

Curso Introdutório de Lógica de Programação: Uma Ação para Aproximar Jovens em Vulnerabilidade Social ao Aprendizado de Computação

Vítor de M. Mandowski, Willian do E. S. Rodrigues, Emerson de V. Vieira,
Ana Carolina Konradt, Santiago D. V. A. Martinez,
Tiago D. Mackedanz, Laura Q. Jurgina, e Leomar S. da Rosa Júnior

¹Universidade Federal de Pelotas (UFPEL)

{vamandowski, wdesrodrigues, edvvieira, ana.cnk, santiago.dvam,
tdmackedanz,lqjurgina, leomarjr}@inf.ufpel.edu.br

Abstract. *The development of computational thinking and logical reasoning is fundamental for problem-solving. Programming logic is a tool that allows one to exercise the ability to structure a task methodically, with the purpose of making the process more efficient and agile. In this context, this study aims to improve programming logic skills in high school students, evaluating the impact that stimulating logical reasoning can have on solving various tasks and challenges. Through logical tests, an improvement in students' performance results was observed after they completed the course. Questionnaires were also conducted regarding the quality of teaching in the course, which students positively evaluated, highlighting the effectiveness of the course.*

Resumo. *O desenvolvimento do pensamento computacional e raciocínio lógico é fundamental para a resolução de problemas. A lógica de programação é uma ferramenta que permite exercitar a capacidade de estruturar uma tarefa de forma metódica, com o propósito de tornar o processo mais eficiente e ágil. Nesse contexto, este trabalho tem como objetivo principal o aprimoramento das habilidades de lógica de programação em alunos do ensino médio, avaliando o impacto que o estímulo ao raciocínio lógico pode ter na resolução de diversas tarefas e desafios. Através de testes lógicos, observou-se uma melhoria nos resultados de desempenho dos alunos após terem feito o curso. Também foram feitos questionários sobre a qualidade de ensino no curso, os quais os alunos avaliaram positivamente, destacando a efetividade do curso.*

1. Introdução

A Tecnologia está cada vez mais presente na sociedade, tornando-se um assunto de interesse para todas as pessoas, principalmente para os jovens, pois o uso de dispositivos tecnológicos é irreversível no cotidiano [Sousa 2011]. Além disso, a tecnologia tornou-se uma ferramenta de grande importância para a educação, tanto que, no atual cenário, educação e tecnologias são fortemente ligadas, sendo que a maior parte das ferramentas tecnológicas são utilizadas como apoio nos processos educativos, mesmo que não tenham sido desenvolvidas para esse fim [Câmara et al. 2016]).

A tecnologia está intrinsecamente ligada ao conceito de pensamento computacional, definido pela aplicação de estratégias computacionais para solucionar desafios

[Wing 2006]. Esse tipo de raciocínio é estruturado sobre quatro fundamentos principais: a fragmentação do problema em partes menores, mais gerenciáveis; a identificação de padrões que ajudem na antecipação de soluções; a abstração focada nos aspectos fundamentais para a resolução do problema; e a criação de algoritmos, que são conjuntos ordenados e precisos de passos que delineiam a solução para o problema.

Até o ano de 2022, era notável a escassez de estímulo aos conteúdos relacionados ao Pensamento Computacional e à Programação nas escolas públicas de ensino básico, como destacado por [de Oliveira et al. 2022]. Essa oportunidade de aprendizado era, em sua maior parte, acessível apenas a estudantes de famílias com condições financeiras mais favoráveis, que podiam se beneficiar de cursos particulares ou instituições de ensino privadas para adquirir conhecimentos em lógica e fundamentos computacionais. Em resposta a essa disparidade a Base Nacional Comum Curricular (BNCC) adotou iniciativas de projeto para integrar o ensino de Computação na Educação Básica, com a efetivação dessas medidas ocorrendo em 2023, conforme mencionado por [Ribeiro et al. 2022].

O aspecto dos algoritmos, um dos pilares do pensamento computacional, sugere uma abordagem estruturada e direcionada para a organização e resolução de problemas. Sob essa perspectiva, o estudo da lógica de programação assume uma importância fundamental, não apenas por facilitar a prática de algoritmos, mas também por contribuir significativamente para o desenvolvimento do pensamento computacional. Esse processo reforça a habilidade de analisar e resolver desafios complexos de maneira lúcida e otimizada.

É fundamental cultivar tanto o pensamento computacional quanto a lógica de programação em alunos do ensino médio, considerando seu potencial de aplicabilidade em variadas esferas do saber [Bordin and Quepfert 2018]. Introduzir esses estudantes à lógica de programação, através de sua incorporação em linguagens de programação específicas, pode não só aguçar a curiosidade deles pelo campo tecnológico, mas também prepará-los para uma ampla gama de oportunidades futuras [Chaves et al. 2019]. Esta abordagem não apenas envolve apropriar-se de conhecimento, mas também equipa com habilidades essenciais para o atual contexto, marcado pela predominância tecnológica.

Com o objetivo de fomentar o pensamento computacional e as habilidades de programação entre os estudantes do ensino médio, o Grupo PET Computação da Universidade Federal de Pelotas (UFPeL) elaborou um curso básico de programação em C. Esse curso foi especialmente desenhado para os alunos do Desafio Pré-Universitário Popular da UFPeL, um curso voltado para o preparo de alunos em vulnerabilidade social para a prova do ENEM.

O restante deste artigo é estruturado da seguinte maneira: a seção 2 discute trabalhos relacionados na área. A seção 3 detalha como o curso foi planejado. A seção 4 analisa e discute os resultados obtidos com o curso. Por fim, a seção 5 apresenta as conclusões, resumindo as conclusões e sugerindo direções para pesquisas futuras.

2. Trabalhos Relacionados

O artigo desenvolvido pelo PET Ciência da Computação da Universidade do Estado do Rio Grande do Norte (UERN) [de Lucena et al. 2018] destaca a importância do papel dos grupos PET na melhoria do ensino superior no Brasil, que enfrenta uma série de desafios

que incluem altos índices de desistência, níveis de entrada com baixa qualificação e resultados insatisfatórios na graduação. O PET Ciência da Computação tem se destacado por suas iniciativas inovadoras para lidar com esses problemas. Além de contribuir para o desenvolvimento acadêmico e pessoal dos participantes, o PET também desempenha um papel fundamental na divulgação e promoção do curso dentro da comunidade, demonstrando sua importância como agente de mudança no panorama educacional do país.

O trabalho proposto por [Chaves et al. 2019] constitui em uma iniciativa realizada pelo PET-TI da Universidade Federal do Ceará, que teve como objetivo principal despertar o interesse de estudantes do ensino médio e/ou ensino tecnológico, que tinham pouca ou nenhuma experiência com computação, pela área de Tecnologia da Informação (TI) e, ao mesmo tempo, promover o que é a computação e a sua importância. Isso foi realizado por meio de atividades introdutórias que abordaram programação e o desenvolvimento do pensamento lógico-matemático.

Os autores [Meneses et al. 2015] promovem anualmente um curso introdutório à programação para estudantes do ensino médio, conhecido como Projeto IntroComp, realizado pelo PET Engenharia de Computação da Universidade Federal do Espírito Santo. No relato de experiência, eles descrevem as iniciativas e estratégias do projeto, assim como alguns resultados obtidos em seus cinco anos de existência. O objetivo geral deste projeto é atrair alunos para uma instigante experiência com a programação, visando, sobretudo, despertar nos alunos o interesse pela computação por meio do desenvolvimento computacional.

Com o objetivo de combater a evasão do curso de Ciência da Computação da Universidade de Brasília (UnB) foi desenvolvido um Programa de Monitoria da Disciplina de Programação Introdutória [Holanda et al. 2024]. Através de um relatório, descobriu-se que a primeira disciplina de programação é uma grande responsável pela evasão do curso de Ciência da Computação. Através de aulas complementares, monitorias e revisões, durante os 4 semestres de aplicação do curso, o programa atingiu 265 alunos dos quais 92% consideraram a monitoria importante ou muito importante.

Com base nos trabalhos relacionados, é possível observar que os grupos PET tem atuado no combate a evasão e elaborando projetos de ensino em programação. Um ponto em comum entre a maioria dos trabalhos relacionados é a dificuldade encontrada pelos alunos ingressantes nas disciplinas de programação. Os projetos de apoio em programação tem trazido resultados positivos de forma geral. Neste contexto, evidencia-se a necessidade de introduzir programação para alunos que ainda não estão no ensino superior, de forma a melhor prepará-los para as disciplinas de programação, desta forma reduzindo a evasão do curso.

3. Metodologia

O planejamento do curso denominado “Compile.exe” foi estruturado com um total de 12 aulas, realizadas em encontros semanais, totalizando um período com duração média de 90 minutos cada. As aulas foram divididas em cerca de 30 minutos dedicados à apresentação do conteúdo teórico, seguidos por aproximadamente 60 minutos destinados às atividades práticas.

O Grupo PET planejou todas as aulas, as quais foram postas à disposição no Portal do PET para acesso dos alunos. Após cada conteúdo ministrado em aula, foram disponibi-

lizadas vídeo-aulas no canal do YouTube do PET e no Portal do PET para complementar o aprendizado. Durante as aulas, os alunos usaram a IDE de programação Geany para escrever e compilar códigos, facilitando a prática.

Inicialmente, as aulas foram organizadas e divididas com o seguinte planejamento de conteúdo: a primeira, dedicada à introdução, abordando a importância da lógica de programação no cotidiano, além de alguns conceitos de hardware e software. Nas demais aulas, foram apresentados os seguintes conteúdos: uma aula sobre lógica, outra sobre variáveis, duas aulas sobre estruturas condicionais, duas aulas sobre estruturas de repetição, uma aula sobre vetores e strings, uma aula sobre matrizes, uma aula sobre switch case, uma aula com exercícios práticos, e a última aula consistiu em uma revisão de todo o conteúdo. A Tabela 1 apresenta a relação de conteúdos.

Tabela 1. Plano de conteúdo das aulas

Aula	Conteúdo
1	Introdução
2	Lógica
3	Variáveis
4	Estruturas condicionais
5	Estruturas condicionais
6	Estruturas de repetição
7	Estruturas de repetição
8	Vetores e <i>strings</i>
9	Matrizes
10	<i>Switch case</i>
11	Exercícios práticos
12	Revisão do conteúdo

O planejamento do curso Compile.exe aplica os princípios fundamentais dos pilares do pensamento computacional de uma maneira integrada e progressiva. Na introdução, a abordagem à importância da lógica de programação no cotidiano conecta-se diretamente com a decomposição, incentivando os alunos a quebrar problemas complexos em partes menores e mais gerenciáveis. Esta etapa é importante para o desenvolvimento inicial do pensamento computacional, preparando o terreno para compreensões mais aprofundadas.

À medida que o curso avança, as aulas sobre lógica, variáveis, e estruturas condicionais e de repetição promovem o reconhecimento de padrões. Os alunos são encorajados a identificar estruturas lógicas e padrões de dados recorrentes, aplicando-os de forma criativa a novos problemas. Esta prática não apenas reforça a habilidade de identificar semelhanças entre diferentes desafios de programação mas também estimula uma abordagem analítica para a solução de problemas.

A abstração é enfatizada particularmente nas aulas sobre vetores, strings, matrizes e switch case. Os alunos aprendem a generalizar e extrair a essência de um problema, concentrando-se nos dados e nas operações relevantes enquanto ignoram detalhes irrelevantes. Esta habilidade é essencial para o desenvolvimento de soluções eficazes e eficientes, permitindo aos alunos aplicar os conceitos aprendidos a uma ampla gama de situações práticas.

Por fim, a formulação de algoritmos é uma competência constantemente desenvolvida ao longo do curso, culminando nas aulas de exercícios práticos e revisão. Nestas sessões, os estudantes aplicam o conhecimento adquirido para estruturar problemas de forma que possam ser resolvidos algorítmicamente. A prática de escrever, testar e refinar códigos promove um entendimento profundo dos princípios algorítmicos, consolidando o aprendizado e preparando os alunos para desafios futuros na programação.

No planejamento do curso, estabeleceu-se um sistema de revezamento entre os membros do Grupo PET para a mediação das aulas, de maneira que um petiano responsabiliza-se por ministrar o conteúdo de cada encontro. Os demais desempenham o papel de monitores, oferecendo apoio aos alunos durante as atividades práticas e proporcionando suporte individualizado quando necessário, tornando assim as aulas mais dinâmicas.

Na primeira aula do curso, aplicou-se um questionário composto por três questões que envolviam testes de habilidades, lógica, pensamento computacional e matemática básica, com o propósito de avaliar os alunos antes do início do curso. Essas três questões foram elaboradas por [Claudete Schorr and Bercht 2020]. O mesmo questionário foi aplicado ao final do curso para avaliar o desenvolvimento do pensamento computacional e das habilidades lógicas desenvolvidas e estimuladas ao longo do curso ministrado.

Ao final de cada encontro, foi aplicado um outro questionário que incluía perguntas de múltipla escolha relacionadas ao conteúdo abordado. Isso proporcionou a capacidade de mensurar o conhecimento adquirido pelos alunos ao término de cada aula. Além disso, esse questionário continha perguntas destinadas a avaliar a qualidade do ensino ministrado na aula em questão. Estas perguntas foram desenvolvidas com base nos critérios estabelecidos por [Jurgina et al. 2020] e estão apresentadas abaixo:

1. O conteúdo desta aula foi de fácil entendimento?
2. Com o que aprendeu na aula, conseguiu realizar os exercícios propostos?
3. Gostou do conteúdo dessa aula?
4. Você achou que a aula foi bem aplicada?
5. Você acha que o que aprendeu hoje terá algum uso no seu cotidiano?

4. Resultados e Discussões

Com o objetivo de avaliar evolução dos alunos ao longo do curso, foi comparado os resultados das aplicações dos 2 testes lógicos. Em relação ao questionário de lógica aplicado na primeira aula, 21 alunos responderam, dos quais 15 concluíram o curso. As respostas foram avaliadas usando uma escala de notas de 0 a 4. Esses resultados iniciais serviram como base para avaliar o desenvolvimento do pensamento computacional e das habilidades lógicas e de matemática básica dos alunos ao longo do curso. No último encontro do curso, o mesmo questionário foi aplicado à turma, com 14 dos alunos presentes na primeira aula respondendo, além de 2 alunos que não estiveram na primeira aula, mas concluíram o curso.

O gráfico apresentado na Figura 1 exibe os resultados a partir da aplicação dos 2 questionários lógicos. É possível observar uma melhoria significativa nas notas dos alunos. Além disso, dado que as questões são dissertativas, observou-se um avanço notável na capacidade dos estudantes em elaborar e construir suas respostas de maneira mais eficiente. A questão 3 foi a que obteve o ganho mais significativo, seguida da questão

1. No entanto, na questão 2, os alunos mantiveram a mesma média nos dois questionários, acredita-se que isso ocorreu devido a uma maior complexidade da resolução dessa questão.

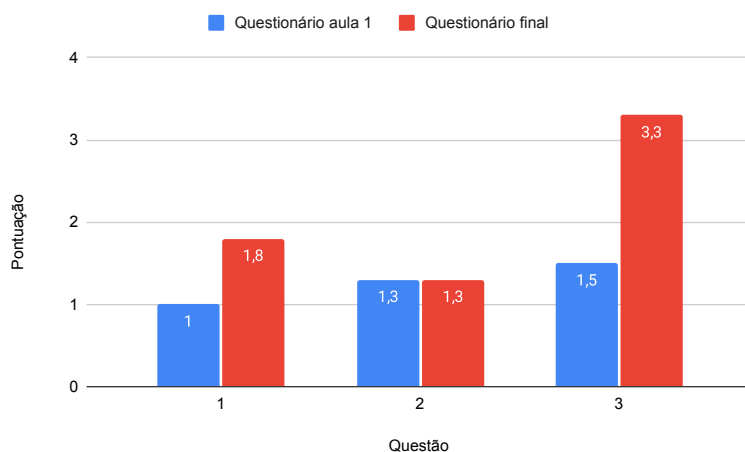


Figura 1. Resultados dos questionários lógicos

Na Figura 2, é possível encontrar o resultado das perguntas relacionadas à qualidade de ensino, acompanhado por um gráfico que exibe a média de todas as respostas das 11 primeiras aulas. Ao analisar este gráfico, conclui-se que a metodologia adotada mostrou-se eficiente em relação à satisfação dos alunos durante o curso, embora alguns alunos acreditem que os conteúdos ministrados podem ter aplicação limitada uso cotidiano.

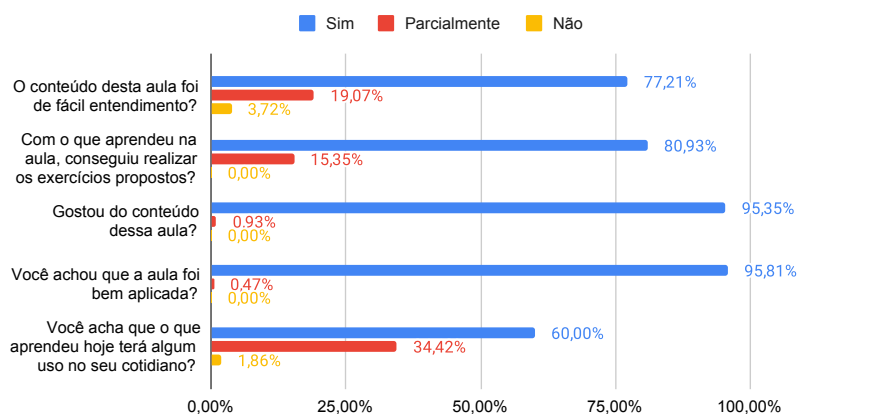


Figura 2. Média das Perguntas sobre Qualidade do Ensino

Em relação aos questionário semanais, os resultados foram divididos por conteúdos, para que se pudesse analisar onde houve mais dificuldade. Estes questionários foram de múltipla escolha, diferente do questionário lógico. o gráfico da Figura 3 mostra os resultados para as perguntas sobre *hardware*, *software* e introdução a lógica de

programação. Neste questionário, 75% das questões tiveram um resultado satisfatório. A questão que obteve o maior índice de erros teve seu conteúdo reforçado durante as aulas subsequentes.

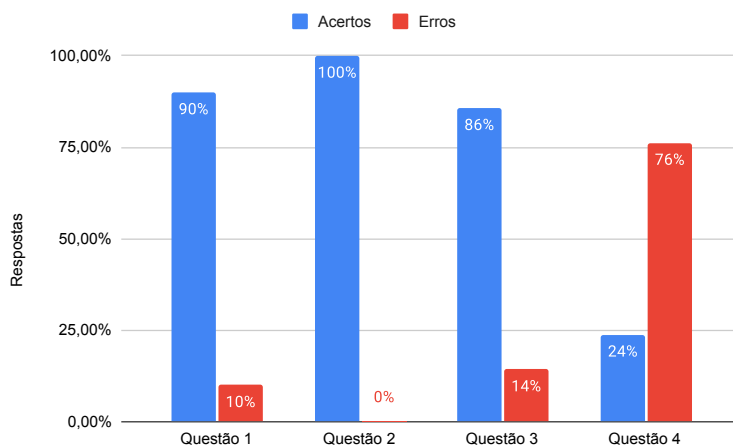


Figura 3. Respostas do questionário sobre *hardware*, *software* e introdução a lógica de programação

O gráfico da Figura 4 contém os resultados das questões relacionadas a variáveis, além de entrada e saída de dados em C, onde 5 das 6 perguntas tiveram desempenho satisfatório, demonstrando um sólido entendimento por parte dos alunos desses conceitos fundamentais.

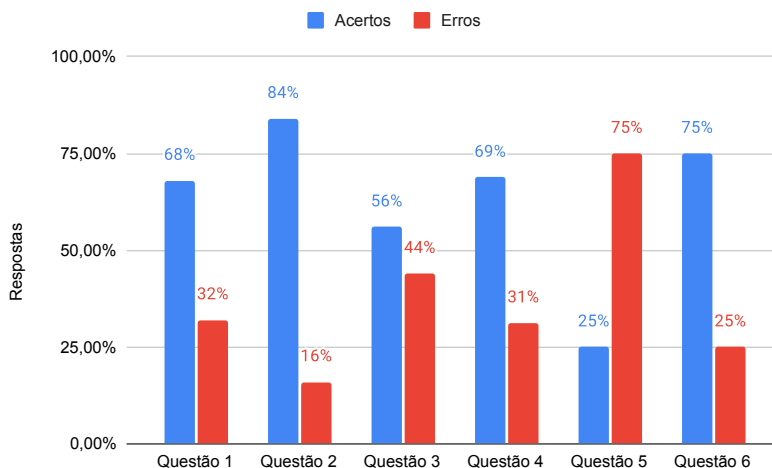


Figura 4. Respostas do questionário sobre variáveis, entrada e saída de dados em C

O terceiro conteúdo abordou o uso de operadores lógicos e condicionais, os resultados estão presentes no gráfico da Figura 5. A partir dessa aula, notou-se a necessidade de dedicar mais tempo à parte prática do curso, visto que o índice de erros aumentou consideravelmente. Foi neste ponto que começamos a realizar alterações no cronograma inicialmente proposto.

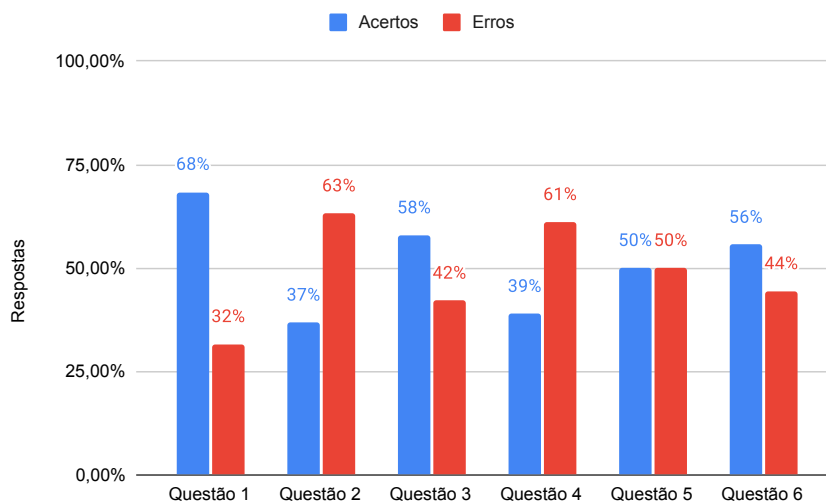


Figura 5. Respostas do questionário sobre operadores lógicos e condicionais

O gráfico da Figura 6 apresenta os resultados do conteúdo que demandou um maior número de aulas para abordagem: laços de repetição, que ocuparam três aulas. Acredita-se que esse conteúdo tenha sido o mais complexo trabalhado ao longo do curso, o que resultou no maior índice de erros nos questionários.

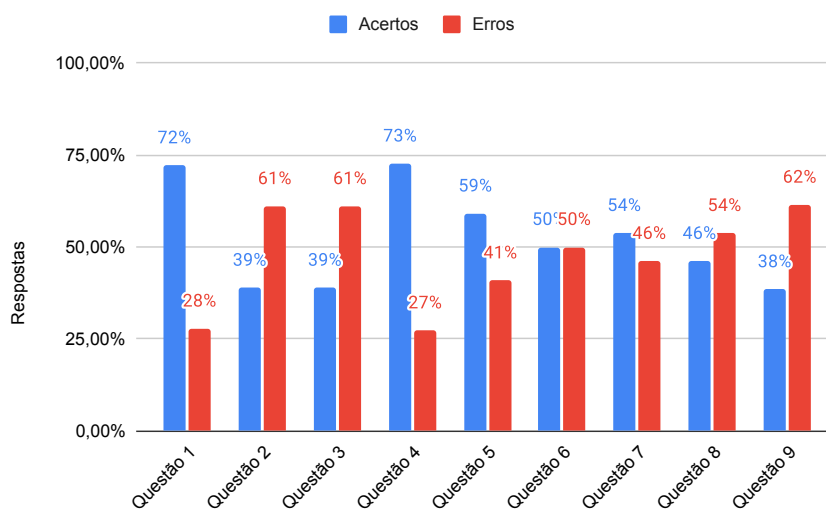


Figura 6. Respostas do questionário sobre laços de repetição

Por último, o gráfico da Figura 7 apresentou uma taxa de erro considerável em 5 das 6 questões sobre vetores e *strings*. Esse alto índice de erro indica que os alunos enfrentaram mais desafios consideráveis ao lidar com esse conteúdo específico, assim como no conteúdo anterior.

Com base nos resultados obtidos com os formulários sobre a qualidade de ensino, pôde-se observar que no geral a metodologia aplicada foi satisfatória, de acordo com as avaliações realizadas pelos alunos. Sobre os exercícios e dificuldade do entendimento

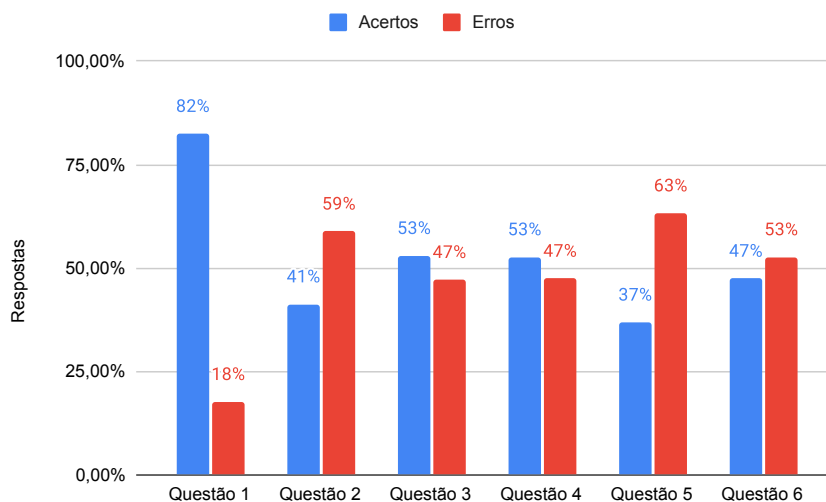


Figura 7. Respostas do questionário sobre vetores e strings

do conteúdo, apesar de satisfatórios os resultados, uma quantidade significativa da turma marcou “parcialmente” o que acredita-se que se deve a falta de prática dos exercícios fora do período de aula. O dado mais preocupante foi em relação ao uso no cotidiano do conteúdo, um fator impactante nisso é que por serem alunos de ensino médio, não necessariamente continuarão na área de tecnologia.

Em relação ao teste lógico aplicado no primeiro e último encontro, houve uma melhora significativa de desempenho, demonstrando que a prática de programação e o estímulo ao raciocínio lógico contribuem para a resolução de exercícios de lógica no geral. Como se tratavam de perguntas dissertativas, observou-se também que as respostas do teste final foram no geral mais completas e estruturadas, quando comparadas ao primeiro teste.

Ao longo do curso, foi observado um aumento na taxa de erro dos questionários sobre os conteúdos, e essa tendência pode ser atribuída à acumulação de conteúdo. Isso se deve ao fato de que a compreensão adequada dos tópicos mais recentes depende do entendimento dos conceitos anteriores. Portanto, devido ao curto período do curso e à falta de prática assíncrona nos exercícios de programação, o conteúdo pode ter se tornado um tanto sobrecarregado, o que levou a uma queda na taxa de acertos. No entanto, é importante destacar que a turma apresentou um desempenho satisfatório e conseguiu acompanhar as aulas de forma geral, o que é um resultado positivo.

5. Conclusões

Este trabalho apresentou os resultados da aplicação do curso Compile.exe. O curso proporcionou uma valiosa oportunidade para o Grupo PET Computação se engajar em um projeto de extensão. Os projetos de extensão têm como objetivo principal estabelecer uma ponte entre a universidade e a comunidade, contribuindo para o fortalecimento do sistema educacional.

Ao longo do curso, avaliou-se o entendimento dos alunos em cada aula e as dificuldades observadas pelos monitores na turma. Com base nisso, ajustamos o cronograma

proposto para atender às necessidades da turma, com o objetivo de tornar as aulas mais dinâmicas e facilitar o entendimento dos alunos. Isso envolveu a otimização do conteúdo, que incluiu a remoção dos tópicos sobre matrizes e *switch case* para manter o curso com 12 aulas, além de reforçar o aprendizado desses conteúdos, contribuindo para a dinâmica das aulas e auxiliando os alunos na compreensão do material.

O curso teve uma taxa de desistência de 31% dos alunos. Isso reflete o engajamento e o interesse da maioria dos participantes em aprender programação. Acredita-se que a taxa de desistência se deve ao fato do curso ter sido ofertado no turno da noite durante o rigoroso inverno do sul do país, algo que os alunos relataram que dificultou a participação de algumas das aulas.

A aplicação do curso se mostrou positiva, tanto em relação a melhora dos resultados dos testes lógicos após o curso, quanto a boa avaliação dos alunos referente a qualidade de ensinamentos das aulas. Esses resultados demonstram a importância de fomentar projetos da área de ensino de lógica e programação.

Para futuras edições do curso, pretende-se aumentar o incentivo da prática assíncrona dos conteúdos, além de reforçar a importância da lógica no cotidiano, pois mesmo não seguindo a área de tecnologia, a lógica pode ser utilizada para diversas finalidades. É de suma importância que acadêmicos da área tecnológica disseminem o conhecimento de lógica de programação, pois exercitar o raciocínio lógico e o pensamento computacional é importante para qualquer área de conhecimento.

Referências

- Bordin, A. and Quepfert, W. (2018). Projeto de ensino de programação para alunos de ensino médio: Uma análise do cenário e das percepções das oportunidades. In *Anais do XXIV Workshop de Informática na Escola*, pages 205–214, Porto Alegre, RS, Brasil. SBC.
- Câmara, F. S. D. S. et al. (2016). A lógica de programação como ferramenta para o ensino e a aprendizagem da matemática. In Editora, R., editor, *Anais II CINTEDI*, Campina Grande. Realize Editora. Acesso em: 22/05/2024 14:01.
- Chaves, L. S., Bezerra, C. I. M., et al. (2019). Ensino de programação em escolas públicas: Relato de uma ação do pet-ti. In *Anais do XXV Workshop de Informática na Escola*, pages 667–676. SBC.
- Claudete Schorr, M. and Bercht, M. (2020). Pcomp-model: desenvolvendo competências para o pensamento computacional. *Revista Novas Tecnologias na Educação*, 18(1).
- de Lucena, L. B., de Andrade, G. L. C., de Fátima Andrade Soares, E., dos Santos Júnior, W., Álvaro Gabriel de Oliveira, de Souza Dantas Batista, L. M., Gomes, D. A., and de Lima, R. W. (2018). As ações do pet no desenvolvimento do curso de ciência da computação. In *Anais do XXVI Workshop sobre Educação em Computação*, Porto Alegre, RS, Brasil. SBC.
- de Oliveira, M. G., da Silva, M. F., and Rodrigues, C. B. (2022). Curso híbrido baseado em moocs de lovelace e oficinas presenciais para aprendizagem ativa e nobre de pensamento computacional e programação. In *Anais do XXVIII Workshop de Informática na Escola*, pages 179–188. SBC.

- Holanda, M., Bandeira, I. N., Castanho, C. D., Sousa, A. B. d., and Silva, D. d. (2024). Programa de monitoria da disciplina de programação introdutória na universidade de Brasília. *Revista Brasileira de Informática na Educação*, 32:157–180.
- Jurgina, L. Q., Zanandrea, R., Júnior, L. S. R., and de Souza Marques, F. (2020). Logicflow: Uma ferramenta para o auxílio de ensino-aprendizagem de circuitos digitais. In *Anais do XXXI Simpósio Brasileiro de Informática na Educação*, pages 322–331. SBC.
- Meneses, L., Mai, L., Rosario, J., de Oliveira, E., and Gomes, R. (2015). Introcomp: Atraindo alunos do ensino médio para uma instigante experiência com a programação. In *Anais do XXIII Workshop sobre Educação em Computação*, pages 366–375, Porto Alegre, RS, Brasil. SBC.
- Ribeiro, L., da Costa Cavalheiro, S. A., Foss, L., da Cruz, M. E. J. K., and de França, R. S. (2022). Proposta para implantação do ensino de computação na educação básica no Brasil. In *Anais do XXXIII Simpósio Brasileiro de Informática na Educação*, pages 278–288. SBC.
- Sousa, C. (2011). Novas linguagens e sociabilidades: como uma juventude vê novas tecnologias. *Revista interações*, pages 170–188.
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3):33–35.