

Aprendizado de Programação na Educação Básica com Python: Aplicação em Simulações Financeiras

Audenir N. Petuba,¹ Elthon Alex da S. Oliveira ²

PROFMAT/UFAL, Arapiraca, AL

Rinaldo V. S. Júnior³

CECA/UFAL, Rio Largo, AL

Resumo. Com uso do Python, apresentamos algumas propostas de recursos computacionais para o ensino na educação básica, abordando conceitos relevantes da matemática financeira, como a composição das parcelas no financiamento de uma motocicleta e na tomada de um empréstimo. Acreditamos que o professor de Matemática, por meio dessas atividades, pode despertar no estudante o interesse em conhecer o que há por trás das operações financeiras e, assim, ser capaz de tomar a melhor decisão diante dessas situações. A partir dos conceitos de matemática financeira, especificamente o de sistemas de amortização, organizamos duas propostas como sequências didáticas para o professor de matemática. Tais atividades podem ser aplicadas em um laboratório de informática com acesso à internet.

Palavras-chave. Ensino, Educação Financeira, Sistemas de Amortização, Linguagem Python.

1 Introdução

Neste trabalho apresentamos os sistemas de amortização SAC e Price com uso da linguagem de programação Python. Essa escolha como ferramenta de implementação não apenas simplifica a aplicação dos conceitos teóricos, mas também proporciona flexibilidade e facilidade na manipulação de dados. Essa abordagem se dará por meio de duas propostas de atividade a serem aplicadas no Laboratório de Informática, onde se dará a conexão entre a teoria ensinada em sala de aula e a prática. “Os laboratórios são espaços onde a teoria se transfigura em realidade, o ensinamento abstrato se concretiza em ação, som ou imagem. Os diversos experimentos desenvolvidos nos laboratórios permitem aos alunos verificar empiricamente os conteúdos ensinados na escola” [1].

A escolha do uso da linguagem Python se dá pelo fato de ser uma linguagem de alto nível, isto é, é mais intuitiva e de fácil compreensão e aprendizado para o entendimento humano. Neste sentido, “o Python é indicado para disciplinas introdutórias por ser uma linguagem intuitiva e de fácil uso.” [4].

Estudar matemática financeira é de extrema importância, pois proporciona o entendimento necessário para tomar decisões informadas e conscientes, permitindo uma gestão mais eficaz dos recursos financeiros e a escolha da melhor opção de amortização para cada situação. Quando adquirimos um financiamento de um imóvel ou de um veículo, ou um empréstimo, pagamos a dívida em parcelas que serão definidas pelo sistema de amortização escolhido. No sistema SAC, as parcelas são decrescentes, enquanto no sistema Price as parcelas são constantes [5].

¹audenir.petuba@arapiraca.ufal.br

²elthon@arapiraca.ufal.br

³rinaldo.silva@ceca.ufal.br

Estudantes do Ensino Médio, em sua maioria adolescentes, brevemente entrarão na vida adulta e muitos lidarão com tomada de empréstimos e financiamentos, que deverão ser pagos através do parcelamento de seu valor principal, com cobrança de juros embutidos nas parcelas.

Nessa perspectiva, após se formarem no Ensino Médio, não poucos discentes lidarão com a aquisição de bens e serviços, tais como veículos e imóveis. Alguns se depararão com situações de financiamentos de cursos de graduação e outros cursos de formação profissional, ou ainda, empréstimos para a quitação de faturas/boletos. Diante disso, esse público carece de estratégias que o auxiliem na decisão sobre a aquisição desses e outros itens, de forma consciente. [5]

Este trabalho atende às competências e habilidades da BNCC do Ensino Médio, área de Matemática, que versam sobre o uso de tecnologias e linguagens de programação, dentre as quais destacamos a Competência Específica 4: “Compreender e utilizar, com flexibilidade e precisão, diferentes registros de representação matemáticos (algébrico, geométrico, estatístico, computacional etc.), na busca de solução e comunicação de resultados de problemas” [3] e a habilidade EM13MAT405: “Utilizar conceitos iniciais de uma linguagem de programação na implementação de algoritmos escritos em linguagem corrente e/ou matemática” [3]. Com relação aos conceitos de matemática financeira, indicamos a referência [6].

Na seção 2, apresentamos, inicialmente, a linguagem Python, destacando sua simplicidade e acessibilidade para iniciantes. Em seguida, detalhamos o Sistema de Amortização Constante (SAC) e o Sistema Price, explicando como são calculadas as parcelas em cada sistema de amortização. Finalmente, propomos duas atividades práticas para estudantes de Ensino Médio, utilizando Python para criar tabelas e gráficos que simulam financiamentos, permitindo a análise e comparação entre os sistemas SAC e Price. Na seção 3, apresentamos as considerações finais do trabalho.

2 Linguagem Python e Sistemas de Amortização

No que segue, apresentaremos informações sobre a linguagem de programação Python e abordaremos os sistemas SAC e Price, os dois mais usados no Brasil [5], descrevendo como é composta cada parcela de um empréstimo ou financiamento com cada sistema de amortização, assim como a evolução do saldo devedor. As instituições financeiras podem ter um dos dois sistemas já definidos para uma situação financeira ou podem oferecer ao cliente a possibilidade de escolher um deles. Neste trabalho, estamos levando em conta essa possibilidade de escolha. Apresentaremos duas propostas de atividade, a serem realizadas no laboratório de informática de escola de Ensino Médio.

2.1 Linguagem Python

Python é uma linguagem de programação desenvolvida por Guido van Rossum e lançada pela primeira vez em 1991. É uma linguagem de alto nível, de sintaxe simples e de fácil compreensão.

A linguagem Python se destaca como primeira linguagem aprendida por muitos iniciantes em programação. “Ela foi projetada para ser concisa e fácil de ler, tornando-a mais acessível para iniciantes do que muitas outras linguagens de programação. Como tal, o Python foi construído para ser acessível a iniciantes.” [7] Isso pode ser verificado no comparativo da Figura 1.

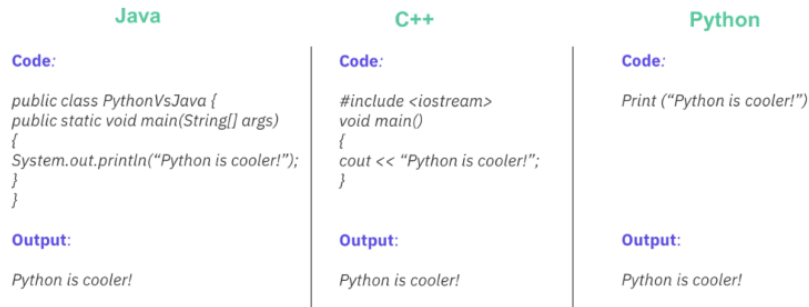


Figura 1: Comparativo entre as linguagens Java (esquerda), C++ (centro) e Python (direita). Fonte: [7]

2.2 Sistema SAC

O Sistema de Amortização Constante (SAC), se caracteriza por ter a amortização de valor constante. Este valor é obtido dividindo-se o valor principal do empréstimo ou financiamento pela quantidade de parcelas da operação financeira. Sendo A o valor da amortização no sistema SAC, V o valor principal e n a quantidade de parcelas, o valor de A é calculado fazendo-se $A = \frac{V}{n}$.

Seja S_k , com $1 \leq k \leq n$, o saldo devedor após o pagamento da k -ésima parcela. Temos que S_k é igual à diferença do saldo devedor no período anterior e da amortização. Assim, $S_k = S_{k-1} - A$.

Sejam J_k os juros na k -ésima parcela e i a taxa de juros por período. O valor de J_k é o produto de S_{k-1} e i , isto é:

$$J_k = S_{k-1} \cdot i. \tag{1}$$

A evolução dos saldos devedores, desde o valor principal até a liquidação da dívida, é mostrada na Tabela 1.

Tabela 1: Evolução do saldo devedor no sistema SAC.

| Número da parcela | Saldo devedor |
|-------------------|-------------------------|
| 0 | $V = S_0$ |
| 1 | $S_1 = S_0 - A$ |
| 2 | $S_2 = S_1 - A$ |
| ... | ... |
| n | $S_n = S_{n-1} - A = 0$ |

Finalmente, seja P_k a k -ésima parcela. O valor de P_k é a soma de A e J_k , isto é, obtemos o valor de cada parcela pela soma dos valores da amortização e dos juros cobrados em cada período. Então, pela Equação 1, temos:

$$P_k = A + S_{k-1} \cdot i. \tag{2}$$

2.3 Sistema Price

No sistema de amortização Price, ou tabela Price, ou sistema de amortização Francês, temos parcelas de valor constante. O valor da parcela é calculado de modo que a soma de todos os pagamentos seja equivalente ao pagamento integral da dívida.

Sejam P o valor da parcela de um empréstimo ou financiamento com sistema Price, V o valor principal, i a taxa de juros e n a quantidade de parcelas a serem pagas. O valor da parcela a

ser paga no sistema Price é dado pela Equação 3, cuja obtenção se dá pela soma dos n primeiros termos de uma progressão geométrica. Para os interessados indicamos [6].

$$P = \frac{V \cdot (1 + i)^n \cdot i}{(1 + i)^n - 1} \quad (3)$$

Sejam J_k os juros na k -ésima parcela, i a taxa de juros por período e S_{k-1} o saldo devedor após o pagamento da $(k - 1)$ -ésima parcela. O valor dos juros é dado por $J_k = S_{k-1} \cdot i$.

Seja A_k a amortização na k -ésima parcela. O valor de A_k é dado pela diferença entre os valores da parcela e dos juros, isto é $A_k = P - J_k \iff A_k = P - S_{k-1} \cdot i$.

O saldo devedor é decrescente, então os juros, que incidem diretamente sobre o saldo devedor, são decrescentes, enquanto a amortização é crescente ao longo do tempo. A evolução dos saldos devedores no sistema Price é análoga àquela observada na Tabela 1.

2.4 Propostas de Atividades: Simulações com Linguagem Python

Apresentaremos, nesta subseção, duas propostas de atividades, que podem ser aplicadas pelo professor de Matemática após a explanação do conteúdo de Matemática Financeira e Sistemas de Amortização em sala de aula.

Proposta 2.1. *Suponha que uma motocicleta custa R\$ 18.000,00. Uma pessoa está disposta a pagar uma entrada de R\$ 3.000,00 e financiar o restante em 18 parcelas, com taxa de 2,5% ao mês, podendo escolher entre os sistemas SAC ou Price. Qual é o sistema mais vantajoso?*

Por meio de uma classe criada em linguagem Python, podemos realizar uma simulação que exhibe, por meio de uma tabela, a descrição das parcelas a serem pagas por esse financiamento. A Figura 2 mostra um trecho da classe em linguagem Python que gera a tabela que descreve as parcelas do financiamento no sistema SAC. Na *linha 9* vemos a definição do valor da amortização como a razão entre o valor principal e a quantidade de parcelas, conforme a equação $A = \frac{V}{n}$.

```

1 class Tabela_SAC:
2     def __init__(self, principal, taxa, parcelas):
3         self.principal = principal
4         self.taxa = taxa / 100.0 # Convertendo a taxa para decimal
5         self.parcelas = parcelas
6
7     def gerar_tabela(self):
8         dash = '-' * 68
9         amortizacao = self.principal / self.parcelas
10        saldo_devedor = self.principal

```

Figura 2: Trecho da classe em Python para gerar a tabela das parcelas do financiamento com SAC.

A Figura 3 mostra um trecho da classe em linguagem Python que gera a tabela que descreve as parcelas no sistema Price. Na *linha 9* definimos o valor da parcela conforme a Equação 3.

```

7     def gerar_tabela(self):
8         dash = '-' * 68
9         parcela = (self.principal * (1 + self.taxa) ** (self.parcelas) * self.taxa) / ((1 + self.taxa) ** (self.parcelas) - 1)
10        saldo_devedor = self.principal

```

Figura 3: Trecho da classe em Python para gerar a tabela com as parcelas no sistema Price.

A Tabela 2 organiza as abordagens metodológicas que podem ser exploradas pelo professor de Matemática na Proposta 2.1.

Tabela 2: Propostas de abordagens metodológicas.

| Abordagem Metodológica | Descrição | Resultado Esperado |
|--|---|---|
| Simulação das Parcelas com SAC e Price | Simular a evolução das parcelas utilizando SAC e Price, observando as características de cada sistema. | Obter parcelas decrescentes com SAC e parcelas constantes de R\$ 1.045,05 em Price (Figura 4). |
| Cálculo da Soma das Parcelas com SAC e Price | Calcular a soma de todas as parcelas no sistema SAC (usando progressão aritmética) e no sistema Price (multiplicando o valor da parcela pela quantidade de períodos) e adicionar o valor pago na entrada para obter o total do financiamento. | Com SAC: total das parcelas de R\$ 18.562,50; com a entrada: R\$ 21.562,50. Com Price: total das parcelas de R\$ 18.810,90; com a entrada: R\$ 21.810,90. |
| Análise Comparativa das Parcelas e Tomada de Decisão | Comparar qual sistema de amortização resulta no menor valor total e em parcelas iniciais menores e decidir qual sistema é mais vantajoso financeiramente. | Escolher SAC para ter o menor valor total dos pagamentos ou escolher o sistema Price para ter parcelas iniciais de menor valor. |
| Uso do Matplotlib, uma biblioteca do Python com vários recursos gráficos, para Visualização Gráfica das Características de cada Sistema de Amortização | Plotar gráficos que permitem observar graficamente a evolução dos juros, amortizações e parcelas em cada sistema de amortização ao longo do tempo. | Perceber, conforme a Figura 5, no sistema SAC, o decrescimento das parcelas (azul), as amortizações constantes (verde) e o decrescimento dos juros (vermelho). Já no sistema Price, verificar as parcelas de valor constante (azul), o crescimento das amortizações (verde) e o decrescimento dos juros (vermelho). |

A Figura 4 mostra a simulação das parcelas com cada sistema de amortização.

| Parcela | Juros | Amortização | Parcela | Saldo Devedor | Parcela | Juros | Amortização | Parcela | Saldo Devedor |
|---------|--------|-------------|---------|---------------|---------|--------|-------------|---------|---------------|
| 0 | 0.00 | 0.00 | 0.00 | 15000.00 | 0 | 0.00 | 0.00 | 0.00 | 15000.00 |
| 1 | 375.00 | 833.33 | 1208.33 | 14166.67 | 1 | 375.00 | 678.05 | 1045.05 | 14329.95 |
| 2 | 354.17 | 833.33 | 1187.50 | 13333.33 | 2 | 358.25 | 686.80 | 1045.05 | 13643.15 |
| 3 | 333.33 | 833.33 | 1166.67 | 12500.00 | 3 | 341.08 | 703.97 | 1045.05 | 12939.17 |
| 4 | 312.50 | 833.33 | 1145.83 | 11666.67 | 4 | 323.48 | 721.57 | 1045.05 | 12217.60 |
| 5 | 291.67 | 833.33 | 1125.00 | 10833.33 | 5 | 305.44 | 739.61 | 1045.05 | 11477.99 |
| 6 | 270.83 | 833.33 | 1104.17 | 10000.00 | 6 | 286.95 | 758.10 | 1045.05 | 10719.89 |
| 7 | 250.00 | 833.33 | 1083.33 | 9166.67 | 7 | 268.00 | 777.05 | 1045.05 | 9942.84 |
| 8 | 229.17 | 833.33 | 1062.50 | 8333.33 | 8 | 248.57 | 796.48 | 1045.05 | 9146.35 |
| 9 | 208.33 | 833.33 | 1041.67 | 7500.00 | 9 | 228.66 | 816.39 | 1045.05 | 8329.96 |
| 10 | 187.50 | 833.33 | 1020.83 | 6666.67 | 10 | 208.25 | 836.80 | 1045.05 | 7493.16 |
| 11 | 166.67 | 833.33 | 1000.00 | 5833.33 | 11 | 187.33 | 857.72 | 1045.05 | 6635.44 |
| 12 | 145.83 | 833.33 | 979.17 | 5000.00 | 12 | 165.89 | 879.17 | 1045.05 | 5756.27 |
| 13 | 125.00 | 833.33 | 958.33 | 4166.67 | 13 | 143.91 | 901.14 | 1045.05 | 4855.13 |
| 14 | 104.17 | 833.33 | 937.50 | 3333.33 | 14 | 121.38 | 923.67 | 1045.05 | 3931.46 |
| 15 | 83.33 | 833.33 | 916.67 | 2500.00 | 15 | 98.29 | 946.76 | 1045.05 | 2984.69 |
| 16 | 62.50 | 833.33 | 895.83 | 1666.67 | 16 | 74.62 | 970.43 | 1045.05 | 2014.26 |
| 17 | 41.67 | 833.33 | 875.00 | 833.33 | 17 | 50.36 | 994.69 | 1045.05 | 1019.56 |
| 18 | 20.83 | 833.33 | 854.17 | -0.00 | 18 | 25.49 | 1019.56 | 1045.05 | -0.00 |

Figura 4: Simulação com sistema SAC (esquerda) e com sistema Price (direita) em linguagem Python.

A Figura 5 mostra os gráficos que ilustram a evolução dos juros, das amortizações e das parcelas ao longo do tempo com cada sistema de amortização.

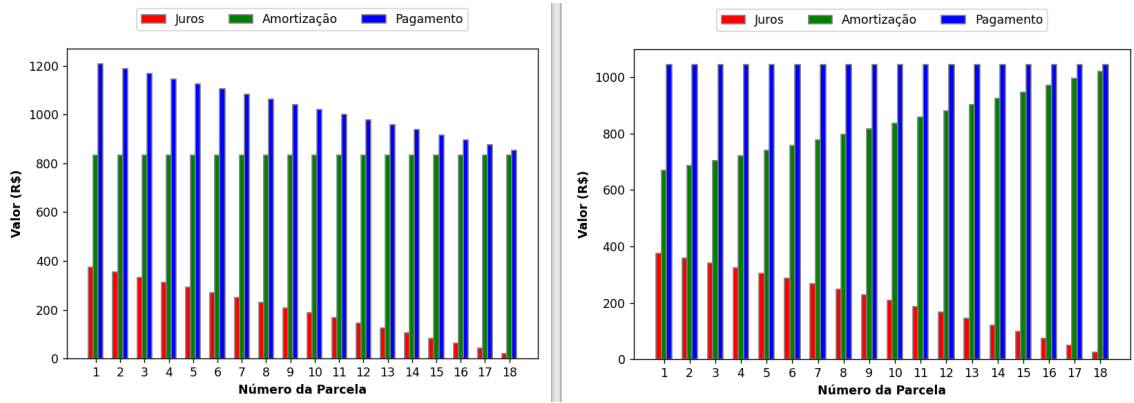


Figura 5: Representações gráficas em Python. SAC (esquerda) e Price (direita).

Proposta 2.2. *Suponha que uma pessoa precise contratar um empréstimo de R\$ 10.000,00, e pagará em 12 parcelas. Ao avaliar duas propostas, verifica que o Banco A oferece o empréstimo com sistema SAC, enquanto o Banco B oferece com sistema Price, ambos com taxa de juros de 5,5% ao mês. Em qual banco essa pessoa deve tomar o empréstimo?*

O professor pode questionar aos estudantes em qual banco essa pessoa pagaria o menor valor total pelo empréstimo. Ao comparar os resultados obtidos nas tabelas geradas em Python (Figura 6), o estudante deve responder que a melhor opção é ofertada pelo Banco A, pois as parcelas resultam no total de R\$ 13.575,00, enquanto no Banco B o total é de R\$ 13.923,48.

| Parcela | Juros | Amortização | Parcela | Saldo Devedor | Parcela | Juros | Amortização | Parcela | Saldo Devedor |
|---------|--------|-------------|---------|---------------|---------|--------|-------------|---------|---------------|
| 0 | 0.00 | 0.00 | 0.00 | 10000.00 | 0 | 0.00 | 0.00 | 0.00 | 10000.00 |
| 1 | 550.00 | 833.33 | 1383.33 | 9166.67 | 1 | 550.00 | 610.29 | 1160.29 | 9389.71 |
| 2 | 504.17 | 833.33 | 1337.50 | 8333.33 | 2 | 516.43 | 643.86 | 1160.29 | 8745.85 |
| 3 | 458.33 | 833.33 | 1291.67 | 7500.00 | 3 | 481.02 | 679.27 | 1160.29 | 8066.58 |
| 4 | 412.50 | 833.33 | 1245.83 | 6666.67 | 4 | 443.66 | 716.63 | 1160.29 | 7349.95 |
| 5 | 366.67 | 833.33 | 1200.00 | 5833.33 | 5 | 404.25 | 756.05 | 1160.29 | 6593.90 |
| 6 | 320.83 | 833.33 | 1154.17 | 5000.00 | 6 | 362.66 | 797.63 | 1160.29 | 5796.28 |
| 7 | 275.00 | 833.33 | 1108.33 | 4166.67 | 7 | 318.80 | 841.50 | 1160.29 | 4954.78 |
| 8 | 229.17 | 833.33 | 1062.50 | 3333.33 | 8 | 272.51 | 887.78 | 1160.29 | 4067.00 |
| 9 | 183.33 | 833.33 | 1016.67 | 2500.00 | 9 | 223.68 | 936.61 | 1160.29 | 3130.39 |
| 10 | 137.50 | 833.33 | 970.83 | 1666.67 | 10 | 172.17 | 988.12 | 1160.29 | 2142.27 |
| 11 | 91.67 | 833.33 | 925.00 | 833.33 | 11 | 117.82 | 1042.47 | 1160.29 | 1099.80 |
| 12 | 45.83 | 833.33 | 879.17 | -0.00 | 12 | 60.49 | 1099.80 | 1160.29 | -0.00 |

Figura 6: Simulação das parcelas do empréstimo com sistema SAC (esquerda) e Price (direita).

Essas propostas de atividades foram aplicadas numa turma da terceira série do Ensino Médio de uma Escola Estadual em Arapiraca - AL. Os sistemas SAC e Price foram apresentados com uma explicação sobre a composição das parcelas, isto é, a soma da amortização e dos juros. Após essa introdução teórica, iniciamos a aplicação das atividades no laboratório de informática da escola. Com a aplicação dessas propostas, os estudantes observaram que o sistema SAC gera parcelas mais altas no início, mas que vão sendo reduzidas ao longo do tempo, resultando em um valor total menor comparado ao sistema Price, sendo estes fatores determinantes para uma boa tomada de decisão financeira.

3 Considerações Finais

O desenvolvimento deste trabalho proporcionou uma experiência enriquecedora para os estudantes do Ensino Médio, explorando os sistemas de amortização SAC e Price através da linguagem de programação Python. A integração entre teoria e prática contribuiu para o desenvolvimento de habilidades em Ciência, Tecnologia, Engenharia e Matemática (STEM) [2], ao permitir a realização de cálculos matemáticos usando linguagem de programação.

Ao aplicar as atividades propostas em um ambiente de laboratório de informática, os estudantes puderam calcular parcelas de financiamentos e visualizar graficamente como esses sistemas de amortização impactam o valor a ser pago ao longo do tempo. Esta abordagem fortaleceu o entendimento teórico adquirido em sala de aula e incentivou uma aprendizagem mais ativa e participativa.

Este trabalho ampliou o conhecimento dos estudantes sobre sistemas de amortização, preparando-os para enfrentar futuros desafios financeiros complexos. O uso de Python como ferramenta de aprendizagem revelou-se uma escolha acertada, possibilitando aos estudantes compreender, aplicar e visualizar os conceitos matemáticos de forma eficaz e estimulante.

Usamos neste trabalho o paradigma de programação POO (Programação Orientada a Objetos), pois o foco foi o estudante manipular os parâmetros dados nas situações-problema para observar o comportamento dos valores nas tabelas geradas. Como trabalho futuro, o código poderá ser refatorado para o paradigma imperativo para facilitar o entendimento do estudante e permitir que ele dê instruções diretas à máquina.

Agradecimentos

Os autores agradecem o apoio financeiro da CAPES.

Referências

- [1] M. Abramovay, M. G. Castro et al. “Ensino médio: múltiplas vozes”. Em: (2003).
- [2] British Council Brasil. **Panorama de educação STEM no Brasil [livreto eletrônico] : reflexões sobre a análise de dados e documentação bibliográfica**. Online. Acessado em 12/03/2024, <https://www.britishcouncil.org.br/atividades/escolas/panorama-de-educacao-stem-no-brasil>.
- [3] Brasil e Ministério da Educação. **Base Nacional Comum Curricular**. Brasília, 2018.
- [4] A. C. Costa, R. Jullian, C. V. A. Mota, A. A. Franco, V. H. J. Muniz, L. de Lima Maia e T. M. Liese. “Python: Será que é possível numa escola pública de Ensino Médio?” Em: **Anais do Workshop de Informática na Escola**. Vol. 23. 1. 2017, pp. 255–264.
- [5] B. Freitas, F. A. Ferreira e V. G. Moreira. “Empréstimos & financiamentos: uma revisão sistemática sobre o ensino de sistemas de amortização”. Em: **Revista Internacional de Pesquisa em Educação Matemática** 11.3 (2021), pp. 151–172.
- [6] L. L. Vendite. **Matemática Financeira e a Utilização de Planilhas Eletrônicas**. Ciência Moderna, 2017.
- [7] K. Walker. **5 Reasons Why Python is the Easiest Coding Language to Learn First**. Online. Acessado em 24/06/2024, <https://codeop.tech/5-reasons-why-python-is-the-easiest-coding-language-to-learn-first/>.