

PROGRAMAÇÃO COMO UMA FERRAMENTA NO ENSINO DE MATRIZES

Pedro Araújo Kalile¹

GD6 – Educação Matemática, Tecnologias e Educação a Distância

Resumo: O presente trabalho é uma pesquisa em desenvolvimento do Programa de Mestrado Profissional em Matemática – PROFMAT e que tem como objetivo analisar a influência das linguagens de programação no ensino de matemática. Para tanto, será traçado um paralelo entre as competências e habilidades necessárias para utilização dos recursos tecnológicos atuais e futuros e as competências e habilidades que devem ser desenvolvidas durante o ensino de matemática no ensino básico. Para o levantamento das competências e habilidades, foram utilizados o pensamento computacional e a BNCC. Por fim, serão desenvolvidas propostas de atividades para ensinar um conteúdo matemático do ensino médio, álgebra matricial, através da programação e serão coletados dados para posterior análise.

Palavras-chave: Pensamento Computacional. Programação. Matriz.

CONSIDERAÇÕES INICIAIS

Com a descoberta da contagem, o homem começou a representar os números de seu cotidiano através de riscos em ossos ou em pinturas rupestres. Com o passar do tempo, contar não era mais suficiente, então surgiram as operações matemáticas e a dificuldade de realizar essa computação mais primitiva. Com o novo desafio, a representação numérica não era mais suficiente, então o homem passou a buscar maneiras de representar os cálculos e novas formas de representar os números, no intuito de facilitar na árdua tarefa de resolver as operações matemáticas.

Dessa forma, pedras, conchas, buracos no chão, dedos das mãos e muitos outros objetos foram utilizados na tentativa de facilitar o desenvolvimento das operações. Muitos séculos depois, como fruto do acúmulo de experiências dessas tentativas, nasce o ábaco, considerada a primeira calculadora primitiva. Por muito tempo, o ábaco foi o principal instrumento para auxiliar em cálculos. Posteriormente, segundo Eves (2004), em 1642, Blaise Pascal (1623-1662) construiu um instrumento para auxiliar seu pai em cálculos mais complexos, pois ele era coletor de impostos. Na sequência, o avanço da tecnologia permitiu o surgimento de diversos instrumentos que realizam cálculos como o ASCC

¹ Centro Federal de Educação Tecnológica de Minas Gerais – CEFET-MG; PROFMAT; Mestrado Profissional em Ensino de Matemática; pedrokal@gmail.com; orientador: Carlos Magno Martins Cosme.

(*Automatic Sequence Controlled Calculator*), em 1944, nos Estados Unidos, o *ENIAC* (*Electronic Numerical Integrator and Computer*), em 1945, também nos Estados Unidos e o *ACE* (*Automatic Computing Engine*), em 1946, na Inglaterra. No início, esses instrumentos eram grandes e pesados (o *ASCC* pesava cerca de cinco toneladas, o *ENIAC* cerca de trinta toneladas e o *ACE* cerca de 10 toneladas), mas, com o tempo, eles foram se tornando cada vez menores e mais acessíveis à população, além de realizarem operações e cálculos mais complexos e muito mais rapidamente. Nesse novo cenário, “a computação conseguiu conquistar muitos dos aspectos de nossa sociedade e, para se encaixar, as pessoas precisam ser versáteis e adaptáveis à tecnologia moderna e futura” (SAELI *et al.*, 2011, p. 73)

Naturalmente, “o impacto da tecnologia na vida de cada indivíduo vai exigir competências que estão além do simples lidar com as máquinas” (PCN, 1999, p. 41). Logo, é papel da escola introduzir em seu currículo atividades e conteúdos para que os alunos do ensino médio “possam ser estimulados a desenvolver o *pensamento computacional*, por meio da interpretação e da elaboração de algoritmos, incluindo aqueles que podem ser representados por fluxogramas” (BNCC, 2017, p. 528, *itálico do autor*).

Para alguns autores o pensamento computacional está intimamente ligado ao ato de programar, sendo que

Programming involves the ability to generate a solution to a problem. Generating solutions means that one of the learning outcomes is the ability to solve problems and also, if the problem is a big problem, the ability to split the problem into sub problems and create a generalizable central solution. In addition, the student achieves the ability to create usable, readable and attractive solutions. (SAELI *et al.*, 2011, p. 77-78)²

De acordo com Wing (2006), o pensamento computacional vai além, pois é o ato de conceituar e não somente programar, sendo uma habilidade fundamental e complementar à Matemática e Engenharias, por fim é a forma de o homem pensar. Nessa perspectiva, não é suficiente ao indivíduo desenvolver o *letramento computacional* que é “a habilidade de (apenas) operar adequadamente os computadores” (BARCELOS; SILVEIRA, 2012, p. 5). Os mesmos autores traçam um paralelo entre o ensino de Matemática apresentado nos PCN e o pensamento computacional, trazendo algumas relações entre ambos. Por outro lado, a BNCC já trata do pensamento computacional ligado à Matemática.

² A programação envolve a capacidade de gerar uma solução para um problema. Gerar soluções significa que um dos resultados da aprendizagem é a capacidade de resolver problemas e também, se o problema for grande, a capacidade de dividi-lo em partes e criar uma solução central generalizável. Além disso, o estudante desenvolve a habilidade de criar soluções atraentes, úteis e claras.

É notável, então, a importância do pensamento computacional ligado ao ensino de Matemática e, nesse cenário, o presente trabalho possui o objetivo de apresentar uma proposta de atividade relacionando o ensino de Matemática com o uso de softwares matemáticos e linguagens de programação.

PROPOSTA DE ARTICULAÇÃO ENTRE O ENSINO DE MATEMÁTICA E O PENSAMENTO COMPUTACIONAL

Inicialmente, foi feito um levantamento bibliográfico sobre a utilização da programação como ferramenta para auxiliar no ensino de matemática. Esse levantamento bibliográfico não se restringiu ao Brasil e, dessa forma, foi possível constatar que esse é um tema que vem sendo estudado desde meados do século passado e é objeto de estudo de comunidades de professores de matemática e pesquisadores na área de ciências da computação até os dias atuais.

Em seguida, serão desenvolvidas atividades sobre resolução de sistemas lineares com o auxílio de matrizes a serem aplicadas a alunos do 2º ano do Ensino Médio em pelo menos duas modalidades de ensino: médio técnico e médio regular. Durante a realização das atividades, serão feitos os ajustes que sejam considerados importantes e a coleta de dados para posterior análise.

Por fim, será realizada a análise de dados coletados. Para tanto, será introduzido o conceito de transferência de competências entre domínios diferentes e será feita a discussão das competências e habilidades inerentes ao pensamento computacional que auxiliaram no desenvolvimento do pensamento lógico matemático.

Sobre a linguagem de programação escolhida

O primeiro tópico a ser apresentado aos alunos será a forma como a máquina “pensa” e quais as consequências dessa diferença. Será feita, então, uma rápida abordagem da aproximação numérica feita pela máquina, os erros gerados e seus cálculos. Em seguida, será rerepresentada a noção de algoritmo, pois os alunos já conhecem algoritmos matemáticos, como as quatro operações básicas e os métodos de resolução de equações polinomiais do 1º e 2º graus. A convergência de algoritmos não será apresentada, pois

exige um aprofundamento matemático que não é necessário no momento. Por fim, serão apresentadas as linguagens de programação e os recursos computacionais que podem ser trabalhados pelos alunos.

Podemos definir a linguagem de programação como o “idioma” da máquina e, da mesma forma que existem vários idiomas, existem várias linguagens de programação, cada uma com suas particularidades. Utilizar uma linguagem de programação mais simples e visual nos anos escolares iniciais auxilia no desenvolvimento do raciocínio lógico sem uma preocupação com a sintaxe, ou seja, com a forma de escrita do algoritmo.

Com o desenvolvimento da abstração, linguagens de programação mais robustas e menos visuais podem ser adotadas pelos alunos, de forma que a linguagem de programação não seja um fator muito importante para os alunos do Ensino Médio. Segundo Szlávi e Zsakó (2006), é muito importante ensinar as estruturas de programação tradicionais independentes de linguagem de programação. A escolha da linguagem de programação passa a ser do aluno que deve realizar essa escolha de acordo com suas preferências.

Por outro lado, a maioria dos alunos brasileiros não possui contato com linguagens de programação nos anos iniciais de sua escolarização, então será utilizado o programa Maxima para a realização dos cálculos iniciais. Além de ser um software livre GPL, o Maxima é um sistema de manipulação de expressões numéricas e algébricas que possui uma interface visual mais simples.

Concomitante ao trabalho com o Maxima, será feita uma oficina para ensinar os elementos básicos da linguagem de programação C. A escolha dessa linguagem de programação se deve ao fato de ser uma linguagem mais didática que as linguagens de máquina e a facilidade do aluno migrar futuramente para uma linguagem de sua preferência.

Conteúdo matemático escolhido

Segundo Parlett (2003), um dos três principais problemas da computação matricial é a resolução de sistemas lineares. De fato, muitos problemas da matemática aplicada podem ser transformados em sistemas lineares, porém com dimensões muito altas, dificultando a sua resolução. Ainda segundo Parlett (2003), durante o projeto de desenvolvimento do ACE (meados dos anos 1940), os matemáticos Leslie Fox (1918-

1992), Turing (1912-1954) e Wilkinson (1919-1986) demoraram duas semanas para resolver um sistema linear 18×18 com calculadoras de mesa. As transformações geométricas com ênfase na rotação são outro grande exemplo de aplicação da álgebra matricial, principalmente na movimentação de braços robóticos. Novamente, a quantidade de cálculos envolvidos para o estudo de um braço robótico inviabiliza a utilização desse exemplo para o ensino médio.

Por isso, o conteúdo matemático escolhido para realizar essa articulação é a álgebra matricial computacional. Dessa forma, é possível contextualizar um assunto tão importante do ensino médio e apresentar pelo menos duas possibilidades futuras para os alunos: programação e robótica.

O estudo de matrizes será introduzido pela matriz de rotação no espaço. Uma vez definido esse novo objeto matemático, será feita toda a sua construção conceitual e a construção das suas principais operações, com ênfase na multiplicação de matriz por vetor e matriz por matriz. Naturalmente, as operações com matrizes são algoritmos muito bem definidos e a primeira tarefa será desenvolver esses cálculos utilizando o Maxima ou desenvolvendo um programa em C ou outra linguagem.

Os alunos farão uma atividade empírica, em que realizarão a multiplicação de uma matriz quadrada por ela mesma mil vezes. Em um primeiro momento, esse número parece ser muito grande, mas para se garantir a movimentação de forma fluida de um braço robótico são necessárias muitas multiplicações matriciais. Os alunos, então, enfrentarão um dos grandes problemas da programação: a velocidade de processamento. Serão propostas algumas alterações nos códigos dos alunos para uma melhor otimização e novos testes serão feitos. A biblioteca da *Basic Linear Algebra Subprograms (BLAS)* será muito utilizada como fonte de pesquisa para os alunos nesse processo de melhoramento de seus códigos.

Em seguida, será apresentado o processo de resolução de sistemas lineares através do escalonamento de matrizes. Esse processo possui um alto custo computacional, pois a mudança de linhas não é uma tarefa fácil para o computador. Como solução alternativa, serão apresentados o algoritmo de Cholesky³ e um algoritmo iterativo de resolução de sistemas lineares. Serão apresentadas duas alternativas para que os alunos entendam as

³ O algoritmo de Cholesky realiza a decomposição de uma matriz quadrada em duas matrizes triangulares não singulares, uma inferior e outra superior.

características de cada uma e possam realizar a escolha de qual algoritmo utilizar. Essa tomada de decisão será determinada pelo problema que o aluno deseja solucionar ou modelar, uma vez que as características iniciais podem ser mais favoráveis para um algoritmo em detrimento do outro e vice-versa.

Essa construção matemática será pautada em três grandes referências: Burden e Faires (2011); Golub e Loan (2013); e Stewart (1973). Vale ressaltar que os alunos não terão acesso a essa bibliografia de forma direta, pois possuem um alto nível de abstração e profundidade. Dessa forma, serão construídos estudos dirigidos e apostilas do conteúdo.

Além disso, para o melhor desenvolvimento dos algoritmos iniciais, utilizaremos o método de Resolução de Problemas que “consiste essencialmente em um *procedimento*, um curso de ação, um esquema de operações bem inter-relacionadas, um *modus operandi*” (POLYA 1981, p.122) que é muito próximo do pensamento computacional. Ainda segundo Polya (1981), o estudo de matemática deve propiciar, sempre que possível, um ambiente de trabalho criativo e independente.

Já para o estudo dos sistemas lineares, será utilizada a modelagem matemática, afinal

modelo matemático consiste em se ter uma linguagem concisa que expressa nossas ideias de maneira clara e sem ambiguidades, além de proporcionar um arsenal enorme de resultados (teoremas) que propiciam o uso de métodos computacionais para calcular suas soluções numéricas. (BASSANEZI, 2002, p. 20).

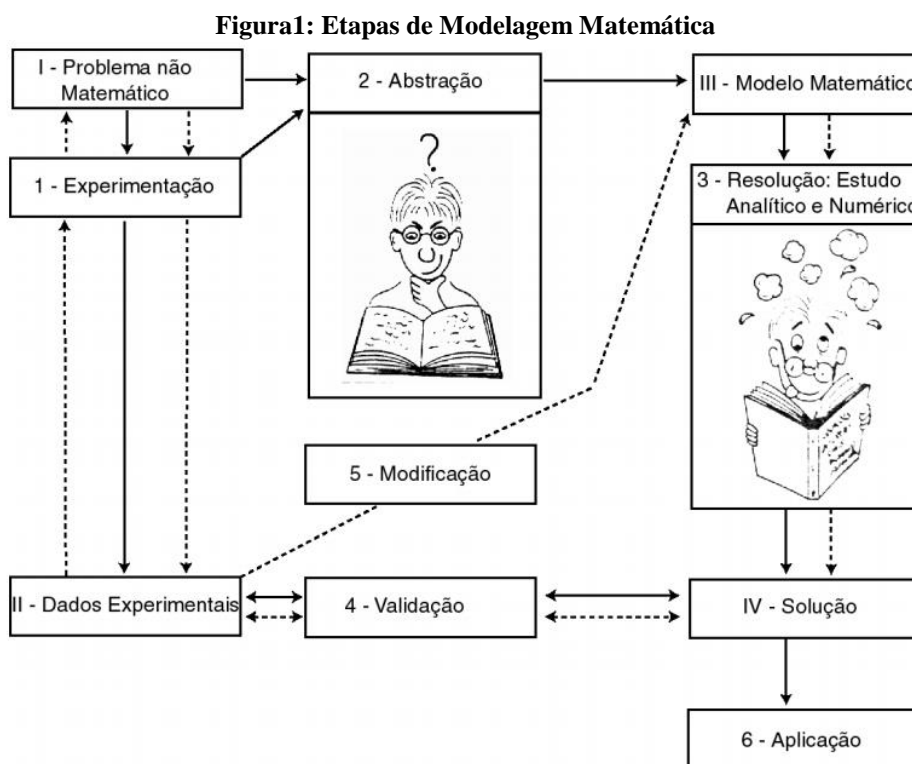
Segundo Bassanezi (2002), a modelagem matemática possui cinco etapas: experimentação, abstração, resolução, validação e modificação. A experimentação ocorre no primeiro contato com a situação problema real. Em geral, não é necessária a presença de um matemático nessa etapa, mas é muito importante que ele esteja presente, pois ele pode direcionar os parâmetros a serem coletados. Nas atividades desenvolvidas, a experimentação não será feita, de forma que os alunos já terão acesso direto aos dados coletados e descrição do experimento.

O próximo passo é a abstração etapa em que é feita a seleção das variáveis, a problematização, a formulação de hipóteses e a simplificação. Todos esses passos devem ser realizados pelos alunos e já é possível delinear a semelhança entre a modelagem matemática e o pensamento computacional.

A próxima etapa é a resolução. Nessa etapa são construídas expressões, funções, figuras que expressem a realidade estudada. Muitos modelos matemáticos são impensáveis sem a programação.

A penúltima etapa é a validação, etapa em que o modelo proposto é aceito ou não. Novamente, a programação possui um papel importante, uma vez que será feito o cálculo dos pontos obtidos através da modelagem e posterior comparação com os dados reais.

Por fim, o processo passa pela etapa de modificação. Nessa etapa o modelo é ajustado de acordo com os primeiros resultados obtidos pela resolução e validação. É nessa etapa que as hipóteses são reavaliadas e verificadas se são falsas ou verdadeiras, assim como é analisado o impacto que a simplificação através da exclusão de um parâmetro pode causar. O esquema abaixo mostra de forma simplificada essas etapas e como elas se relacionam.



Fonte: BASSANEZI, 2002, p.27.

RESULTADOS ESPERADOS

Segundo Feurzeig *et al.* (1970), a inserção do computador nas aulas de matemática cria um ambiente de experimentação matemática. De fato, o aluno poderá “testar”

numericamente os teoremas que são ensinados, verificar se uma hipótese levantada por ele mesmo possui contra exemplos mais simples de serem identificados, perceber em tempo real que seu código possui falhas e, naturalmente, a possibilidade de identificar e depurar o erro.

Dessa forma, a possibilidade de executar cálculos de grandes proporções possibilita a inserção de exemplos e situações problema antes impensáveis. Assim, será possível realizar a contextualização de alguns conteúdos, tornando a Matemática mais “atraente” para os alunos.

Outro ponto positivo da inserção da programação em sala é a utilização da linguagem formal matemática, pois “declarações diretas e precisas não são mais impostas pelo decreto de um professor, mas pelas necessidades óbvias de fazer o computador cumprir suas ordens” (FEURZEIG *et al.* 1970, p.10). Naturalmente, com o uso de uma linguagem mais formal, conceitos abstratos como funções, variáveis e incógnitas ficam mais fáceis de entender. Mais ainda, a utilização de letras para representar elementos numéricos sem confusão de conceitos e a descoberta de padrões em situações reais tornam-se tarefas rotineiras e sem maiores dificuldades. Além disso, espera-se que o aluno crie o hábito de rever suas soluções matemáticas e o processo de buscar possíveis erros se torne mais natural, pois

The process of "debugging" programs gives students a rich base of personal experiences with the activity of solving mathematical problems. It enables them to confront and better understand their own thought processes. Thus, it is a valuable means of contributing to teaching the informal, intuitive, heuristic aspects of mathematical thinking and work.⁴ (FEURZEIG *et al.* 1970, p. 20).

Por outro lado, espera-se que o aluno desenvolva o pensamento computacional, ou seja, as competências e habilidades que o tornem capaz de utilizar os recursos computacionais de forma versátil, criativa e que não tenha dificuldades em se adaptar às novas tecnologias. Por fim, espera-se incentivar alguns alunos a continuarem seus estudos na área da Matemática, Matemática Aplicada, Matemática Computacional, Ciência da Computação, Engenharia da Computação e demais áreas afins.

⁴ O processo de “depurar” programas dá, ao estudante, uma base rica de experiências pessoais com a atividade de resolução de problemas matemáticos. Permite-lhes confrontar e melhor entender seus próprios processos mentais. Portanto, é um meio valioso de contribuir para o ensino de aspectos informais, intuitivos e heurísticos do pensamento e do trabalho matemático.

REFERÊNCIAS

- BARCELOS, T.; SILVEIRA, I. Pensamento Computacional e Educação Matemática: Relações para o Ensino de Computação na Educação Básica. In: Workshop sobre Educação em Computação, 20, 2012, Curitiba.
- BRASIL. MINISTÉRIO DA EDUCAÇÃO. **Base Nacional Comum Curricular**. Brasília: MEC/SEB, 2017.
- BRASIL. MINISTÉRIO DA EDUCAÇÃO. **Parâmetros Curriculares Nacionais: Ensino Médio**. Brasília: MEC/SEB, 1999.
- BRASIL. MINISTÉRIO DA EDUCAÇÃO. **PCN+ Ensino Médio: Orientações Curriculares Complementares aos Parâmetros Curriculares Nacionais**. Brasília: MEC/SEB, 2002.
- BURDEN, R. L.; FAIRES, J. D. **Análise Numérica**, 9. ed. São Paulo: Pioneira Thomson Learning, 2003.
- EVES, H. **Introdução à história da matemática**. 4. ed. Campinas: UNICAMP, 2004
- FEURZEIG, W.; PAPERT, S.; BLOOM, M.; GRANT, R.; SOLOMON, C. **Programming-language as a conceptual framework for teaching mathematics**. 1. Ed., Cambridge, Bolt Beranek and Newman, 1970
- GOLUB, G. H.; LOAN C. F. V. **Matrix computations**, 3. ed, London: The Johns Hopkins University Press, 1996.
- PARLETT, B. Very Early Days of Matrix Computations. In: **SIAM News**. Philadelphia, v. 36, n.9, p. 1-3, nov. 2003. Disponível em: <http://people.cs.uchicago.edu/~fortnow/classes/f04/cmssc311/matrix.pdf>. Acesso em: 10 ago. 2019.
- SAELI, M.; PERRENET, J.; JOCHEMS, W.; ZWANEVELD, B. Teaching Programming in Secondary School: A Pedagogical Content Knowledge Perspective. In: **Informatics In Education**. Vilnius, v. 10, n.1, p. 73-88, abr. 2011. Disponível em: <https://www.researchgate.net/publication/228951140>. Acesso em: 10 ago. 2019.
- STEWART, G. W. **Introduction to Matrix Computations**. 1. ed. New York: Academic, 1973.
- SZLÁVI, P.; ZSAKÓ, L. Programming versus application. In: MITTERMEIR, R.T. (Ed.). **Informatics Education – The Bridge between Using and Understanding Computers**. Vilnius: Springer, 2006, v. 2, p. 48–58.
- WING, J. M. Computational thinking. In: **Communications of the ACM**, New York, v. 49, n. 3, p. 33–35, mar 2006.