



ESTIMULANDO O DESENVOLVIMENTO DO PENSAMENTO COMPUTACIONAL POR MEIO DO ENSINO DE PROGRAMAÇÃO BASEADO NO MÓDULO PBL

DOI: 10.37702/2175-957X.COBENGE.2024.5220

Autores: EMANUELE DUARTE SILVA, ANDRÉ OLIVEIRA CARVALHO DA SILVA, ELEN PRISCILA DE SOUZA LOBATO, WELLINGTON DA SILVA FONSECA

Resumo: O Pensamento Computacional, essencial no século XXI, permite que os estudantes desenvolvam habilidades de resolução de problemas. Essas competências podem ser aprimoradas por meio de plataformas online que permitem planejar, executar e aperfeiçoar projetos de forma interativa e visual. Integradas ao Aprendizado Baseado em Projetos (PBL), essas ferramentas tornam a aprendizagem ainda mais eficaz. Neste projeto, utilizou-se o módulo PBL para ensinar jovens e adolescentes lógica de programação e desenvolver o pensamento computacional. Foi combinada a utilização do Scratch e do Micro:bit/MakeCode em um ambiente de PBL, promovendo um aprendizado ativo, colaborativo e iterativo. Nessa abordagem, os alunos podem experimentar, receber feedback imediato e melhorar continuamente seus projetos. A aplicação dessa metodologia mostrou-se eficaz, proporcionando resultados positivos no ensino de programação e no desenvolvimento do pensamento computacional entre os alunos.

Palavras-chave: Pensamento computacional, Aprendizagem com base em projetos, Scratch, MakeCode, Algoritmo, Micro:bit.

ESTIMULANDO O DESENVOLVIMENTO DO PENSAMENTO COMPUTACIONAL POR MEIO DO ENSINO DE PROGRAMAÇÃO BASEADO NO MÉTODO PBL

1 INTRODUÇÃO

A Máquina de *Turing* é um conceito abstrato desenvolvido pelo matemático Alan Turing. O próprio Turing, conceituou que a sua máquina seria capaz de operar de acordo com um conjunto finito de estados e regras de transição, determinando como ela deve se comportar em cada situação, ou seja, guiada por algoritmos (Silva, 2023). Através desse conceito, *Turing* demonstrou que, teoricamente, essa máquina abstrata poderia resolver qualquer problema computável, desde que fosse dada uma descrição precisa do algoritmo a ser executado (De Toni, 2022).

Ao entender os princípios da Máquina de *Turing*, percebe-se que ela representa um modelo abstrato de como um computador opera e, por meio desse entendimento, é possível aplicar os princípios da computação para resolver problemas em diversas áreas. Esse é o princípio básico do Pensamento Computacional, desenvolvido pela pesquisadora Jeannette M. Wing (2006). Segundo Wing (2006 *apud* Silva, 2021), esse pensamento envolve a formulação de problemas complexos e a capacidade de decompor os mesmos em etapas menores, como forma de identificar padrões, gerenciá-los, abstrair informações relevantes e criar algoritmos eficazes para resolver esses problemas.

No cenário contemporâneo, a importância do Pensamento Computacional se destaca, especialmente quando combinado com avanços tecnológicos. Essa junção tem se mostrado essencial na introdução de novos métodos destinados a facilitar e potencializar os processos de aprendizagem (Pasinato *et al.*, 2020). Com o uso de tecnologias educacionais, como ambientes de programação visual e componentes de robótica, é possível criar experiências de aprendizagem mais interativas e envolventes (Sousa, 2022). Tais ferramentas permitem que os alunos desenvolvam habilidades de Pensamento Computacional de forma prática e *hands-on* (em tradução literal, "mão na massa"), explorando conceitos como algoritmos, abstração, solução de problemas e trabalho em equipe (Shute, 2017).

Além de utilizar a programação como ferramenta, é importante transmitir esse conhecimento de forma eficaz e buscar maneiras de facilitar o processo de aprendizagem (Pereira, 2020). Nesse contexto, o uso de metodologias ativas, como o Aprendizado Baseado em Projetos (PBL, sigla em inglês de *Problem Based Learning*), surge como uma abordagem valiosa.

Com a metodologia PBL os alunos são desafiados a identificar e analisar questões relevantes, colaborar com seus colegas, realizar pesquisas, desenvolver habilidades de resolução de problemas e apresentar suas conclusões de maneira eficaz (Oliveira, 2020). Dessa forma, ao integrar o PBL no ensino de programação e Pensamento Computacional, os educadores podem proporcionar uma experiência de aprendizado mais engajadora e significativa, preparando os alunos para enfrentar os desafios do mundo real com confiança e competência.

Rodrigues (2023) destaca que as Universidades Federais, devem reconhecer a importância do Pensamento Computacional e está engajada no desenvolvimento de projetos voltados para promovê-lo, entre seus alunos, preparando-os para os desafios do mundo moderno. Nessa perspectiva, é evidente que as metodologias ativas vêm

ganhando destaque nas salas de aula, suplantando o ensino tradicional ao longo dos últimos anos (Freitas *et al*, 2020).

Dentro desse contexto, como forma de fomentar a adoção de metodologias ativas e estimular o Pensamento Computacional, o Programa de Educação Tutorial de Engenharia Elétrica (PET-EE) de uma Universidade Federal do Norte do Brasil, desenvolve projetos de ensino e extensão com esse foco. Para isso, são ofertados cursos *online* e presenciais, tanto para alunos da própria Universidade, quanto para pessoas externas.

Mediante ao exposto, o objetivo deste trabalho é descrever os métodos utilizados para introduzir o Pensamento Computacional, em conjunto com conceitos básicos de eletrônica, em uma turma de Introdução à Robótica. A turma que será apresentada é composta por jovens e adolescentes em um Centro Social externo à Universidade, e o curso teve a duração de 4 meses.

O restante do trabalho está estruturado da seguinte forma: na seção 2 será apresentada a metodologia utilizada no curso ministrado; na seção 3 tem-se os resultados obtidos e, por último, na seção 4, as considerações finais.

2 METODOLOGIA

Nesta seção será descrita a metodologia utilizada durante o desenvolvimento do curso de Introdução à Robótica de uma turma de jovens e adolescentes, realizado em um Centro Social e Educacional da cidade de Ananindeua, região metropolitana de Belém do Pará. As aulas foram ministradas no laboratório de informática do centro equipado com x computadores, 1 projetor, cadeiras e mesas. Para esse curso foram realizadas aulas semanais, durante o período de 3 meses, sendo cada aula com duração de 2 horas. A turma era composta por 6 alunos, com idades entre 13 e 17 anos. O número reduzido de participantes deveu-se às limitações de espaço e à disponibilidade de equipamentos para o curso.

Para o curso em questão, utilizou-se as plataformas *Scratch* (Costa *et al*. 2022) e *Make Code*, em conjunto com a placa *Micro:Bit* (Brandão, 2024), para o ensino da programação e o desenvolvimento do Pensamento Computacional. Dessa forma, o curso foi dividido em dois módulos distintos. Sendo que no primeiro módulo, foi utilizada a plataforma *Scratch*, e no segundo módulo foram utilizadas as plataformas *Micro:Bit* e *Make Code*. Em cada módulo foram abordados diferentes assuntos, conforme ilustra a Figura 1.

Figura 1 – Módulos e assuntos abordados no curso Introdução à Robótica.



Fonte: Os autores.

O primeiro módulo, com a plataforma *Scratch*, teve como objetivo introduzir a lógica de programação por meio de blocos e uma compreensão básica dos fundamentos da programação. Para isso foram abordados assuntos como operadores matemáticos (soma, subtração, divisão, igualdade, maior que, menor que, *and*, *or* e *not*), estruturas condicionais (*if* e *else*) e laços de repetição (*for* e *while*).

Já o segundo módulo, com as plataformas *Micro:Bit* e *Make Code*, teve como objetivo aplicar de forma prática os conceitos adquiridos no primeiro módulo. Tal objetivo foi possível, principalmente por conta do uso do *Micro:Bit*, com o qual os alunos aprenderam sobre os sensores integrados na placa e foram introduzidos ao uso de botões. Além disso, utilizando a plataforma *Make Code* os alunos aprenderam sobre a criação de jogos, proporcionando uma experiência prática e divertida para consolidar os conhecimentos adquiridos.

A cada aula um novo assunto foi abordado. Dessa forma, em todas as aulas, optou-se por fazer uma introdução teórica sobre os conteúdos que seriam abordados, uma vez que, para a maioria dos alunos era a primeira vez que eles estavam tendo contato com determinados assuntos. As aulas eram expositivas, onde os monitores utilizavam *slides* e projetor para ministrar as aulas, conforme mostra a Figura 2. O conteúdo era apresentado de forma introdutória, através de explicações teóricas e exemplos demonstrados pelos monitores, de acordo com o tema da aula.

Figura 2 – Aula de apresentação da plataforma *Scratch*:

(a) Parte expositiva da aula (b) Esclarecimento de dúvidas dos alunos.



Fonte: Os autores.

Posteriormente, era apresentado um “desafio” para os alunos resolverem. Os desafios foram inseridos nas aulas com o objetivo de motivar os alunos a identificar problemas e desenvolver métodos para resolvê-los. Como forma de auxiliar os alunos na resolução do desafio, era fornecido um guia inicial para desenvolver o algoritmo durante a fase do projeto. A exemplo desses desafios, têm-se:

1. Crie um programa que pergunte a altura do usuário e diga se ele está apto ou não a ir no brinquedo (Altura máxima: 180 cm / Altura mínima: 100 cm);
2. Crie um programa que pergunte 2 números ao usuário e informe se a soma deles é ímpar ou par;
3. Crie um programa que avalia o nível de luminosidade e faça o *Micro:Bit* acender por inteiro caso este valor esteja abaixo de 120;

4. Crie um termômetro que mostra a temperatura em graus *Celsius* (ao apertar o botão a), *Fahrenheit* (ao apertar o botão b) e *Kelvin* (ao apertar ambos os botões).

Como forma de aplicar a metodologia PBL, após a apresentação do desafio e leitura do guia de orientação, os alunos eram encorajados a desenvolver o seu próprio algoritmo para solucionar o desafio proposto, conforme ilustra a Figura 3.

Figura 3 - Resolução de um desafio pelo alunos:

- (a) Criação do algoritmo de solução na plataforma *Scratch* (b) Execução do algoritmo no *Micro:Bit*



Fonte: Os autores.

O método PBL foi adotado devido à sua eficácia e atratividade para os alunos (Ybarra, 2022). Com essa abordagem, observou-se uma maior participação dos alunos e os incentivou a buscar conhecimento de forma autônoma. Além disso, observou-se, ainda, que os alunos tornaram-se mais independentes na formulação de soluções para problemas, o que contribuiu significativamente para o desenvolvimento do Pensamento Computacional. Ao decompor os problemas e solucioná-los passo a passo, os alunos puderam fortalecer suas habilidades nessa área.

3 RESULTADOS E DISCUSSÕES

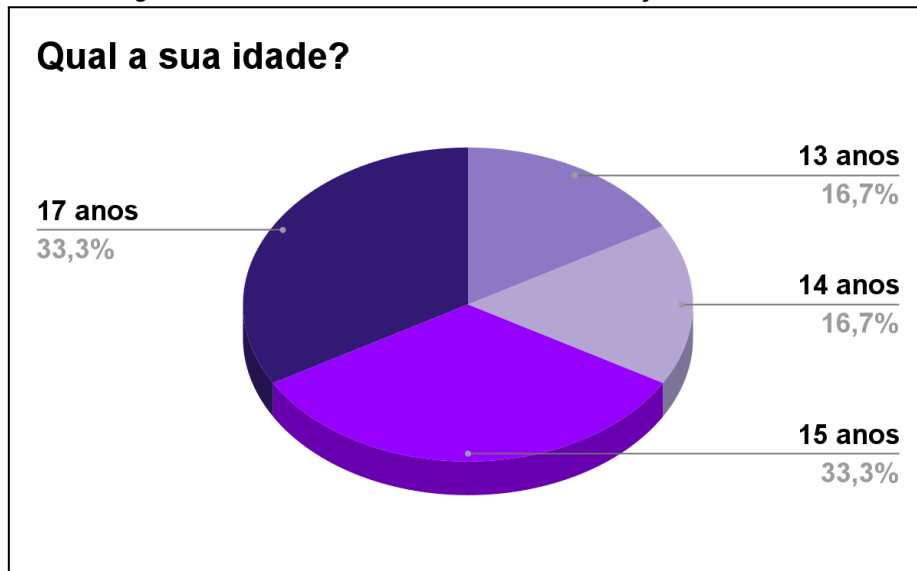
Ao longo do curso foram aplicados formulários para coleta de dados dos alunos. Os formulários foram elaborados no *Google Forms* e disponibilizados *online*. O primeiro formulário foi aplicado, após a inscrição dos alunos, como forma de identificar e traçar o perfil dos alunos que participaram do curso. Os demais formulários foram aplicados no término de cada módulo, como forma de receber o *feedback* dos alunos em relação ao conteúdo ministrado. A seguir, serão apresentados uma parte dos dados coletados nos formulários.

3.1 Perfil dos alunos

Os alunos participantes do curso possuíam entre 13 e 17 anos, conforme ilustra o gráfico da Figura 4. Ao todo, participaram 6 alunos. Entre eles, 2 alunos tinham 17 anos, o que representa 33,3% do grupo, e outros 2 alunos tinham 15 anos, também representando 33,3% do total. Dos dois alunos restantes, um tinha 14 anos e o outro tinha

13 anos, cada um representando 16,7% dos participantes. Ao serem expostos a essas habilidades na juventude, os alunos se tornam mais aptos a enfrentar desafios e aproveitar as oportunidades do mundo moderno, desenvolvendo-se como indivíduos mais competentes e confiantes.

Figura 4 - Idade dos alunos da turma Introdução à Robótica.



Fonte: Os autores.

Outro dado importante a ser apresentado está no quantitativo de alunos do gênero feminino participando do curso, conforme mostra o gráfico da Figura 5. Ao todo 50% dos alunos participantes do curso eram do gênero feminino e 50% eram do gênero masculino. Um ponto interessante a ser apresentado, já que em edições anteriores desse mesmo curso no mesmo local, as turmas apresentavam um maior quantitativo de meninos. Demonstrando assim, que esse cenário está mudando e mais mulheres demonstram interesse pela área de tecnologia.

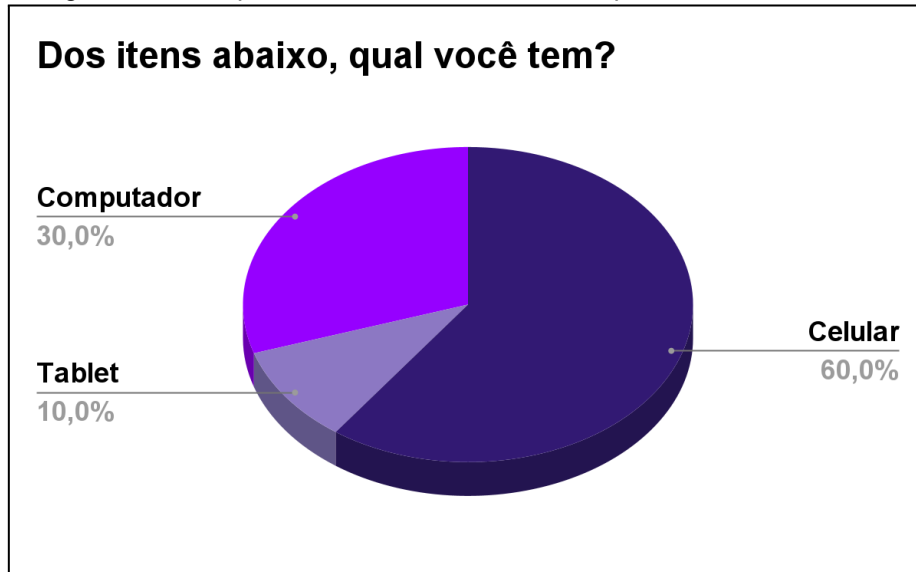
Figura 5 - Quantitativo de meninas e meninos no curso.



Fonte: Os autores.

Quanto ao acesso à internet por meio de um dispositivos eletrônicos, como computador, *tablet*, ou celular, a maioria dos alunos afirmaram acessar a internet por meio de um celular. De acordo com o gráfico da Figura 6, 60% acessam a internet através de celular, 30% através de um computador e 10% através de *tablet*. Essa informação foi útil para os monitores do curso, que puderam direcionar mais atenção para auxiliar os alunos no manuseio do computador, já que poucos deles tinham familiaridade prévia com esse tipo de dispositivo.

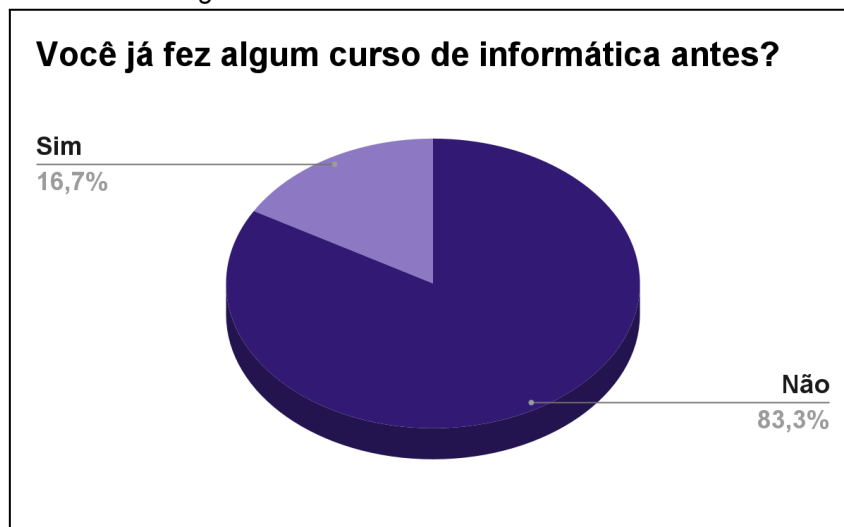
Figura 6 - Itens que os alunos afirmaram utilizar para acessar a internet.



Fonte: Os autores

Dos alunos que participaram do curso, apenas 16,7% afirmaram ter realizado algum curso relacionado à informática, como apresentado no gráfico da Figura 7. Entretanto, nenhum possuía conhecimentos em lógica de programação e construção de algoritmos.

Figura 7 - Cursos realizados anteriormente.



Fonte: Os autores.

Assim, foi possível obter informações sobre o público-alvo, o que permitiu um direcionamento mais preciso durante as aulas, em relação aos assuntos e temas que seriam abordados ao longo do curso. Esse conhecimento prévio visou melhorar a compreensão de acordo com a realidade individual de cada aluno.

3.2 Avaliação dos módulos

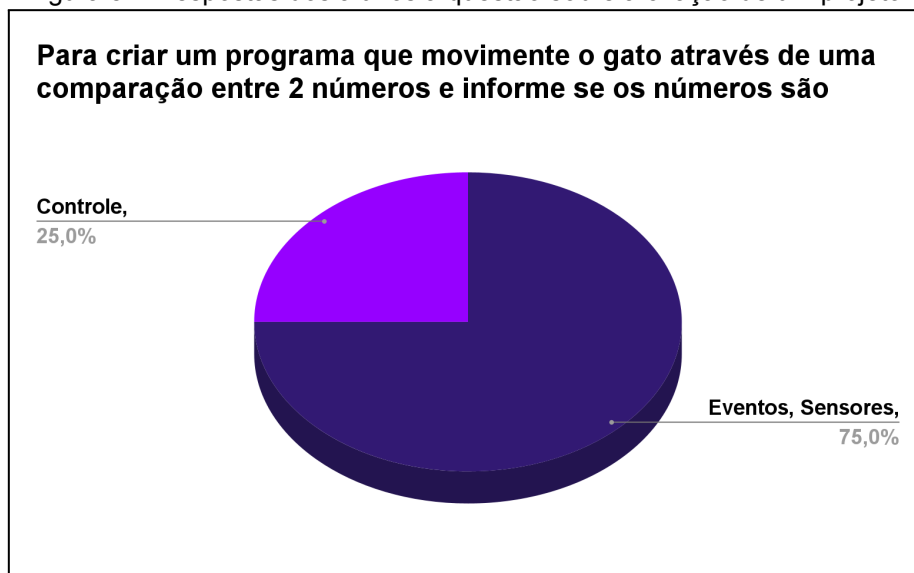
Ao final de cada módulo, sendo o primeiro de *Scratch* e o segundo de *Micro:Bit* juntamente com *MakeCode*, os alunos preencheram mais um formulário. Como forma de avaliar o módulo ministrado. Através deste formulário os alunos deveriam informar as dificuldades enfrentadas ao longo do módulo. Além disso, os alunos responderam algumas perguntas sobre o conteúdo de lógica de programação ministrado durante as aulas.

Primeiro Módulo: Scratch

Com o intuito de avaliar o conhecimento adquirido pelos alunos durante o módulo, fez-se perguntas relacionadas aos conteúdos ministrados. O principal aspecto a ser avaliado, é a lógica de programação. Para isso, foi elaborada uma pergunta relacionada ao desenvolvimento de um projeto em *Scratch*.

O gráfico da Figura 8 apresenta as respostas dos alunos à pergunta. É possível verificar que 75% dos alunos conseguiram compreender a pergunta e assinalar a resposta correta. Portanto, os conhecimentos adquiridos pelos alunos na combinação de blocos e na aplicação de conceitos matemáticos, foram comprovados através das respostas.

Figura 8 - Respostas dos alunos à questão sobre a criação de um projeto.



Fonte: Os autores.

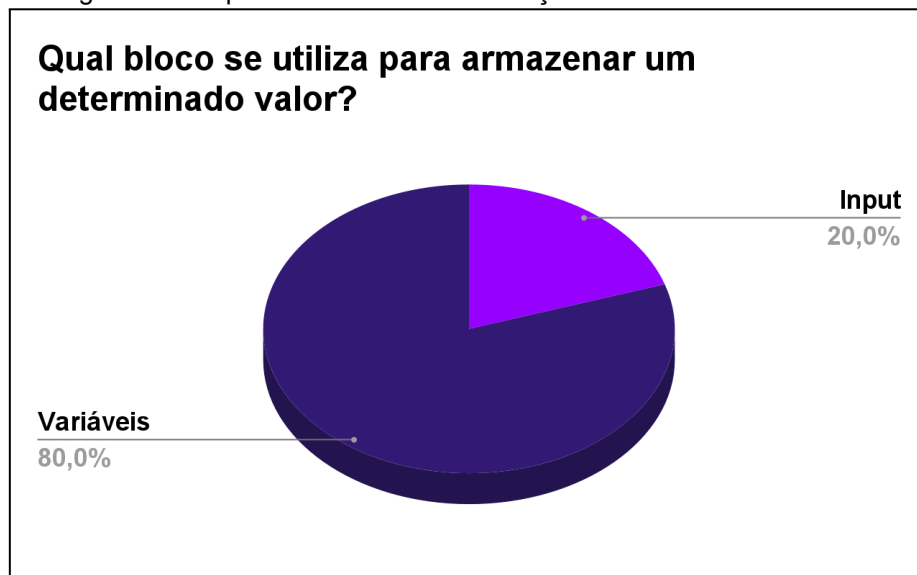
Segundo Módulo: Micro:Bit

Ao longo das aulas sobre a plataforma *MakeCode* e a placa *Micro:Bit*, os alunos tiveram a oportunidade de explorar diferentes sensores, como temperatura, luminosidade e acelerômetro e outros elementos de entrada de dados. Para facilitar a compreensão

desse tópico, decidiu-se ministrar as aulas como foco ao conceito de variáveis. Essa escolha se deu porque as variáveis desempenham um importante papel nas atividades com sensores, e também na criação de jogos.

Durante as aulas, os estudantes aprenderam como utilizar variáveis para armazenar e manipular dados. No gráfico da Figura 9 é possível visualizar as respostas dos alunos em relação ao conceito de variáveis que foi ensinado. Os resultados revelam que 80% dos alunos conseguiram compreender plenamente o conteúdo, demonstrando um alto nível de assimilação e aplicação prática dos conhecimentos adquiridos ao longo das aulas.

Figura 9 - Respostas dos alunos em relação ao conceito de variáveis.

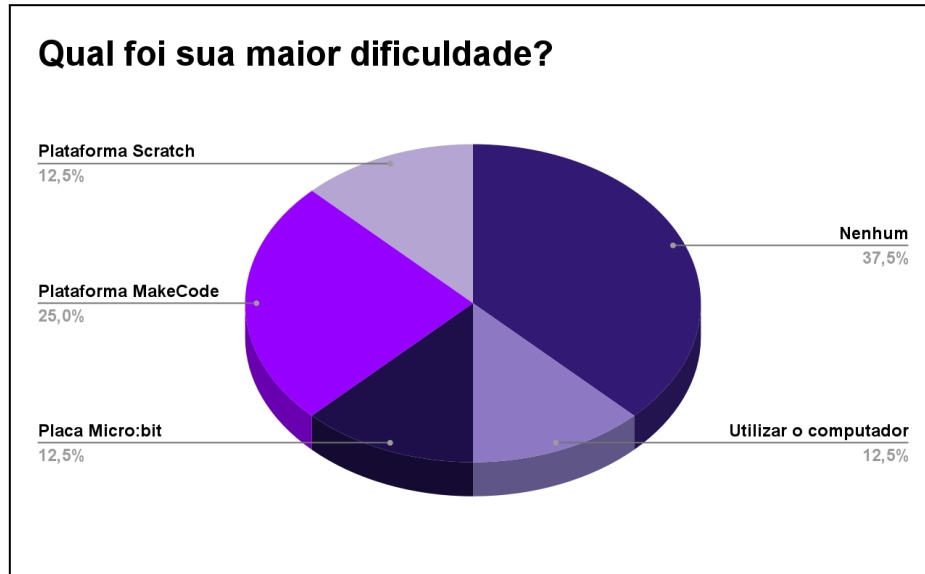


Fonte: Os autores.

Dificuldades apresentadas pelos alunos

Com o objetivo de analisar os desafios na aprendizagem dos alunos, foi realizado um levantamento a respeito das principais dificuldades que os alunos tiveram durante as aulas, conforme mostra o gráfico da Figura 10. Uma porção considerável dos alunos declaram não ter tido dificuldades durante o manuseio das plataformas (*Scratch* e *MakeCode*) e da placa (*Micro:Bit*). No entanto, conforme indicado no gráfico Figura 6, a pouca familiaridade com o uso de computadores emergiu como uma das dificuldades. Essa inexperiência com o uso de computadores influenciou significativamente a capacidade dos alunos de interagir com as plataformas, tanto de *Scratch* quanto a do *Micro:Bit*.

Figura 10 - Dificuldades apresentadas pelos alunos ao final do curso



Fonte: Os autores.

4. CONSIDERAÇÕES FINAIS

Mediante ao exposto ao longo deste trabalho, observou-se que os alunos que participaram do curso, foram estimulados ao desenvolvimento de lógica de programação, Pensamento Computacional e o manuseio de computadores e sensores. Além disso, ao aprender a programar de forma interativa e visual, os estudantes foram incentivados a explorar conceitos abstratos de forma prática.

Os autores deste trabalho consideram que o curso de programação com *Scratch*, *MakeCode* e *Micro:Bit*, não apenas complementa, mas potencializa os benefícios da metodologia PBL por projetos e da introdução de novas tecnologias no ambiente educacional.

Além disso, foi possível observar que a metodologia PBL por meio de projetos de extensão e ensino impulsionam os alunos a buscarem conhecimento de forma autônoma e contribui para a habilidade de identificar e propor soluções para problemas em diversas áreas. Esse processo é fundamental para o desenvolvimento do Pensamento Computacional, pois oferece uma nova perspectiva sobre o mundo e seus problemas.

Assim, os autores deste trabalho, ressaltam que através do Pensamento Computacional, as faculdades mentais são aprimoradas, especialmente no que diz respeito ao senso crítico e à resolução de problemas. Estimulando os alunos a desenvolverem competências essenciais para enfrentar os desafios atuais e futuros, tanto no âmbito acadêmico quanto na vida profissional.

Como facilitador desse desenvolvimento, tem-se a introdução de novas tecnologias no cotidiano de jovens e adolescentes, como *Scratch*, *MakeCode* e *Micro:Bit*. A exposição dessas novas tecnologias ajuda os estudantes a se adaptarem rapidamente às demandas tecnológicas do mercado, tornando-os candidatos mais competitivos e versáteis. Além disso, a proficiência em tecnologias digitais desperta o interesse dos alunos por áreas como engenharia e ciências exatas, incentivando-os a considerarem essas disciplinas em suas futuras escolhas acadêmicas e profissionais.

AGRADECIMENTOS

Os autores agradecem à Universidade Federal do Pará (UFPA), ao Programa de Educação Tutorial de Engenharia Elétrica (PET-EE/UFPA) e à Pró-Reitoria de Ensino de Graduação, pelo apoio concedido para a realização deste trabalho. Agradecemos também ao Centro de Integração Social Piedade por fornecer a estrutura necessária e o apoio aos ministrantes.

REFERÊNCIAS

- BRANDÃO, Pedro; PEDRO, Neuza. Micro: bit no suporte ao desenvolvimento do Pensamento Computacional: uma prática de ensino na disciplina de TIC. **Medi@ções**, v. 12, n. 1, p. 80-94, 2024.
- COSTA, Newerlyson Correia et al. O uso da plataforma Scratch como ferramenta facilitadora durante o ensino de lógica de programação para alunos do ensino médio: The use of the Scratch platform as a facilitating tool during the teaching of programming logic for high school students. **Brazilian Journal of Development**, p. 59279-59293, 2022.
- DE ANDRADE, Juliana Cristina dos Santos et al. Aprendizagem Baseada em Projetos aplicada ao ensino de Programação: revisão sistemática de literatura. **Terra e Didática**, v. 19, p. e023041-e023041, 2023.
- DE TONI, Thiago Antonio et al. O Teste de Turing. **ANAIIS DA MOSTRA DE INICIAÇÃO CIENTÍFICA DO CESUCA-ISSN 2317-5915**, n. 16, p. 44-45, 2022.
- FREITAS, Augusto et al. Sala de aula invertida: percepções docentes e discentes a partir de um relato de experiência das aulas de Tópicos em Química na Pós-graduação. **Revista Brasileira de Ensino de Ciências e Matemática**, v. 4, n. 1, 2021
- OLIVEIRA, Sebastião Luís de; SIQUEIRA, Adriano Francisco; ROMÃO, Estaner Claro. Aprendizagem Baseada em Projetos no Ensino Médio: estudo comparativo entre métodos de ensino. **Bolema: Boletim de Educação Matemática**, v. 34, p. 764-785, 2020.
- PEREIRA, Francisco Tito Silva Santos. Avaliação de uma Abordagem de Ensino-Aprendizagem de Programação Orientada a Objetos. **Anais dos Seminários de Iniciação Científica**, n. 24, 2020.
- RODRIGUES, Iris Caroline Santos; FONSECA, Wellington Silva; FARIAS, Aleqssandro Alexandre de Oliveira; SILVA, Kaylane de Souza. Aproximações entre a educação básica e as engenharias: introdução ao pensamento computacional no ensino fundamental. In: CONGRESSO BRASILEIRO DE EDUCAÇÃO EM ENGENHARIA - COBENGE, 2023, Rio de Janeiro. Anais do Congresso Brasileiro de Educação em Engenharia - COBENGE 2023, 2023.
- RODRIGUES, Iris Caroline Santos; FONSECA, Wellington Silva; FARIAS, Aleqssandro Alexandre de Oliveira. Iniciativas do Programa de Educação Tutorial (PET): enfrentando a evasão nos cursos de engenharia. In: CONGRESSO BRASILEIRO DE EDUCAÇÃO EM ENGENHARIA - COBENGE, 2023, Rio de Janeiro. Anais do Congresso Brasileiro de Educação em Engenharia - COBENGE 2023, 2023.
- SILVA, Denilson Rodrigues da. Desenvolvimento do pensamento computacional como dimensão estruturante da atividade do professor de cursos superiores de computação. 2021.
- SILVA, Joélio Marinho Batista. Máquina de turing emulando um codificador binário (nibble) com saída em 7-segmentos. **Trabalho de conclusão de curso**, 2023

SILVA, Saulo Kaue Santos; SILVA, André Oliveira Carvalho; CAMPOS, Matheus Leão; SHUTE, Valerie J.; SUN, Chen; ASBELL-CLARKE, Jodi. Demystifying computational thinking. **Educational research review**, v. 22, p. 142-158, 2017.

DE SOUSA, Janisley Oliveira et al. Uso da robótica na educação pública como ferramenta de ensino-aprendizagem: um relato de experiência Using robotics in public education as a teaching-learning tool: a report of experience. **Brazilian Journal of Development**, v. 8, n. 4, p. 24954-24963, 2022.

WING, Jeannette M. Computational thinking. *Communications of the ACM*, [S. l.], v. 49, n. 3, p. 33, 2006. DOI: 10.1145/1118178.1118215. Disponível em: <http://portal.acm.org/citation.cfm?doid=1118178.1118215>.

YBARRA, Luis Antonio Ccopa; SOARES, Marisa. A robótica e o pensamento computacional na educação: Uma proposta de avaliação da aprendizagem baseada em projetos. **Dialogia**, n. 40, p. e21524-e21524, 2022.

Abstract: *Computational Thinking, essential in the 21st century, enables students to develop problem-solving skills. These competencies can be enhanced through online platforms that allow planning, executing, and refining projects in an interactive and visual manner. Integrated with Project-Based Learning (PBL), these tools make learning even more effective. In this project, the PBL method was used to teach young people and adolescents programming logic and to develop computational thinking. The combination of Scratch and Micro:Bit/MakeCode was utilized in a PBL environment, promoting active, collaborative, and iterative learning. In this approach, students can experiment, receive immediate feedback, and continuously improve their projects. The application of this methodology proved to be effective, yielding positive results in teaching programming and developing computational thinking among students.*

Keywords: *Computational Thinking, Project-Based learning, Scratch, MakeCode, Algorithm, Micro:Bit.*

