código e às habilidades manuais necessárias para montar as atividades propostas. Em todos os encontros, haviam momentos criativos, os quais os alunos eram encorajados a aplicar suas habilidades.

3.3. Resultados

Com base nas respostas para as questões I_1 e I_2 do formulário de categorização de perfil, notou-se os participantes tinham algum conhecimento em programação. No entanto, muitas informações fornecidas não correspondiam à realidade durante os encontros, ou seja, muitos superestimaram suas habilidades e precisaram de mais intervenções dos tutores durante as atividades de programação.

Com relação ao formulário de autoavaliação final, os resultados demonstraram uma percepção diferente da apresentada inicialmente por parte dos participantes, ou seja, os alunos atribuíram notas menores para seus conhecimentos prévios em programação nas respostas da (Q_{11}) e aumentaram suas notas após a oficina nas respostas da (Q_{12}), como é visível na Figura 2. Este resultado foi considerado relevante, pois evidencia que a distorção da realidade, ou seja, a “falsa sensação” de domínio em lógica de programação, pode influenciar a disposição dos participantes em buscar novos aprendizados.

Com base nas respostas para as perguntas D_1 , D_2 e D_3 do formulário de autoavaliação, pode-se sumariá-las da seguinte forma:

- “Você acha que materiais como Arduino, ESP-32, entre outras, deveriam ser utilizados nas disciplinas regulares de programação? Explique” Todos os participantes foram favoráveis e mencionaram que a plataforma Arduino contribui com desenvolvimento prático das aulas ao ver de forma tangível sua evolução durante o processo de aprendizado. Isso mostra a possibilidade de aderir a introdução de conceitos da robótica como novo método e ferramenta de ensino da programação.
- “Você acha que a oficina contribuiu positivamente para lhe proporcionar desafios como estudante de computação?”, 90% dos participantes concordaram, destacando que a novidade em manipular o Arduino e utilizar os conceitos de eletrônica junto com a programação despertaram sua curiosidade e o desafio de trabalhar em equipe para finalizar as tarefas, o que mostra a efetividade do estudo.
- “Quais foram suas considerações finais sobre a oficina?”, 100% das respostas foram positivas, mas alguns participantes (P_2 , P_3 , P_{10} , P_{13} E P_{17}) mencionaram a importância de continuar a iniciativa e aumentar o número de encontros. Outros (P_8 e P_{14}) mencionaram que os desafios poderiam ser mais aprofundados.

Com relação a questão Q_9 , sobre a experiência de trabalhar em equipe e se este método contribuiu para o seu aprendizado, a maioria dos participantes se sentiram confortáveis ao trabalhar em grupo, atribuindo notas 4 e 5 na escala, totalizando 78,9%.

Na Figura 2 ilustra-se a variação da percepção das habilidades em programação dos alunos antes da oficina (Q_{11}) e depois da oficina (Q_{12}), onde observa-se que a maioria dos participantes deram uma nota menor para suas habilidades iniciais e se avaliaram como programadores melhores no final, o que mostra uma evolução positiva reportada por mais de 88% dos participantes do estudo.

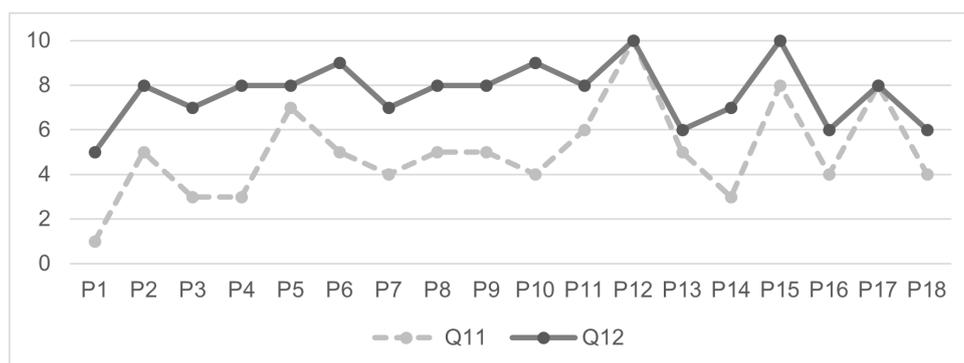


Figura 2. Perspectiva dos participantes quanto às suas habilidades em programação (Q_{11} e Q_{12})

QP_1 : Como a utilização de plataformas de prototipagem eletrônica contribui para superar as dificuldades dos estudantes com lógica e abstração durante o aprendizado de programação?

O uso de plataformas de prototipagem eletrônica contribuiu ao fornecerem um método de visualização melhor e mais interativo aos estudantes que possuíam dificuldade em se conectar com as disciplinas de programação e entender como os seus códigos realmente funcionavam ao invés de se restringir a somente a saída do compilador. Isso é corroborado aos participantes relatarem na pergunta D_2 do último formulário que além da contribuição em programação, a oficina fez eles refletirem que precisavam rever os conceitos básicos, além disso, ela melhorou o engajamento nas atividades em equipe ao promover o trabalho colaborativo e compartilhamento de opiniões.

A utilização de plataformas de prototipagem eletrônica, como o Arduino e seus componentes, desempenharam um papel fundamental para que os alunos superassem as dificuldades com lógica e abstração superassem durante a oficina. Por exemplo, ao acender um LED e criar um semáforo os alunos visualizavam diretamente o efeito de suas atribuições e desenvolvimentos de funções por meio da intervenção dos microcontroladores. Além disso, os equívocos cometidos no código também refletiam no comportamento e leitura incorreta dos sensores, ou seja, os estudantes poderiam visualizar seus erros de forma mais assertiva e corrigi-los. Assim, os participantes eram constantemente encorajados a encontrarem o defeito no código e refletirem de forma lógica para sua correção.

Embora as respostas as perguntas dissertativas tenham sido variadas, notou-se a preferência pelo uso de materiais tangíveis pois ajudam a simular um determinado comportamento do implementado em código gerando movimentos e mudanças de estados nos

dispositivos ligados na placa do Arduino. Juntamente com as atividades realizadas pela plataforma Quizizz, pode-se notar uma evolução no conhecimento dos estudantes também em outras linguagens, como C++ e Python.

QP_2 : Quais são os efeitos da utilização dessas plataformas no comportamento dos estudantes, especialmente em relação à persistência, colaboração e autoconfiança?

Com base na observação direta e dados coletados no formulário de autoavaliação pode-se notar que os participantes desenvolveram suas habilidades de comunicação no decorrer das aulas, com outros colegas, principalmente durante os desafios na sala de aula, quando precisavam se unir para resolver os problemas e tirarem as dúvidas, além disso, no formulário de autoavaliação, os participantes relataram que se sentiram mais motivados em alcançar os objetivos dos desafios propostos. Ademais, foi possível ver os alunos compartilhando mais experiências durante as atividades em sala. Com as respostas fornecidas para Q_8 do formulário de autoavaliação, quatorze alunos atribuíram uma nota positiva (5) e quatro alunos atribuíram nota (4), o que demonstra que todos os alunos consideram que o trabalho em equipe seguindo a metodologia proposta contribuiu para melhorar suas habilidades em comunicação, resolução de problemas e trabalho em equipe. Além disso, nas análises da Q_9 , nove participantes atribuíram (5) e seis atribuíram (4). Dessa forma um total de quinze participantes consideram que a organização em duplas contribuiu para a evolução no aprendizado. Adicionalmente, 11 participantes avaliaram a oficina com nota máxima (5) e afirmaram que as atividades práticas propostas desafiaram suas habilidades como programadores ao analisar os resultados do Q_3 . Ou seja, a oficina ajudou a tirá-los da zona de conforto em relação ao aprendizado de programação.

Em relação às respostas no item D_2 , diversos alunos ($P1, P3, P5, P6, P8, P9, P11$ e $P16$) mencionam que a metodologia adotada permitiu-lhes uma correlação mais efetiva entre a sintaxe e a lógica de programação. Destacou-se também que o ambiente estimulou sua criatividade, promoveu o desenvolvimento de habilidades em trabalho em equipe e aumentou seu engajamento na compreensão dos conceitos relacionados à programação.

4. Conclusões

Como conclusão do estudo notou-se que métodos que estimulam o trabalho em equipe e ajudam a superar os desafios com habilidades não técnicas e abstrações são necessários para melhorar o aprendizado em programação. Em geral, os participantes aprimoraram suas habilidades de programação e entenderam melhor a lógica de sensores e dispositivos conectados. A cultura maker se mostrou positiva para o desenvolvimento das habilidades cognitivas necessárias na programação. Além disso, alguns participantes mudaram suas percepções sobre suas habilidades de programação.

O estudo proporcionou uma compreensão mais profunda do impacto da robótica no desenvolvimento das habilidades de programação e comportamento dos alunos. Como perspectivas futuras pretende-se explorar outras plataformas de prototipagem como o NodeMCU ESP32 para construção de soluções IoT (Internet das Coisas) e realizar estudos experimentais no contexto de ensino de programação na graduação.

Agradecimentos

A Pró-Reitoria de Extensão (PROEX) da UFPA pelo suporte financeiro (Proex N.º 04/2023 - Navega Saberes) e aos participantes do CombuMaker.

Referências

- Barrence, A. (2023). Panorama de talentos em tecnologia. Acesso em 10 de out., 2023.
- Brasscom (2021). Demanda de Talentos em TIC e Sigma TCEM v117. Acesso em 10 de out., 2023.
- da Silva Correa, R. B., Teixeira, N. G., Portilho, F. J., Junior, C. X. P., and Aranha, R. V. (2023). A formação em computação e a (falta de) acessibilidade em sistemas computacionais: acaso ou resultado? In *Anais do XXXI Workshop sobre Educação em Computação*, pages 488–498. SBC.
- de Sousa, R. T., de Azevedo, I. F., and Alves, F. R. V. (2022). Quizizz como estratégia de gamificação no ensino de geometria plana. *Revista Docentes*, 7(19):41–48.
- Junior, J. B. B., Piedade, J. M. N., Wunsch, L. P., and Medeiros, L. (2020). Formação no contexto do pensamento computacional, da robótica e da inteligência artificial na educação.
- Leavy, P. (2022). *Research design: Quantitative, qualitative, mixed methods, arts-based, and community-based participatory research approaches*. Guilford Publications.
- Medeiros, R. P. (2019). Hello, world: uma análise sobre dificuldades no ensino e na aprendizagem de introdução à programação nas universidades.
- Ministério da Educação (MEC) (2022). Base Nacional Comum Curricular (BNCC) de Computação.
- Mota, L. and Neves, I. (2020). Robótica como ferramenta para o desenvolvimento do pensamento computacional e introdução a lógica de programação. In *Anais do XXVIII Workshop sobre Educação em Computação*, pages 141–145, Porto Alegre, RS, Brasil.
- Qian, Y. and Lehman, J. (2017). Students’ misconceptions and other difficulties in introductory programming: A literature review. *ACM Transactions on Computing Education (TOCE)*, 18(1):1–24.
- Rubinacci, F., Ponticorvo, M., Passariello, R., and Miglino, O. (2017). Robotics for soft skills training. *Research on Education and Media*, 9(2):20–25.
- The CCNY STEM Institute (2013). Intro robotics - syllabus 2013. Online. Documento de curso.
- Wiltgen, F. (2022). Robótica prática como ferramenta mãos à obra no ensino. *Revista de Engenharia e Tecnologia*, 14(2).