



**UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO**  
**PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO**  
**PROGRAMA DE MESTRADO PROFISSIONAL EM MATEMÁTICA EM**  
**REDE NACIONAL**

**ANTONIO ROBSON BARROS**

**O USO DA LINGUAGEM PYTHON NA VERIFICAÇÃO DE EXPRESSÕES**  
**NUMÉRICAS**

**MOSSORÓ**

**2023**

ANTONIO ROBSON BARROS

O USO DA LINGUAGEM PYTHON NA VERIFICAÇÃO DE EXPRESSÕES  
NUMÉRICAS

Dissertação apresentada ao Mestrado Profissional em Matemática do Programa de Pós-Graduação em Matemática da Universidade Federal Rural do Semiárido como requisito para obtenção do título de mestre em Matemática.

Linha de Pesquisa: Ensino de Matemática.

Orientadora: Profa. Dra. Maria Joseane  
Felipe Guedes Macêdo  
Coorientadora: Profa. Dra. Suene  
Campos Duarte

MOSSORÓ

2023

©Todos os direitos estão reservados à Universidade Federal Rural do Semiárido. O conteúdo desta obra é de inteira responsabilidade do (a) autor (a), sendo o mesmo, passível de sanções administrativas ou penais, caso sejam infringidas as leis que regulamentam a Propriedade Intelectual, respectivamente, Patentes: Lei nº 9.279/1996, e Direitos Autorais: Lei nº 9.610/1998. O conteúdo desta obra tornar-se-á de domínio público após a data de defesa e homologação da sua respectiva ata, exceto as pesquisas que estejam vinculadas ao processo de patenteamento. Esta investigação será base literária **para novas pesquisas, desde** que a obra e seu (a) respectivo (a) autor (a) seja devidamente citado e mencionado os seus créditos bibliográficos.

BA635 u	Barros, Antonio Robson . O uso da linguagem Python na verificação de expressões numéricas / Antonio Robson Barros. - 2023. 77 f. : il.
	Orientador: Maria Joseane Felipe Guedes Macedo. Coorientador: Suene Campos Duarte. Dissertação (Mestrado) - Universidade Federal Rural do Semi-árido, Programa de Pós-graduação em Matemática, 2023.
	1. Linguagem. 2. Python. 3. Matemática. 4. Ensino. 5. Aprendizagem. I. Macedo, Maria Joseane Felipe Guedes , orient. II. Duarte, Suene Campos, co-orient. III. Título.

Ficha catalográfica elaborada por sistema gerador automático em conformidade  
com AACR2 e os dados fornecidos pelo(a) autor(a).  
Biblioteca Campus Mossoró / Setor de Informação e Referência  
Bibliotecária: Keina Cristina Santos Sousa e Silva  
CRB: 15/120

O serviço de Geração Automática de Ficha Catalográfica para Trabalhos de Conclusão de Curso (TCC's) foi desenvolvido pelo Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo (USP) e gentilmente cedido para o Sistema de Bibliotecas da Universidade Federal Rural do Semiárido (SISBI-UFERSA), sendo customizado pela Superintendência de Tecnologia da Informação e Comunicação (SUTIC) sob orientação dos bibliotecários da instituição para ser adaptado às necessidades dos alunos dos Cursos de Graduação e Programas de Pós-Graduação da Universidade.

ANTONIO ROBSON BARROS

O USO DA LINGUAGEM PYTHON NA VERIFICAÇÃO DE EXPRESSÕES  
NUMÉRICAS

Dissertação apresentada ao Mestrado Profissional em Matemática em Rede Nacional do Programa de Pós-Graduação em Matemática da Universidade Federal Rural do Semiárido como requisito para obtenção do título de mestre em Matemática.

Defendida em: 15 / 12 / 2023.

BANCA EXAMINADORA

---

Profa. Dra. Maria Joseane Felipe Guedes (UFERSA)  
Presidente e Orientadora

---

Profa. Dra. Suene Campos Duarte (UFERSA)  
Coorientadora

---

Profa. Dra. Fabiane Regina da Cunha Dantas (UFERSA)  
Membro Examinador (Interno)

---

Prof. Dr. Leomaques Francisco Silva Bernardo (UFCG)  
Membro Examinador (Externo)

## **AGRADECIMENTOS**

Agradeço aos meus pais por terem me colocado neste universo maravilhoso e principalmente a minha genitora Maria Odete de Almeida Barros, por ter me apresentado a vida, o lápis, a caneta, o caderno e livros que não podia comprar em muitas situações.

Agradeço minha família: esposa e filhas pelo incansável incentivo e que sempre me diziam a seguinte frase: “o senhor é dedicado e conseguirá”.

Agradeço aos excelentes professores que ministraram, com dedicação, conhecimento e profissionalismo, as disciplinas do curso.

Agradeço a todos os colegas de turma do PROFMAT 2021 por serem, também, meus incentivadores e parceiros no questionamento de questões.

Agradeço a minha orientadora professora Dra. Maria Joseane Felipe Guedes Macêdo e a minha coorientadora professora Dra. Suene Campos Duarte que sempre estiveram presentes, preocupadas, focadas no objetivo principal deste trabalho.

Agradeço a Banca Examinadora pela postura ética e profissional durante a defesa deste trabalho.

Nada que se refere à humanidade cai tão bem quanto à matemática. Aí, e só aí, tocamos a mente humana, em seu ápice.

Carl Boyer

“Às folhas tantas do livro de matemática, um quociente apaixonou-se um dia doidamente por uma incógnita. Olhou-a com seu olhar inumerável e viu-a, do ápice à base. Uma figura ímpar olhos romboídes, boca trapezoide, corpo ortogonal, seios esferoides. Fez da sua uma vida paralela a dela até que se encontraram no infinito. "Quem és tu?" - indagou ele com ânsia radical. "Eu sou a (raiz quadrada da) soma dos quadrados dos catetos, mas pode me chamar de hipotenusa". E de falarem descobriram que eram o que, em aritmética, corresponde a almas irmãs, primos entre si. E assim se amaram ao quadrado da velocidade da luz sexta potenciação traçando ao sabor do momento e da paixão retas, curvas, círculos e linhas senoidais. Nos jardins da quarta dimensão, escandalizaram os ortodoxos das fórmulas euclidianas e os exegetas do universo finito. Romperam convenções Newtonianas e Pitagóricas e, enfim, resolveram se casar, constituir um lar mais que um lar, uma perpendicular. Convidaram os padrinhos: o poliedro e a bissetriz, e fizeram os planos, equações e diagramas para o futuro, sonhando com uma felicidade integral e diferencial. E se casaram e tiveram uma secante e três cones muito engraçadinhos e foram felizes até aquele dia em que tudo, afinal, vira monotonia. Foi então que surgiu o máximo divisor comum, frequentador de círculos concêntricos viciosos, ofereceu-lhe, a ela, uma grandeza absoluta e reduziu-a a um denominador comum. Ele, quociente percebeu que com ela não formava mais um todo, uma unidade. Era o triângulo tanto chamado amoroso desse problema, ele era a fração mais ordinária. Mas foi então que Einstein descobriu a relatividade e tudo que era espúrio passou a ser moralidade, como, aliás, em qualquer Sociedade”, Millôr Fernandes.

## RESUMO

Este trabalho dissertativo versa sobre a utilização da Linguagem de programação Python com a finalidade de verificar ou resolver expressões numéricas na linguagem matemática. Para justificar essa abordagem prática nas aulas de Matemática, este trabalho respalda-se na Base Nacional Comum Curricular de 2018 que aponta o referido tema como um método de ensino eficiente que utiliza a referida linguagem como um recurso digital e tecnológico nas aulas de Matemática. Dessa forma propomos uma sequência didática, para abordar o conteúdo de expressões numéricas, utilizando a linguagem de programação Python. A sequência foi aplicada em sala de aula e apresentou resultados satisfatórios por parte dos alunos.

**Palavras-chave:** Python; matemática; ensino-aprendizagem.



## **ABSTRACT**

This study deals with the use of the Python programming language with the purpose of verifying or solving numerical expressions in mathematical language. To justify this practical approach in Mathematics classes, this work is supported by the National Common Curricular Base of 2018, which points to the aforementioned topic as an efficient teaching method that uses the aforementioned language as a digital and technological resource in Mathematics classes. Therefore, we propose a didactic sequence to address the content of numerical expressions, using the Python programming language. The sequence was applied in the classroom and showed satisfactory results from the students.

**Keywords:** Python; Mathematics; Teaching-learning.

## LISTA DE FIGURAS

<b>Figura 1</b> – Guido Van Rossun desenvolvedor do <i>Python</i> .....	22
<b>Figura 2</b> – Linguagens de Programação mais utilizadas no mundo.....	23
<b>Figura 3</b> – Filosofia da linguagem <i>Pytohn</i> .....	24
<b>Figura 4</b> – Exemplo do uso do comando <i>type</i> .....	26
<b>Figura 5</b> – Exemplo de comentário (#) .....	27
<b>Figura 6</b> – Alguns exemplos de operadores relacionais.....	30
<b>Figura 7</b> - Exemplos de uso de alguns operadores de atribuição.....	31
<b>Figura 8</b> – Exemplos de modos de cálculo de potências.....	32
<b>Figura 9</b> – Exemplo de cálculo usando a radiciação.....	33
<b>Figura 10</b> – Uso do parêntesis.....	34
<b>Figura 11</b> – Uso da função <i>format</i> .....	35
<b>Figura 12</b> – Um simples programa de adição.....	36
<b>Figura 13</b> – Programa usando antecessor e sucessor.....	36
<b>Figura 14</b> – Programa usando múltiplos e raiz.....	37
<b>Figura 15</b> – Programa usando médias.....	38
<b>Figura 16</b> – Programa com unidades de comprimento.....	38
<b>Figura 17</b> – Programa calculando o desconto.....	39
<b>Figura 18</b> – Programa usando operações com frações.....	39
<b>Figura 19</b> – Programa resolvendo uma equação de segundo grau.....	40
<b>Figura 20</b> – Programa resolvendo uma equação de primeiro grau.....	41
<b>Figura 21</b> – Conjuntos numéricos.....	48
<b>Figura 22</b> – Ordem de precedência.....	51
<b>Figura 23</b> – O maior queijo.....	56
<b>Figura 24</b> – Imagem do resultado da expressão no <i>Python</i> .....	59
<b>Figura 25</b> – Imagem do resultado da expressão no <i>Python</i> .....	60
<b>Figura 26</b> – Imagem do resultado da expressão no <i>Python</i> .....	61
<b>Figura 27</b> - Imagem do resultado da expressão no <i>Python</i> .....	62
<b>Figura 28</b> - Imagem do resultado da expressão no <i>Python</i> .....	63
<b>Figura 29</b> - Imagem do resultado da expressão no <i>Python</i> .....	64
<b>Figura 30</b> - Imagem do resultado da expressão no <i>Python</i> .....	64

<b>Figura 31-</b> Imagem do resultado da expressão no <i>Python</i> .....	65
<b>Figura 32 -</b> Imagem do resultado da expressão no <i>Python</i> .....	66
<b>Figura 33 -</b> Imagem do resultado da expressão no <i>Python</i> .....	67
<b>Figura 34 -</b> Imagem do resultado da expressão no <i>Python</i> .....	67
<b>Figura 35 -</b> Imagem do resultado da expressão no <i>Python</i> .....	68
<b>Figura 36-</b> Imagem do resultado da expressão no <i>Python</i> .....	68

## LISTA DE QUADROS

<b>Quadro 1</b> – Comando <i>and</i> .....	28
<b>Quadro 2</b> – Comando <i>or</i> .....	28
<b>Quadro 3</b> – Comando <i>not</i> .....	29
<b>Quadro 4</b> – Operador de atribuição.....	30
<b>Quadro 5</b> – Operador básico.....	31
<b>Quadro 6</b> – Sequência didática.....	54
<b>Quadro 7</b> - Comparativo entre linguagem.....	59
<b>Quadro 8</b> – Comparativo entre linguagem.....	60
<b>Quadro 9</b> – Comparativo entre linguagem.....	60
<b>Quadro 10</b> – Comparativo entre linguagem.....	61
<b>Quadro 11</b> – Comparativo entre linguagem.....	62
<b>Quadro 12</b> – Comparativo entre linguagem.....	63
<b>Quadro 13</b> – Comparativo entre linguagem.....	64
<b>Quadro 14</b> – Comparativo entre linguagem.....	65

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO .....</b>	<b>14</b>
<b>2</b>	<b>BNCC E O USO DE NOVAS TECNOLOGIAS NAS AULAS DE MATEMÁTICA. 17</b>	
2.1	A Matemática e a BNCC.....	17
2.2	Uso de tecnologias nas aulas de Matemática .....	18
2.3	A Linguagem de Programação Python .....	20
2.3.1	Um Breve Histórico sobre a Linguagem Python .....	22
2.3.2	Principais Conceitos e Comandos do Python.....	24
2.4	Problemas propostos utilizando a linguagem Python .....	34
<b>3.</b>	<b>CONCEITOS PRELIMINARES SOBRE EXPRESSÕES NUMÉRICAS.....</b>	<b>40</b>
3.1	Conjuntos Numéricos .....	40
3.2	Números Naturais.....	40
3.3	Números Inteiros .....	41
3.4	Números Racionais .....	41
3.5	Números Irracionais: um pouco da história.....	43
3.6	Números Reais .....	46
3.7	Expressões Numéricas .....	49
<b>4</b>	<b>SEQUÊNCIA DIDÁTICA COM USO DO PYTHON PARA EXPRESSÕES NUMÉRICAS .....</b>	<b>52</b>
4.1	Sequência didática proposta .....	52
4.2	Um caso particular para a sequência didática proposta .....	54
4.3	Resultados da aplicação da Sequência Didática em sala de aula .....	64
<b>5</b>	<b>CONSIDERAÇÕES FINAIS.....</b>	<b>68</b>
	<b>REFERÊNCIAS BIBLIOGRÁFICAS.....</b>	<b>69</b>
	<b>ANEXOS .....</b>	<b>74</b>

## 1 INTRODUÇÃO

A Matemática é considerada, pela maioria dos alunos, de difícil compreensão (PISA - Programa internacional de avaliação de estudantes, divulgado a cada 3 anos pela OCDE - Organização para Cooperação e Desenvolvimento Econômico). Cada vez mais precisamos de profissionais que saibam criar e planejar métodos de ensino eficientes e que saibam interpretar dados que descrevam problemas comuns da sociedade. A modelagem e interpretação de situações cotidianas possibilitam uma maior interação entre professor e aluno, um maior envolvimento com o conteúdo e assim uma melhor compreensão dos temas discutidos (referendado pelo artigo: *A Study of the Use of Mathematical Modelling as a Teaching-Learning Strategy* - Lourdes Maria Werle de Almeida e Michele Regiane Dias). Para realizar uma abordagem mais prática nas aulas de Matemática, atualmente, é frequentemente realizado o uso das linguagens de programação. Através da programação, ocorre o despertar do pensar, da criatividade e da inovação, por parte do aluno, na resolução de problemas (RESNICK, 2012).

Conforme a BNCC de 2017, existem objetivos e meios para auxiliar o professor no processo de ensino e aprendizagem de seus alunos. Com base no mundo real, muito do que é observado por estes alunos pode ser levado para sala de aula, analisado e apresentado por meio de problemas e suas resoluções. Alguns problemas na aprendizagem matemática ainda se fazem presentes atualmente, como: baixo desempenho na aprendizagem e na execução dos conhecimentos; instrução de baixa qualidade em matemática; falta de aperfeiçoamento dos profissionais da educação envolvidos, entre outros (PAPERT, 2008). Os núcleos de educação podem favorecer o ensino de Matemática, ofertando a seus profissionais cursos de capacitações que propiciem o acesso a novas metodologias de ensino para que possam fazer uso em sua prática docente. Alves (2016) afirma:

Assim, é necessário que o pedagogo pesquise sobre esta área, pois não temos como ensinar o que não sabemos, e é necessário ter o domínio sobre o que irá ser trabalhado. (ALVES, 2016).

Os professores por meio de novas práticas ou metodologias conseguem instigar, motivar e incentivar seus alunos para o desenvolvimento do conhecimento matemático de forma gradual ou progressiva. Desde o primeiro contato com a Matemática, um mundo de possibilidades pode se abrir, e assim, a busca do conhecimento nesta disciplina pode significar descoberta e construção do saber.

Neste trabalho pretende-se fazer uma investigação sobre o uso da linguagem de programação *Python* nas aulas de Matemática para o Ensino Fundamental voltado para jovens e adultos. Borba, Silva e Gadanidis (2018) evidenciam a importância de formar alunos críticos, conectados às novas tecnologias e capazes de selecionar conhecimentos para serem utilizados na resolução e/ou análise de determinado problema. Dentro deste contexto, ao sugerir o uso da linguagem de programação *Python* na abordagem de temas de Matemática, algumas questões surgem: a linguagem de programação *Python* pode potencializar o processo de ensino-aprendizagem de Matemática? Como inserir conhecimentos da linguagem *Python* junto aos conhecimentos da linguagem matemática de forma eficiente?

Visando restringir nosso campo de atuação, trabalharemos especificamente os impactos do uso da linguagem *Python* no ensino das expressões numéricas. O público-alvo será o corpo discente do ensino fundamental do Centro de Educação de Jovens e Adultos (CEJA) da cidade de Quixeramobim-CE. O objetivo geral deste trabalho é investigar o impacto do uso da linguagem de programação Python como ferramenta metodológica no ensino expressões numéricas. Ademais para responder às questões da pesquisa descritas anteriormente, elencamos os seguintes objetivos específicos:

1. Apresentar a importância da programação em *Python* como abordagem de ensino e aprendizagem da Matemática para alunos do CEJA;
2. Propiciar novas práticas para o ensino da Matemática, mais especificamente, o ensino do tema expressões numéricas;
3. Desenvolver aulas que proporcionem maior interação professor-aluno e maior envolvimento de todos com o conteúdo explorado.
4. Elaboração de uma situação-problema no contexto das expressões numéricas visando justificar os elementos constituintes da expressão e introduzir o aluno ao conteúdo através do problema gerador.
5. Elaboração de uma sequência didática visando a resolução de expressões numéricas utilizando a linguagem matemática e a linguagem de programação Python.

Investigaremos o interesse dos alunos na aprendizagem de programação, a desenvoltura deles nas atividades que envolvem a linguagem Python e como essa utilização pode trazer melhoria na aprendizagem de conceitos matemáticos. Na atividade planejada, busca-se proporcionar um ambiente de aprendizagem em que o professor

desempenha um papel de mediador e que os estudantes sejam protagonistas da própria aprendizagem.

Este trabalho está organizado na seguinte forma: Capítulo 1, composto pela Introdução; Capítulo 2, apresentando a BNCC e o uso de novas tecnologias nas aulas de matemática; Capítulo 3, onde listamos alguns conceitos preliminares necessários ao uso do Python para expressões numéricas; Capítulo 4, onde descrevemos uma sequência didática com uso do Python para expressões numéricas; Capítulo 5, com as considerações finais.



## **2 BNCC E O USO DE NOVAS TECNOLOGIAS NAS AULAS DE MATEMÁTICA**

Neste capítulo falaremos rapidamente sobre a preocupação na abordagem da Matemática de acordo com a Base Nacional Comum Curricular (BNCC) e o uso de novas tecnologias para tal fim. Além disso, vamos abordar alguns dos principais conceitos e comandos utilizados na linguagem Python, pois achamos necessário para o leitor que não tem familiaridade alguma com tal linguagem de programação. Para o leitor mais experiente, aconselhamos omitir a leitura deste capítulo.

### **2.1 A Matemática e a BNCC**

No ano de 2017, o MEC em parceria com o CONSED e UNDIME desenvolveram a “terceira e última versão” (BRASIL, 2017) da BNCC voltadas para a Educação Infantil e o Ensino Fundamental. Das competências registradas para a disciplina de Matemática, a BNCC detalha o seguinte:

- 1.Reconhecer que a Matemática é uma ciência humana, fruto das necessidades e preocupações de diferentes culturas, em diferentes momentos históricos, e é uma ciência viva, que contribui para solucionar problemas científicos e tecnológicos e para alicerçar descobertas e construções, inclusive com impactos no mundo do trabalho.
- 2.Desenvolver o raciocínio lógico, o espírito de investigação e a capacidade de produzir argumentos convincentes, recorrendo aos conhecimentos matemáticos para compreender e atuar no mundo.
- 3.Compreender as relações entre conceitos e procedimentos dos diferentes campos da Matemática (Aritmética, Álgebra, Geometria, Estatística e Probabilidade) e de outras áreas do conhecimento, sentindo segurança quanto à própria capacidade de construir e aplicar conhecimentos matemáticos, desenvolvendo a autoestima e a perseverança na busca de soluções.
- 4.Fazer observações sistemáticas de aspectos quantitativos e qualitativos presentes nas práticas sociais e culturais, de modo a investigar, organizar, representar e comunicar informações relevantes, para interpretá-las e avaliá-las crítica e eticamente, produzindo argumentos convincentes.
- 5.Utilizar processos e ferramentas matemáticas, inclusive tecnologias digitais disponíveis, para modelar e resolver problemas cotidianos, sociais e de outras áreas de conhecimento, validando estratégias e resultados.
- 6.Enfrentar situações-problema em múltiplos contextos, incluindo situações imaginadas, não diretamente relacionadas com o aspecto prático utilitário, expressar suas respostas e sintetizar conclusões, utilizando diferentes registros e linguagens (gráficos, tabelas, esquemas, além de texto escrito na língua materna e outras linguagens para descrever algoritmos, como fluxogramas, e dados).
- 7.Desenvolver e/ou discutir projetos que abordem, sobretudo, questões de urgência social, com base em princípios éticos, democráticos, sustentáveis e solidários, valorizando a diversidade de opiniões de indivíduos e de grupos sociais, sem preconceitos de qualquer natureza.
- 8.Interagir com seus pares de forma cooperativa, trabalhando coletivamente no planejamento e desenvolvimento de pesquisas para responder a questionamentos e na busca de soluções para problemas, de modo a identificar

aspectos consensuais ou não na discussão de uma determinada questão, respeitando o modo de pensar dos colegas e aprendendo com eles. (BRASIL, 2017)

Os pontos anteriormente mencionados fazem parte de um processo de estudos que visa auxiliar as escolas e os educadores, com o procedimento a ser adotado para que os alunos compreendam a importância da aprendizagem matemática. Em particular, as Tecnologias Digitais de Informação e Comunicação (TDIC) vêm se destacando no cenário educacional por serem ferramentas atrativas.

## 2.2 Uso de tecnologias nas aulas de Matemática

A palavra tecnologia vem do grego τέχνη (técnica, arte, ofício) e λογία (estudo). É o conjunto de técnicas, habilidades, métodos e processos usados na produção de bens ou serviços, ou na realização de objetivos, como em investigações científicas. Até chegarmos ao que estamos presenciando neste momento sobre o avanço tecnológico, o ser humano, desde a pré-história, vem criando e fazendo uso de tecnologias. Muitos utensílios e ferramentas foram criados em todas as épocas da existência humana (KENSKI, 2013; MARCONDES FILHO, 1994).

Diante do atual contexto em que as tecnologias se fazem presentes no cotidiano, o uso de recursos tecnológicos, computadores, *tablets* e *softwares* podem vir a ser um grande aliado para o professor conforme mostram os Parâmetros Curriculares Nacionais (2017). Contudo, também é imprescindível que a escola compreenda e incorpore mais as novas linguagens e seus modos de funcionamento, desvendando possibilidades de comunicação (e de manipulação), que eduque para usos mais democráticos das tecnologias e para uma participação mais consciente na cultura digital. Ao aproveitar o potencial de comunicação do universo digital, a escola pode instituir novos modos de promover a aprendizagem, a interação e o compartilhamento de significados entre professores e estudantes. (BRASIL, 2017).

A utilização da tecnologia em sala de aula e em especial em aulas de Matemática é de fundamental importância, pois significa deixar de lado a estratégia tradicional que docentes utilizavam onde o foco era o professor e os discentes eram apenas meros espectadores ou ouvintes. Nesse contexto o docente era o centro das atenções com aulas meramente expositivas.

De acordo com (Sá; Machado, 2017) o uso das tecnologias na sala de aula vem se tornando uma ferramenta de grande importância, pois consegue auxiliar tanto o

professor na explicação, quanto o aluno e na compreensão dos conteúdos. Com a tecnologia na aula, os alunos sentem-se mais motivados a aprender e a partir disso o docente consegue ensinar de forma mais dinâmica e criativa.

Segundo Eves (2004) a história da Matemática começa com o surgimento do número, que se deu com a necessidade de contar objetos e coisas, isso há milhares de anos. Na pré-história, o homem vivia da caça e da pesca; com o passar do tempo, foram evoluindo e começaram a plantar. Neste contexto, surgiu a necessidade de contar. A contagem começou com os dedos das mãos, quando era insuficiente, formavam-se montes de pedras, agrupados de cinco em cinco.

Para (BETTEGA, 2010) o ensino é um processo educativo que sempre está em avaliação e revisão, atendendo às necessidades dos alunos. Portanto, não é um processo de entrada e saída de informações. Este processo tem como foco uma prática social global, de forma indispensável, assim, ao avanço tecnológico. A calculadora é sem dúvida um instrumento de tecnologia que no passado surpreendeu e inovou. Conforme a história, o ábaco foi a primeira calculadora criada pelo povo Chinês mais ou menos no século VI a.C., até que em 1642 o francês Blaise Pascal criou uma máquina mais avançada, capaz de efetuar cálculos para a época. O uso da calculadora ficou praticamente obsoleto ou restrito, entre o final do século XIX e início do século XX. Anos depois desse período, o homem investiu em tecnologia, aprimorando e minimizando a parte física (tamanho) e o valor de compra ou aquisição destes aparelhos.

Para Selva e Borba (2010), o uso da calculadora na sala de aula ainda não é um consenso entre os professores e, segundo as autoras, isto está relacionado ao fato de que uma grande parcela dos educadores ainda concebe a calculadora apenas como uma ferramenta útil para a realização de cálculos, conferência de resultados e aplicação dos conhecimentos adquiridos a partir das explicações do professor, não conseguindo reconhecer que este objeto tecnológico pode contribuir para o desenvolvimento conceitual de seus alunos. A calculadora é um instrumento de uso popular. Ela é fruto do desenvolvimento tecnológico alcançado pela humanidade, faz parte do nosso presente e fará do nosso futuro. D'Ambrósio (1986).

Ratificando sobre o uso de novas tecnologias em aulas de matemática Melo (2013, p.39) afirma que:

O professor de Matemática, por exemplo, deve se apropriar de tais tecnologias e dominar suas linguagens para então comunicar os conteúdos

aos alunos, dispondo de recursos dinâmicos que proporcionam a realização de cálculos e construção de gráficos de modo mais rápido e preciso. E que desta forma, softwares que se destinam ao ensino de matemática propiciem novas formas de aprendizagem para estimular o pensar e o fazer Matemática nas atividades escolares (MELO, 2013, p.39).

## 2.3 A Linguagem de Programação Python

Inicialmente o termo ou a palavra “linguagem” nos leva de forma natural a pensar na palavra comunicação. Alguns historiadores acreditam que os antigos romanos cobravam uma taxa como forma de multa entre os habitantes que falavam demais, chamada de *linguarium*. Portanto, a palavra latina *lingua* sempre esteve relacionada com o ato da fala, da comunicação.

A concepção etimológica nos apresenta a linguagem como uma palavra derivada de outra palavra “língua” provinda do latim. Segundo o pensador Evanildo Bechara, a linguagem representa ou é: “qualquer sistema de signos simbólicos empregados na intercomunicação social para expressar e comunicar ideias e sentimentos, isto é, conteúdos de consciência”. Acrescenta que, “para a boa compreensão de linguagem, é importante também entender o sentido de sistema, signo, símbolo e intercomunicação social” (BECHARA, 2009, p. 28).

Mas o que é realmente a linguagem *Python*? Inicialmente, seguindo o raciocínio textual do pensador referenciado anteriormente pode-se entender que nas linguagens de programação e em particular, na linguagem Python, ocorre a comunicação entre a máquina (computador) e um usuário ou uma determinada pessoa.

A linguagem Python foi edificada e construída com o intuito de ser uma linguagem simples, fácil e abrangente. Qualquer usuário pode utilizar esta linguagem com uma finalidade específica, dentre as quais podemos citar:

1. construção de sites, aplicativos, jogos, dentre outros (o Instagram, globo.com, Netflix, Spotify, Google, Uber, todos usam o Python;
2. análise de dados e ciência de dados, inteligência artificial (I.A), neste caso a linguagem Python é pré-requisito;
3. automação de tarefas ou atividades corriqueiramente repetitivas.

O Python é uma linguagem gratuita, com software livre (*open source* de código aberto), popular e utilizada no mundo inteiro. Talvez seja a mais utilizada e preferida no mercado de trabalho. Dentre as suas facilidades ou vantagens, destacamos que um determinado código em Python pode ser utilizado em diversas plataformas ou

sistemas operacionais, tais como o WINDOWS, o LINUX, MAC, etc. Tal característica a classifica como linguagem MULTIPLATAFORMA.

A rigor, utilizando termos mais técnicos e específicos, a linguagem de programação Python é de alto nível, interpretada, orientada a objetos, funcional, tipada, imperativa e de script (The Python Tutorial). A seguir tem-se a explicação de cada termo utilizado na definição:

- 1. Alto nível:** composta de símbolos mais complexos, compreensível pelo ser humano e não-executável diretamente pela máquina. A linguagem de baixo nível refere-se a linguagem de máquina que trabalha com o sistema binário (zeros e uns).
- 2. Interpretada:** o código fonte é executado por um programa de computador chamado interpretador (em vez de serem compiladas), onde a interpretação e a execução do programa acontecem em tempo real e independente do sistema operacional utilizado.
- 3. Orientada a objetos:** significa organizar o mundo real como uma coleção de objetos que incorporam estrutura de dados e um conjunto de operações que manipulam estes dados.
- 4. Funcional:** trata a computação como uma avaliação de uma função matemática e que evita estado ou dado mutável. Por exemplo, se a variável *x* recebe o valor 2 então esta variável jamais assumirá outro valor.
- 5. Tipada:** o que significa que o tipo de uma variável é determinado em tempo de execução, não em tempo de compilação.
- 6. Imperativa:** o usuário ou o desenvolvedor descreve ou aponta detalhadamente os processos que o computador deve executar ao ler aquele código.
- 7. Script:** uma série de instruções executadas de maneira ordenada ou como um conjunto de instruções para a execução de determinada tarefa em um software.

Alguns estudiosos e pensadores dão o seu depoimento e parecer científico sobre a linguagem Python como listado a seguir.

“Hoje em dia a linguagem Python é mais usada a nível global. Num ranking das vinte linguagens mais populares atualmente, o Python se encontra em segundo lugar empatado com o Java e perdendo apenas para o Java script” (VENTURA, 2020).

“Python é, provavelmente, a linguagem de programação popular, mais fácil e agradável de lidar. O código Python é simples para ler e escrever e consegue ser conciso sem ser enigmático” (SUMMERFIELD, 2012, p.1).

“A linguagem de programação Python é muito interessante como primeira linguagem de programação devido a sua simplicidade e clareza. Embora simples, é também uma linguagem poderosa, podendo ser usada para administrar sistemas e desenvolver grandes projetos. É uma linguagem clara e objetiva, pois vai direto ao ponto, sem rodeios” (MENEZES, 2016, p.24).

### 2.3.1 Um Breve Histórico sobre a Linguagem Python

Dissertar sobre a história de uma determinada linguagem de programação é tratar de indagações imprescindíveis que respondam as seguintes expressões: QUEM? QUANDO? ONDE? COMO? (BLOCH, 2001, p. 50).

A linguagem Python foi idealizada em 1989 e criada pelo matemático e programador holandês Guido Van Rossum, nascido em 31 de janeiro de 1956. Na época, o criador trabalhava no Instituto Nacional de Matemática Computacional de Amsterdam, na Holanda, com a missão de implementar a linguagem ABC e por esta necessidade pessoal, passou a estudar, criando assim a linguagem Python.

**Figura 1** – Guido Van Rossum desenvolvedor do Python



**Fonte:** LinkedIn<sup>1</sup> (2023).

A palavra Python, utilizada pelo autor da linguagem, foi extraída do título de uma comédia de séries de TV da época (década de 1970). Trata-se da série Fly circus de month Python, disponível atualmente na Netflix.

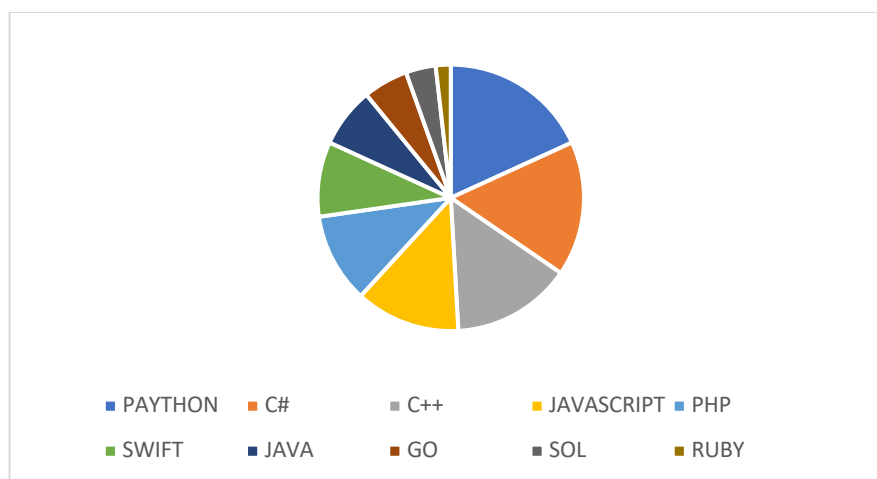
---

<sup>1</sup> Disponível em: < <https://www.linkedin.com/in/guido-van-rossum-4a0756>> . Acesso em 03 mai. 2023, às 20h00

<sup>2</sup>. ABC é uma linguagem de programação da CWI, Países Baixos. Ela é interativa, estruturada, de alto nível e fácil de aprender e usar

A linguagem Python foi criada para ser uma linguagem de comandos simples, prática, popular e hoje é uma das linguagens mais utilizadas por empresas segundo recentes pesquisas (Top Programming Languages 2022).

**Figura 2** – Linguagens de Programação mais utilizadas no mundo



Fonte: <https://www.hostinger.com.br/tutoriais/linguagens-de-programacao-mais-usadas>

Em de 2001 foi criada a fundação *Python Software Foundation - PSF* que é uma empresa sem fins lucrativos que atua como mantenedora e coordenadora do Python. Atualmente a linguagem Python evolui de acordo com os integrantes desta comunidade, fazendo acertos, ajustes de funções e comandos, acatando os votos e decisões da maioria e enfim implementando estas inovações em uma nova versão.

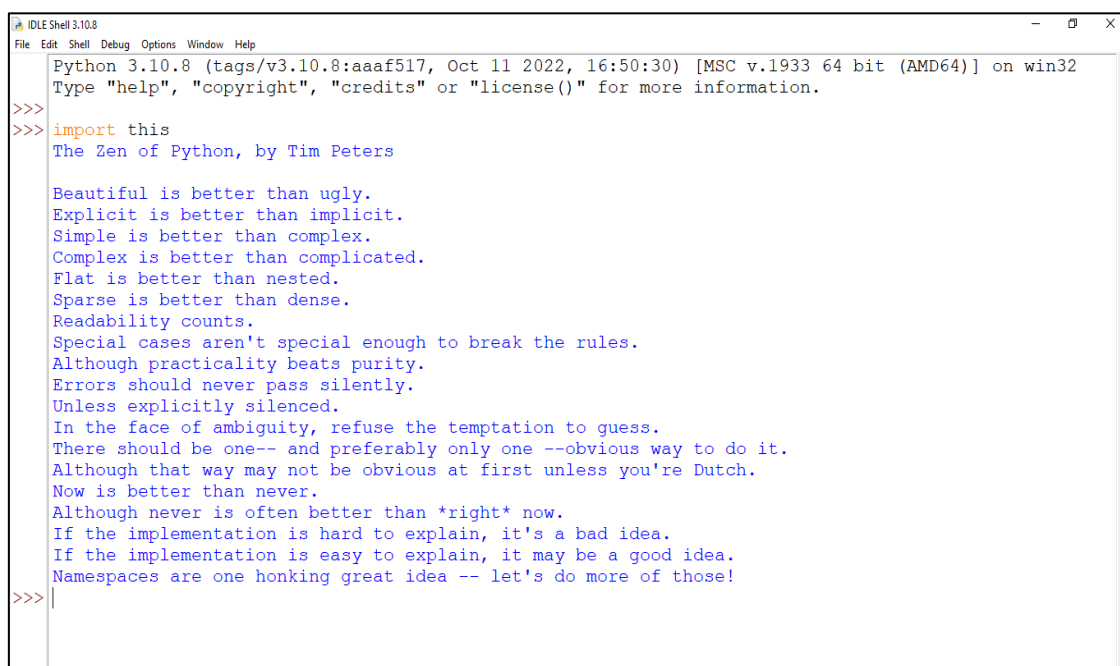
A PSF é apoiada e patrocinada por empresas de renome, tais como a MICROSOFT, GOOGLE e até a empresa brasileira GLOBO.COM. Devido à evolução da linguagem este software já vem instalado em diversos sistemas operacionais tais como AMIGA- OS, NET BSD.MAC- OS e LINUX.

O Python é de código aberto, ou seja, o código fonte é disponibilizado para uso ou modificação por parte de usuários ou de outros desenvolvedores que acharem necessário. Segundo SEVERANCE (2009), algumas características ou especificações da linguagem Python ainda não tratadas ou comentadas neste trabalho são as seguintes:

1. PROPÓSITO GERAL: A linguagem Python é de propósito geral, ou seja, ela não é focada apenas em um propósito, como por exemplo a linguagem JAVA que é focada em desenvolvimento de androides e a linguagem PHP que tem como propósito ou foco, a WEB. A linguagem Python é geral, serve basicamente para tudo.

2. BATTERIES INCLUDED: Imagine um belo brinquedo que você compra e não vem as pilhas ou baterias. A linguagem ou o software, quando baixado, já traz as baterias incluídas, isto é, uma vez baixado e instalado o programa, em sua essência, ali já se tem muita coisa que será utilizado no dia a dia para criação dos códigos, indiferentemente de algumas outras linguagens.
3. BIBLIOTECAS: A linguagem Python possui uma vasta gama de bibliotecas onde algumas servem para criar aplicativo para celular, outras para a criação de jogos, tela de sistemas, dentre outras finalidades.
4. A linguagem Python tem uma filosofia própria criada pelo programador Tim Peters que se chama Zen de Python, do inglês “Zen of Python”. Esta filosofia é composta por uma lista de 19 frases ou princípios orientadores em forma de poema. Relacionamos abaixo a versão original na língua inglesa (na tela do programa já instalado digite ‘import this’) e a sua tradução para a Língua portuguesa.

**Figura 3 – Filosofia da linguagem Python**



```

Python 3.10.8 (tags/v3.10.8:aaaf517, Oct 11 2022, 16:50:30) [MSC v.1933 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
>>> import this
The Zen of Python, by Tim Peters

Beautiful is better than ugly.
Explicit is better than implicit.
Simple is better than complex.
Complex is better than complicated.
Flat is better than nested.
Sparse is better than dense.
Readability counts.
Special cases aren't special enough to break the rules.
Although practicality beats purity.
Errors should never pass silently.
Unless explicitly silenced.
In the face of ambiguity, refuse the temptation to guess.
There should be one-- and preferably only one --obvious way to do it.
Although that way may not be obvious at first unless you're Dutch.
Now is better than never.
Although never is often better than *right* now.
If the implementation is hard to explain, it's a bad idea.
If the implementation is easy to explain, it may be a good idea.
Namespaces are one honking great idea -- let's do more of those!
>>>

```

Fonte: <https://pythonacademy.com.br/zen-of-python>, em 24/05/2023 às 10:11.

### 2.3.2 Principais Conceitos e Comandos do Python

Nesta seção vamos abordar alguns dos principais conceitos e comandos utilizados na linguagem Python. Aqui iremos tratá-los de maneira simples, clara e objetiva. As definições aqui apresentadas estão de acordo com SEVERANCE (2009).



**Definição 2.3.1 (STRINGS):** Strings são sequências de caracteres, símbolos gráficos, tais como as letras de um determinado alfabeto (a, b, c..) ou figuras especiais (@, #, %). As *strings* na linguagem *Python* devem ser escritas ou inseridas dentro de aspas duplas ou aspas simples.

A maioria dos estudiosos e apreciadores desta linguagem utilizam aspas simples ('Python'). Obviamente não existem apenas letras ou caracteres especiais já mencionados, surge também a necessidade de se escrever ou exibir um número. Sendo assim não escreva número no padrão mencionado, entre aspas simples ou duplas. Se, por ventura, isto acontecer, a linguagem Python aceitará e interpretará este seu comando, também, como uma string. Estas aspas são denominadas de **identificador**.

Em síntese, mensagens são utilizadas especialmente para serem exibidas em tela e os números para efetuarem cálculos.

**Definição 2.3.2 (VARIÁVEIS):** Uma variável pode ser entendida como um espaço na memória de um computador destinado a um dado que pode ser alterado durante a execução de um algoritmo. Devem ser declaradas de maneira simples e direta bastando definir ou dar um nome a esta variável e lhe atribuir um determinado valor por exemplo: *var1 = 7* ou *x = 7*, ao invés da nomenclatura *var1*. Assim o que chamamos de *var1* e o que chamamos de *x* recebem o valor igual a 7.

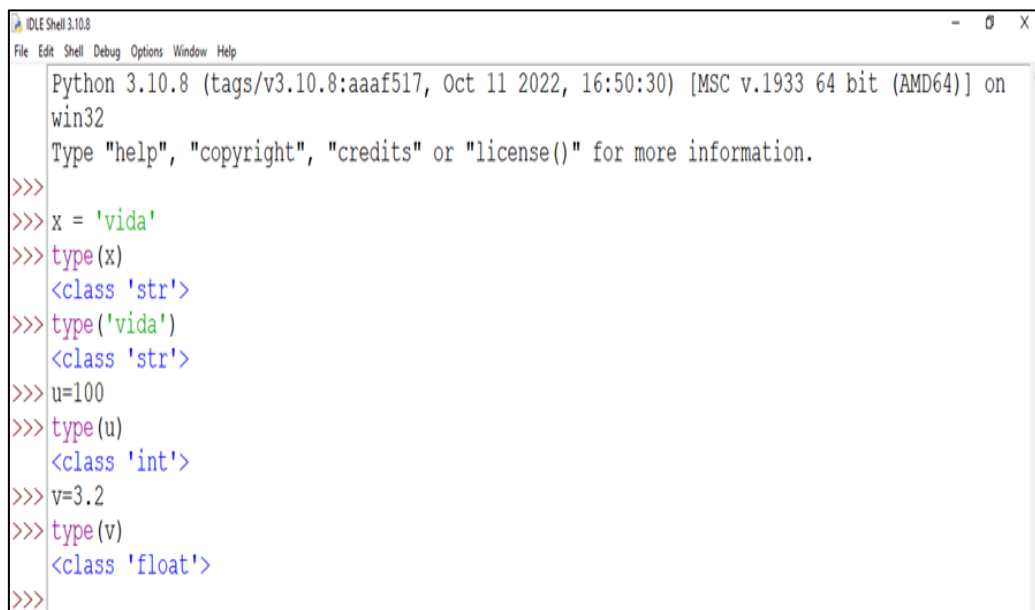
É de suma importância conhecer algumas observações acerca dos nomes das variáveis, pela linguagem Python, as quais listamos:

1. diferenciam letras maiúsculas e letras minúsculas;
2. Não podem ter espaços (caso se deseje fazer alguma separação, deve-se usar o símbolo `_`);
3. Não podem ter acentuação;
4. Não podem usar nome protegido, isto é, alguma palavra reservada ou notação já definida e utilizada pelo vocabulário Python.

Existem na linguagem de programação Python dois tipos importantes de variáveis numéricas, que podem ser verificadas acessando o tutorial da linguagem no endereço, <https://docs.python.org/pt-br/3/tutorial/>. A variável definida como *int* e a variável denominada de *float*. O tipo *int* refere-se a valores inteiros, enquanto o tipo *float* refere-se a números ou parte fracionária, não inteiros ou localizados em pontos flutuantes.

Quando se atribui um valor a uma variável, seja uma palavra (usa-se letra minúscula) ou um número, e se deseja conhecer naquele instante o tipo de variável ali explicitada, é preciso apenas escrever o comando *type* e dentro de um parêntesis a variável em questão: *type* (variável).

**Figura 4 – Exemplo do uso do comando *type***



```
IDLE Shell 3.10.8
File Edit Shell Debug Options Window Help
Python 3.10.8 (tags/v3.10.8:aaaf517, Oct 11 2022, 16:50:30) [MSC v.1933 64 bit (AMD64)] on
win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
>>> x = 'vida'
>>> type(x)
<class 'str'>
>>> type('vida')
<class 'str'>
>>> u=100
>>> type(u)
<class 'int'>
>>> v=3.2
>>> type(v)
<class 'float'>
>>>
```

**Fonte:** Elaborado pelo autor (2023).

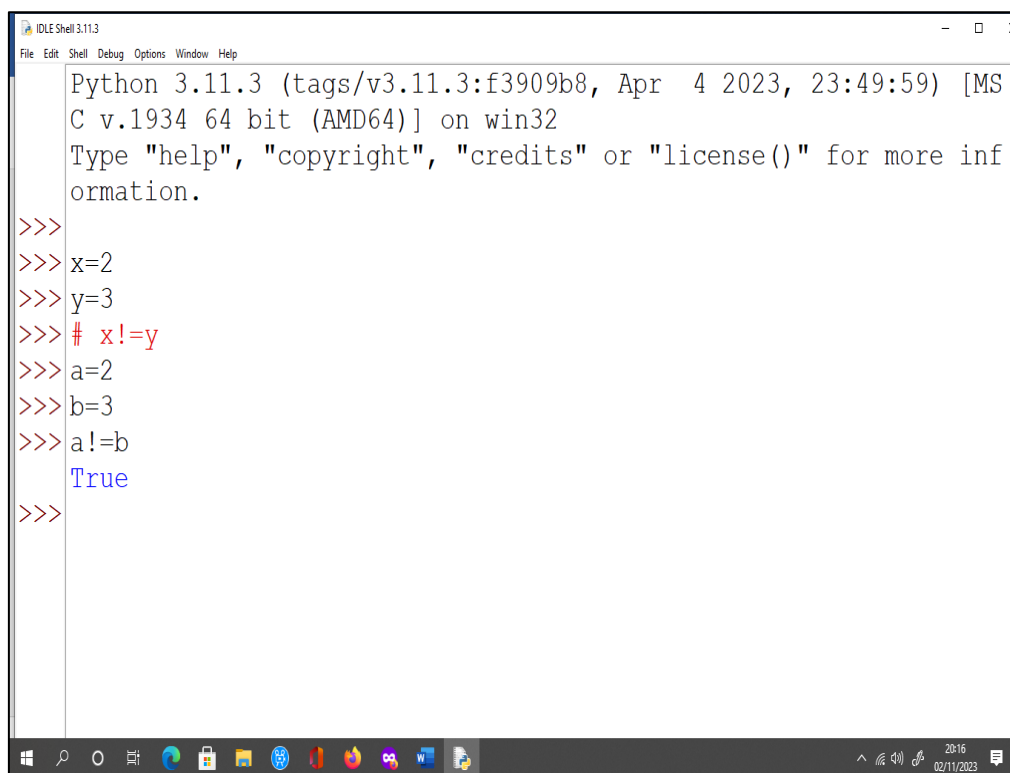
Uma variável pode ser escrita ou declarada de modo explícito ou implícito. O modo explícito é definido quando se escreve o nome da variável e do símbolo = (que significa recebe) acrescido no tipo da variável e dentro de um parêntesis o valor numérico dela. O modo implícito é a negação do modo explícito.

Sabe-se que na linguagem Python não utilizamos a vírgula (,) e sim o ponto (.) para escrevermos um número decimal como costumamos utilizar na linguagem matemática.

**Definição 2.3.3 (COMENTÁRIOS):** Os **comentários** são as ponderações acerca de um texto e em muitas linguagens de programação é possível incluir estes. Para os comentários escritos em uma única linha utiliza-se o símbolo # e, portanto, após encontrar esse símbolo, o que está digitado até o fim da linha é ignorado para fins de execução dentro do programa. Para comentários maiores ou em múltiplas linhas usa-se a sequência “""" (de três aspas) para iniciar e terminar o texto.

Uma característica importante na sintaxe da linguagem Python é que faz parte da organização do código escrito é que a indentação refere-se ao módulo ou distância do início dos caracteres da linha até a margem esquerda. Esse aspecto é bem relevante e indispensável, pois ela indica que a estrutura pertence a um determinado trecho ou bloco do programa. A imagem abaixo mostra trechos de códigos na linguagem Python onde no primeiro e terceiro exemplos não há erro de indentação e, portanto, o interpretador executará o programa e no segundo caso ocorre um erro de indentação gerando assim erro de sintaxe de programa.

**Figura 5 – Exemplo de comentário (#)**



```
Python 3.11.3 (tags/v3.11.3:f3909b8, Apr  4 2023, 23:49:59) [MS
C v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more inf
ormation.
>>>
>>> x=2
>>> y=3
>>> # x!=y
>>> a=2
>>> b=3
>>> a!=b
True
>>>
```

**Fonte:** Elaborado pelo autor (2023).

A linguagem Python utiliza-se de diversos tipos de operadores tais como: operadores lógicos; operadores relacionais; operadores de atribuição e operadores aritméticos.

Dentre os operadores lógicos os mais utilizados são: and (e), or (ou) e not (negativa). Segue a descrição de cada um:

- and: retorna verdadeiro se e somente se receber duas expressões que forem verdadeiras. Veja quadro a seguir:

**Quadro 1 – comando *and***

```

<<< 2 < 3 and 4 < 6 (V, V)
<<< true
<<< 3 < 2 and 8 < 100 (F, V)
<<< false
<<< 23 < 22 and 22 < 21 (F, F)
<<< false

```

**Fonte:** Elaborado pelo autor (2023).

- *or*: retorna falso se, e somente se, receber duas expressões que forem falsas. Veja quadro a seguir:

**Quadro 2 – comando *or***

```

<<< 2 < 1 or 4 < 0 (F, F)
<<< false.
<<< 2 < 5 or 7 < 1 (V, F)
<<< true

```

**Fonte:** Elaborado pelo autor (2023).

- *not*: retorna falso se receber uma expressão verdadeira e vice-versa, ou seja, “*not true*” resulta em “false” e “*not false*” resulta em “*true*”. Veja quadro a seguir:

**Quadro 3 – comando *not***

```

>>> not 3 > 1
>>> false
>>> not 6 < 3
>>> true

```

**Fonte:** Elaborado pelo autor (2023).

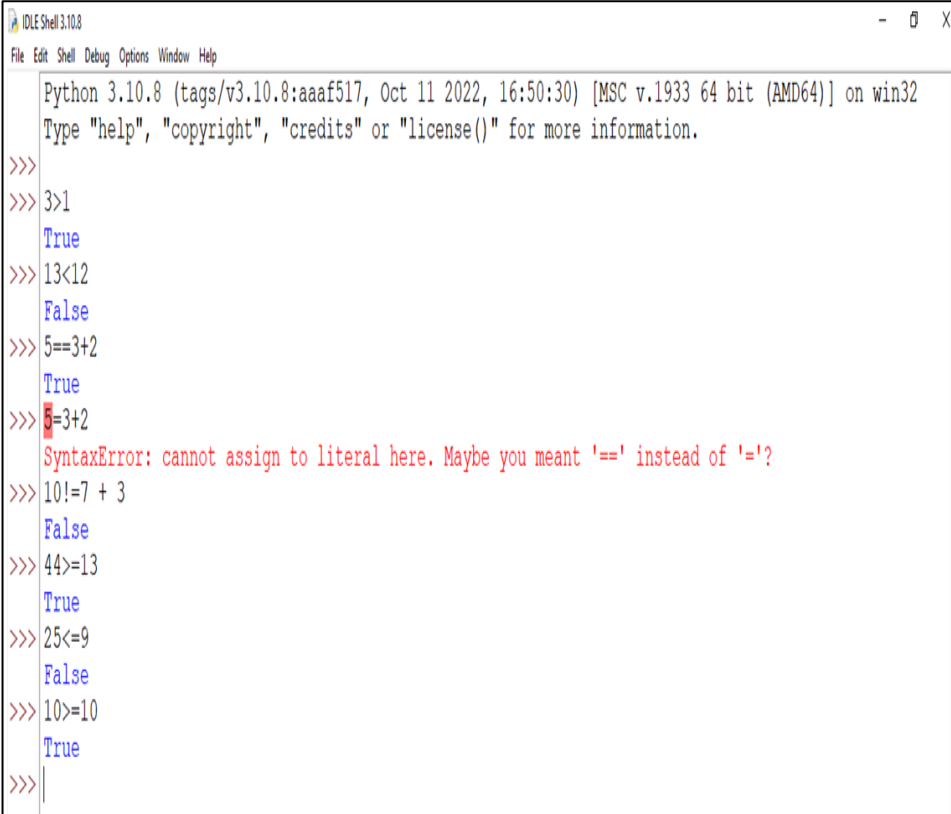
Os dois resultados apresentados, listados anteriormente, (como devem apresentar-se no prompt de comando da linguagem Python), são realmente lógicos pois a

negação de uma sentença verdadeira é algo falso e a negação de uma sentença falsa é algo verdadeiro.

Os operadores relacionais na linguagem Python servem como forma de comparar dois valores, como acontece na linguagem matemática. Por isso, os operadores relacionais são ditos comparativos.

Utiliza-se os seguintes operadores relacionais: maior do que, ( $>$ ); menor do que, ( $<$ ); igual a, ( $=$ ); diferente de, ( $\neq$ ); maior do que ou igual, ( $\geq$ ); menor do que ou igual a, ( $\leq$ ). A Figura 6 apresenta o uso de alguns operadores relacionais.

**Figura 6 – Alguns exemplos de operadores relacionais**



```
Python 3.10.8 (tags/v3.10.8:aaaf517, Oct 11 2022, 16:50:30) [MSC v.1933 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
>>> 3>1
True
>>> 13<12
False
>>> 5==3+2
True
>>> 5=3+2
SyntaxError: cannot assign to literal here. Maybe you meant '==' instead of '='?
>>> 10!=7 + 3
False
>>> 44>=13
True
>>> 25<=9
False
>>> 10>=10
True
>>>
```

**Fonte:** Elaborado pelo autor (2023).

Os operadores de atribuição frequentemente utilizados na linguagem de programação Python são apresentados na tabela a seguir. A figura 7, apresenta um exemplo de uso destes operadores.

**Quadro 4 – operador de atribuição**

Operador	Função
=	Atribuir um determinado valor a uma variável
+=	Soma entre o valor da variável e o valor atribuído
-=	Diferença entre o valor da variável e o valor atribuído
*=	Produto entre o valor da variável e o valor atribuído
%=	Resto da divisão da variável pelo valor atribuído
/=	Divisão entre o valor da variável e o valor atribuído
//=	Parte inteira da divisão da variável pelo valor atribuído

**Fonte:** Elaborado pelo autor (2023).

**Figura 7:** Exemplos do uso de alguns operadores de atribuição

```

Python 3.10.8 (tags/v3.10.8:aaaf517, Oct 11 2022, 16:50:30) [MSC v.1933 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
>>> x=99
>>> print(x)
99
>>>
>>> x=99
>>> x+=2
>>> print(x)
101
>>>
>>> x=99
>>> x-=90
>>> print(x)
9
>>>
>>> x=99
>>> x+=2
>>> print(x)
101
>>>
>>> x=99
>>> x*=2
>>> print(x)
198
>>>
  
```

**Fonte:** Elaborado pelo autor (2023).

Os outros operadores não exemplificados têm a sua operacionalidade análoga aos operadores aqui tratados.

Os operadores aritméticos são essenciais e presentes naturalmente em uma operação matemática, expressão matemática ou numérica. Os principais operadores

aritméticos da linguagem matemática básica e utilizados na linguagem Python e que serão tratados neste trabalho são os apresentados na tabela a seguir.

**Quadro 5-operador básico**

Operador	Função
=	Atribuir um determinado valor a uma variável
+	Soma entre valores de duas variáveis atribuídas
-	Diferença entre valores de duas variáveis atribuídas
*	Produto entre valores de duas variáveis atribuídas
%	Resto da divisão do valor da primeira variável pela segunda variável
/	Divisão do valor de uma variável por outra
//	Valor da parte inteira da divisão de uma variável por outra

**Fonte:** Elaborado pelo autor (2023).

É importante destacar uma peculiaridade quando se trabalha com exponenciação. Pode-se utilizar o operador matemático **\*\*** (duplo asterisco) ou trabalhar com a função interna '**Pow**', sem precisar importar da biblioteca **Math**. Na figura seguinte tem-se exemplos de exponenciação utilizando os dois caminhos.

**Figura 8: Exemplos de modos de cálculo de potências**

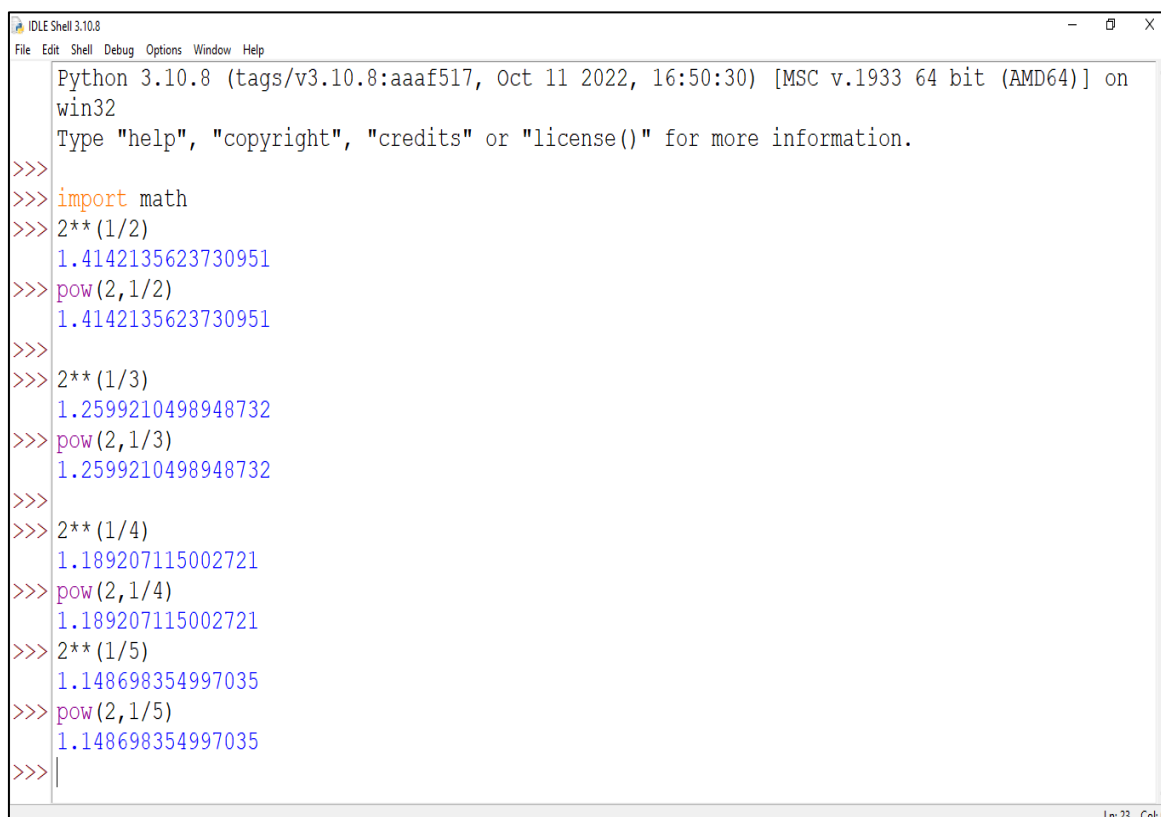
```

Python 3.10.8 (tags/v3.10.8:aaaf517, Oct 11 2022, 16:50:30) [MSC v.1933 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 3+2
5
>>> 3-2
1
>>> 3*2
6
>>> 5/2
2.5
>>> 13%5
3
>>> 2 **10
1024
>>>
  
```

**Fonte:** Elaborado pelo autor (2023).

O cálculo da operação radiciação na linguagem Python pode ser efetuada utilizando um recurso ou a definição matemática de que extrair a raiz quadrada, raiz cúbica, raiz enésima de um número, significa utilizar uma exponenciação cujo valor do expoente será  $\frac{1}{2}, \frac{1}{3}, \dots, \frac{1}{n}$  ou pode-se trabalhar com a biblioteca '*Math*' e utilizar a função '*Pow*'. Na imagem seguinte encontra-se alguns resultados obtidos utilizando a linguagem Python no IDLE. Python 3.10.64-bit.

**Figura 9: Exemplo de cálculo usando a radiciação**



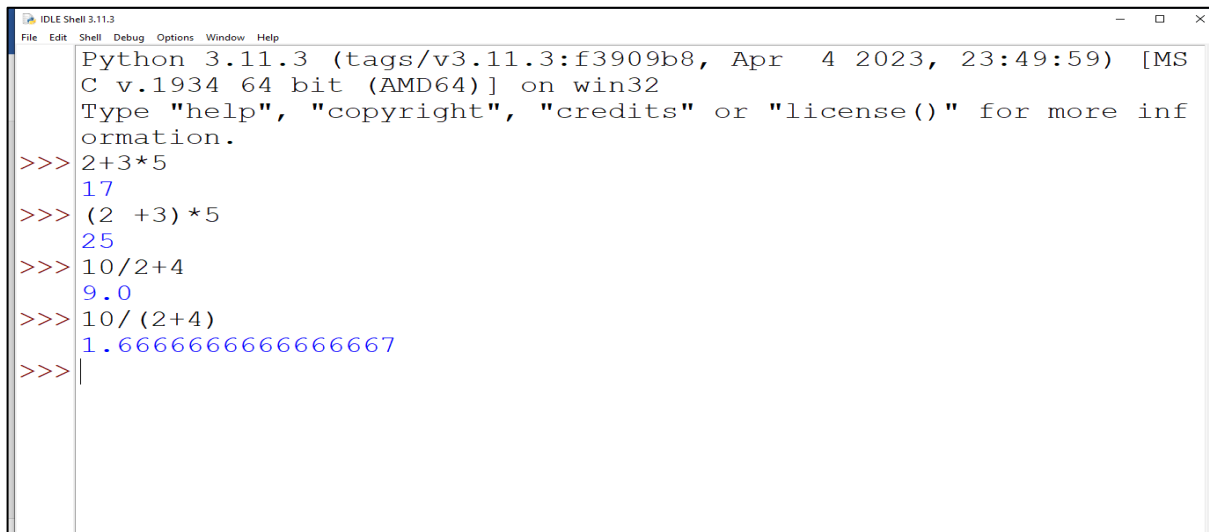
```
Python 3.10.8 (tags/v3.10.8:aaaf517, Oct 11 2022, 16:50:30) [MSC v.1933 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
>>> import math
>>> 2**(1/2)
1.4142135623730951
>>> pow(2,1/2)
1.4142135623730951
>>>
>>> 2**(1/3)
1.2599210498948732
>>> pow(2,1/3)
1.2599210498948732
>>>
>>> 2**(1/4)
1.189207115002721
>>> pow(2,1/4)
1.189207115002721
>>> 2**(1/5)
1.148698354997035
>>> pow(2,1/5)
1.148698354997035
>>> |
```

**Fonte:** Elaborado pelo autor (2023).

Faz-se necessária, neste momento, uma observação quanto ao uso de um simples parêntesis: Na figura 10 perceba que usar ou deixar de usar o símbolo de parêntesis () altera substancialmente o resultado.

**Figura 10: Uso do parêntesis**





```

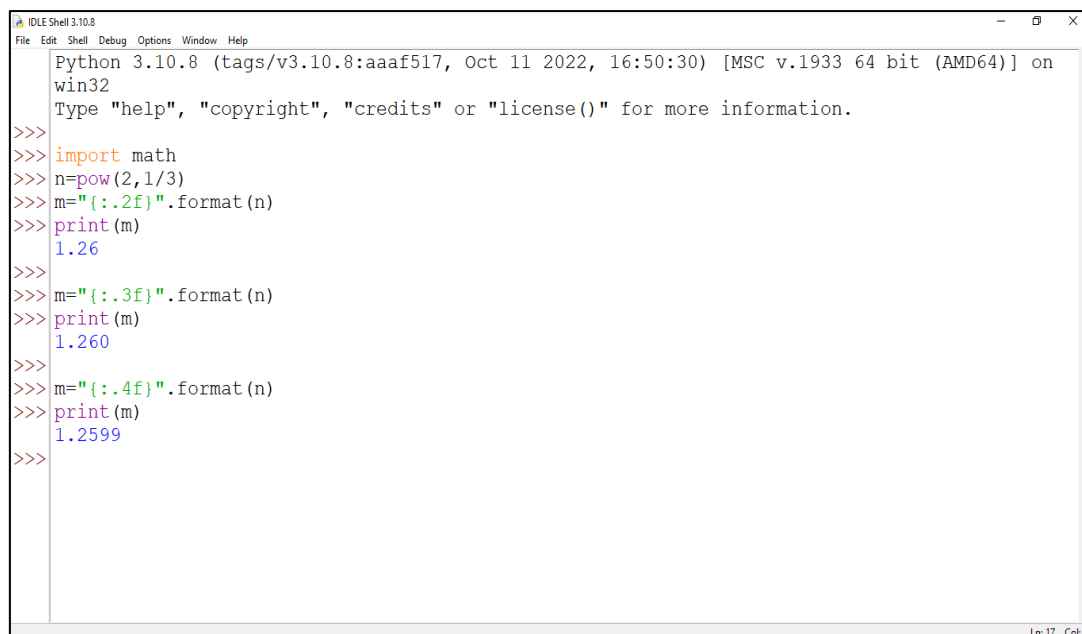
Python 3.11.3 (tags/v3.11.3:f3909b8, Apr  4 2023, 23:49:59) [MS
C v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more inf
ormation.
>>> 2+3*5
17
>>> (2 +3) *5
25
>>> 10/2+4
9.0
>>> 10/(2+4)
1.6666666666666667
>>>

```

**Fonte:** Elaborado pelo autor (2023).

Observa-se nos resultados obtidos para o cálculo da raiz quadrada de 2, raiz cúbica de 2, raiz quarta de 2 e até a raiz quinta de 2 que seus resultados se encontram com 15 casas decimais. Se o usuário pretende utilizar poucas casas decimais, duas, três ou quatro, por exemplo, ele pode recorrer a uma função na linguagem Python denominada de ***Format***. Portanto supondo a existência do número ‘n’ que recebe a raiz cúbica de 2 e outro número ‘m’ que recebe o resultado de ‘n’ com apenas duas, três ou quatro casas decimais e utilizando a função ***Print*** (escreva-se na tela), os resultados serão facilmente obtidos. Vejam a imagem na tela de comando realizado.

**Figura 11: Uso da função Format.**



```

Python 3.10.8 (tags/v3.10.8:aaaf517, Oct 11 2022, 16:50:30) [MSC v.1933 64 bit (AMD64)] on
win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
>>> import math
>>> n=pow(2,1/3)
>>> m="{:.2f}".format(n)
>>> print(m)
1.26
>>>
>>> m="{:.3f}".format(n)
>>> print(m)
1.260
>>>
>>> m="{:.4f}".format(n)
>>> print(m)
1.2599
>>>

```

**Fonte:** Elaborado pelo autor (2023).

## 2.4 Problemas propostos utilizando a linguagem Python

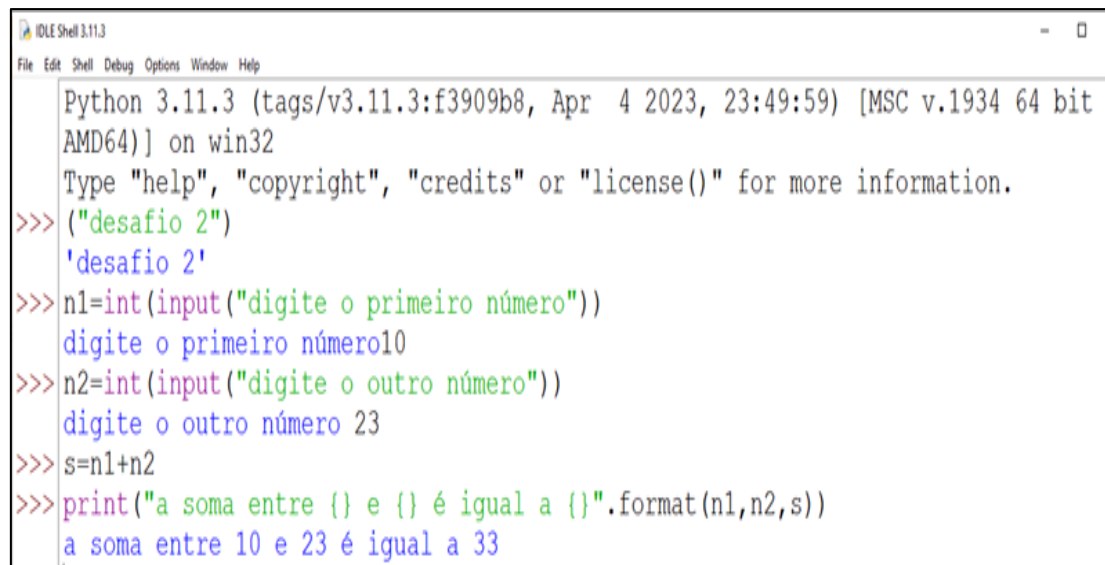
Nesta seção sugerimos a resolução de alguns problemas básicos (desafios) envolvendo a linguagem de programação Python, como forma de aprimorar o conhecimento na referida linguagem. Na sequência didática proposta no Capítulo 4, faremos uso desses desafios como forma de ambientar os alunos com essa linguagem de programação.

**Desafio 1:** Crie um programa que escreva na tela uma mensagem qualquer. Por exemplo, “A matemática e o Python são companheiros”.

**Solução:** `print (“A matemática e o Python são companheiros”)`. A solução também pode ser encontrada no seguinte endereço <https://youtu.be/XwurAzEYrjM>.

**Desafio 2:** Crie um programa que faça a soma de dois números e escreva o resultado de forma explícita. **Solução:** Veja a resposta na Figura 12.

**Figura 12:** Um simples programa de adição



```

Python 3.11.3 (tags/v3.11.3:f3909b8, Apr  4 2023, 23:49:59) [MSC v.1934 64 bit
AMD64] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> ("desafio 2")
'desafio 2'
>>> n1=int(input("digite o primeiro número"))
digite o primeiro número10
>>> n2=int(input("digite o outro número"))
digite o outro número 23
>>> s=n1+n2
>>> print("a soma entre {} e {} é igual a {}".format(n1,n2,s))
a soma entre 10 e 23 é igual a 33

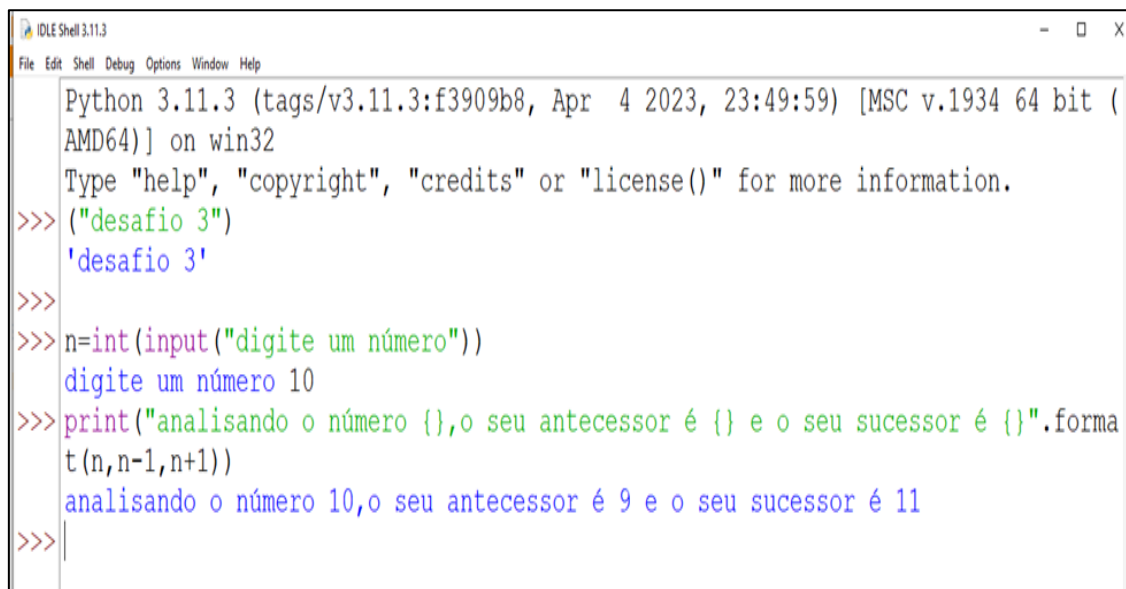
```

**Fonte:** Elaborado pelo autor (2023).

**Desafio 3:** Crie um programa que escreva o antecessor e o sucessor de um número natural  $n$ .

**Solução:** Veja a resposta na Figura 13.

**Figura 13: Programa usando antecessor e sucessor**



```

Python 3.11.3 (tags/v3.11.3:f3909b8, Apr  4 2023, 23:49:59) [MSC v.1934 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> ("desafio 3")
'desafio 3'
>>>
>>> n=int(input("digite um número"))
digite um número 10
>>> print("analizando o número {},o seu antecessor é {} e o seu sucessor é {}".forma
t(n,n-1,n+1))
analizando o número 10,o seu antecessor é 9 e o seu sucessor é 11
>>>

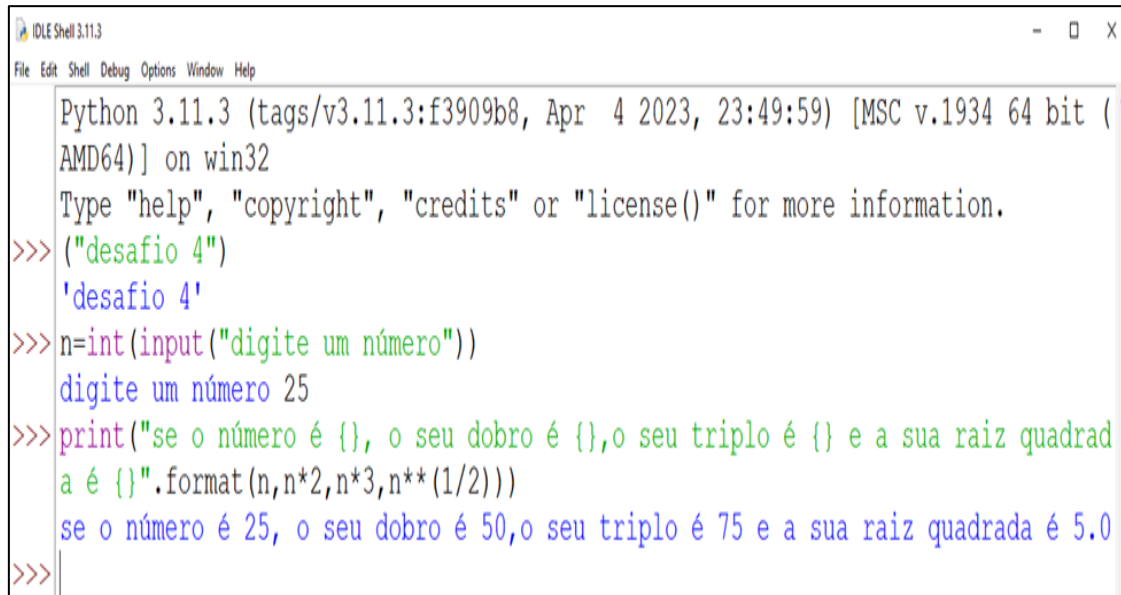
```

**Fonte:** Elaborado pelo autor (2023).

**Desafio 4:** Crie um programa que leia um número e mostre o seu dobro, o seu triplo e a sua raiz quadrada.

**Solução:** Veja a resposta na Figura 14.

**Figura 14: Programa usando múltiplos e raiz**

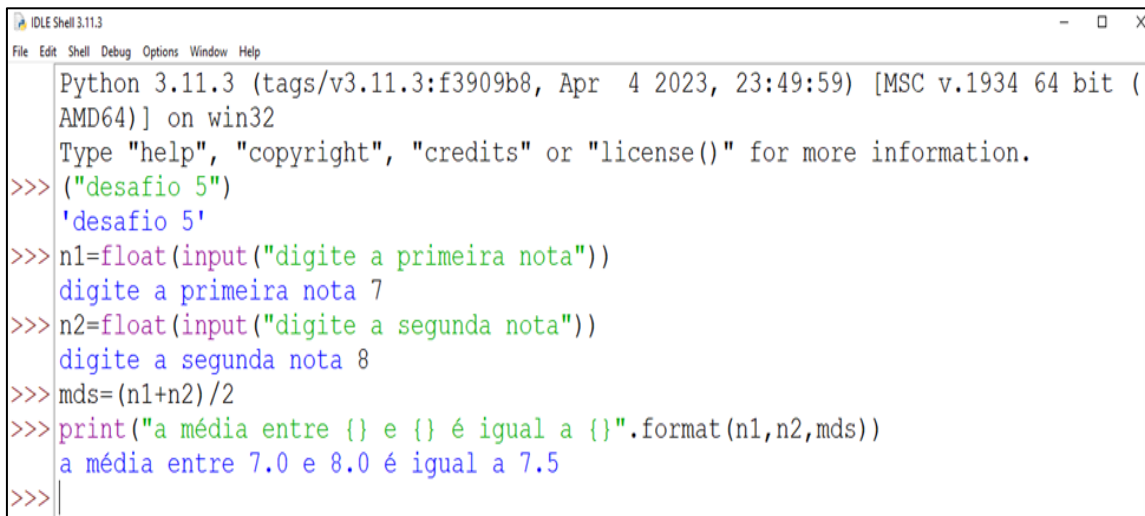


```
Python 3.11.3 (tags/v3.11.3:f3909b8, Apr 4 2023, 23:49:59) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> ("desafio 4")
'desafio 4'
>>> n=int(input("digite um número"))
digite um número 25
>>> print("se o número é {}, o seu dobro é {},o seu triplo é {} e a sua raiz quadrada é {}".format(n,n*2,n*3,n**(1/2)))
se o número é 25, o seu dobro é 50,o seu triplo é 75 e a sua raiz quadrada é 5.0
>>>
```

**Fonte:** Elaborado pelo autor (2023).

**Desafio 5:** Crie um programa que leia duas notas, calcule e mostre na tela o valor de sua média aritmética simples.

**Solução:** Veja a resposta na Figura 15.

**Figura 15: Programa usando médias**


```

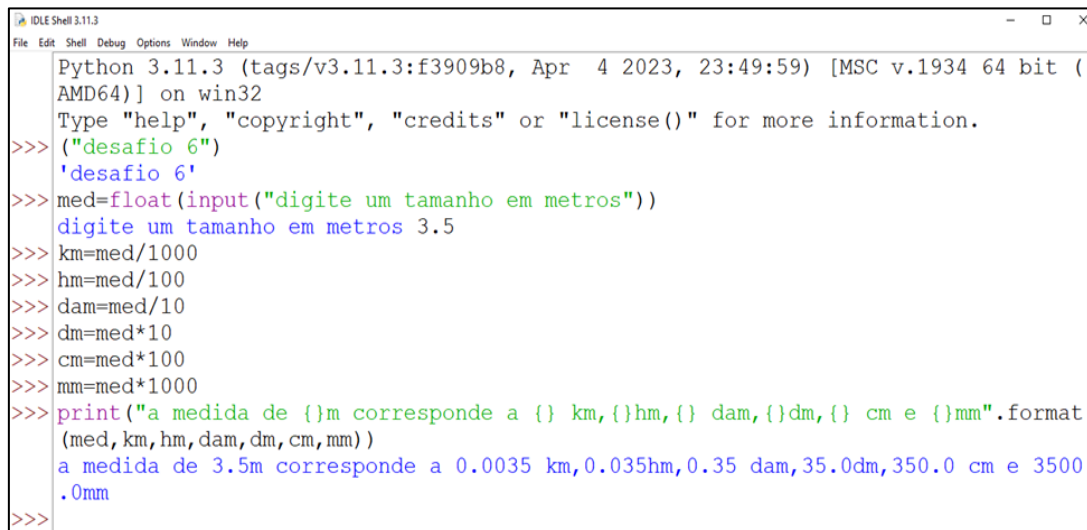
Python 3.11.3 (tags/v3.11.3:f3909b8, Apr  4 2023, 23:49:59) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> ("desafio 5")
'desafio 5'
>>> n1=float(input("digite a primeira nota"))
digite a primeira nota 7
>>> n2=float(input("digite a segunda nota"))
digite a segunda nota 8
>>> mds=(n1+n2)/2
>>> print("a média entre {} e {} é igual a {}".format(n1,n2,mds))
a média entre 7.0 e 8.0 é igual a 7.5
>>>

```

**Fonte:** Elaborado pelo autor (2023).

**Desafio 6:** Crie um programa que leia um valor em metros e faça a transformação em todas as outras unidades de comprimento (múltiplos e submúltiplos do metro).

**Solução:** Veja a resposta na Figura 16.

**Figura 16: Programa com unidades de comprimento**


```

Python 3.11.3 (tags/v3.11.3:f3909b8, Apr  4 2023, 23:49:59) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> ("desafio 6")
'desafio 6'
>>> med=float(input("digite um tamanho em metros"))
digite um tamanho em metros 3.5
>>> km=med/1000
>>> hm=med/100
>>> dam=med/10
>>> dm=med*10
>>> cm=med*100
>>> mm=med*1000
>>> print("a medida de {}m corresponde a {} km, {}hm, {} dam, {}dm, {} cm e {}mm".format(
med, km, hm, dam, dm, cm, mm))
a medida de 3.5m corresponde a 0.0035 km, 0.035hm, 0.35 dam, 35.0dm, 350.0 cm e 3500.0mm
>>>

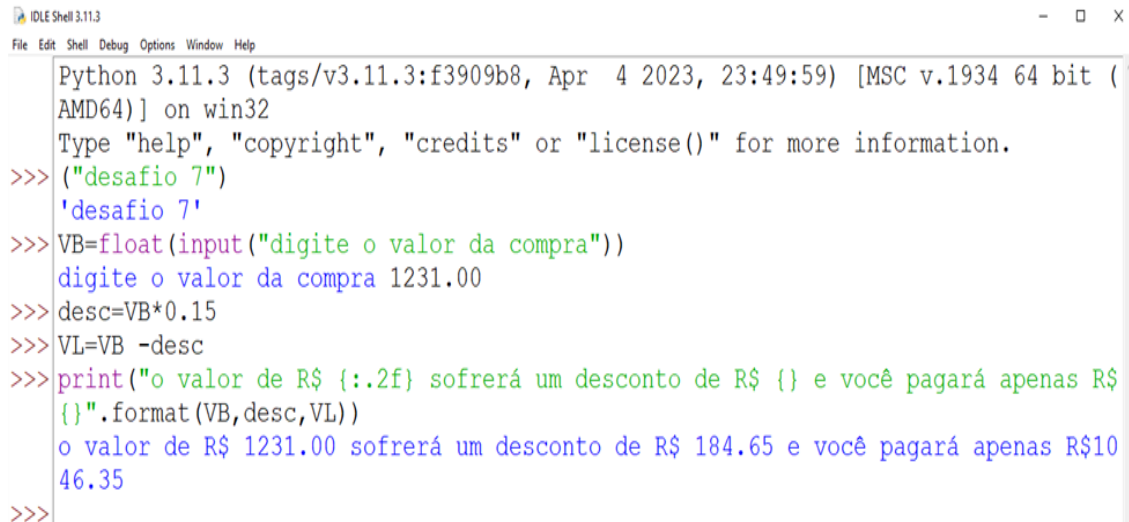
```

**Fonte:** Elaborado pelo autor (2023).

**Desafio 7:** Crie um programa que leia um valor bruto em real de uma compra e exiba na tela o seu valor líquido, após dado um certo desconto.

**Solução:** Veja a resposta na Figura 17.

**Figura 17: Programa calculando o desconto**



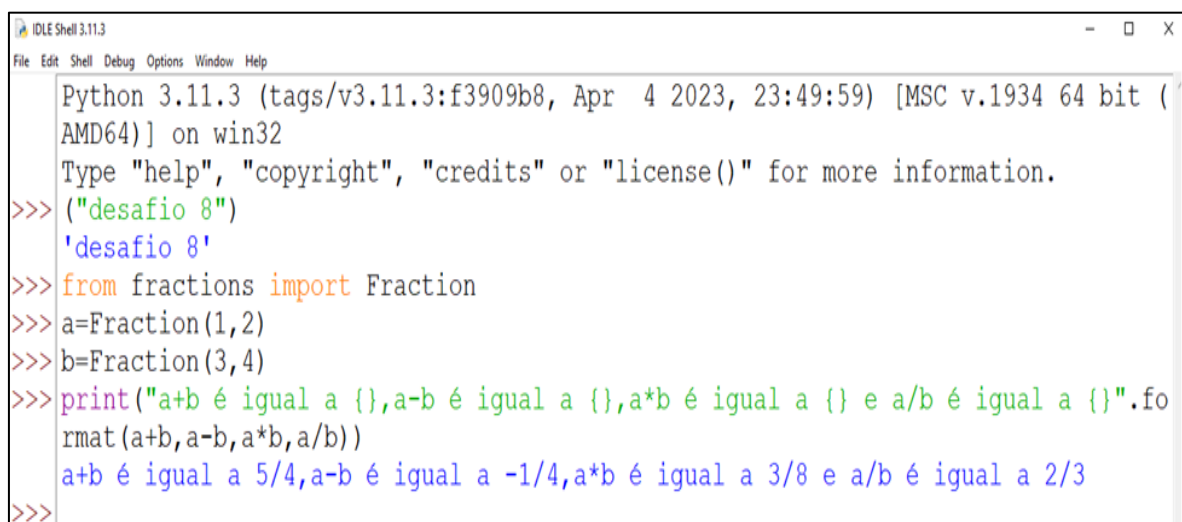
```
Python 3.11.3 (tags/v3.11.3:f3909b8, Apr 4 2023, 23:49:59) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> ("desafio 7")
'desafio 7'
>>> VB=float(input("digite o valor da compra"))
digite o valor da compra 1231.00
>>> desc=VB*0.15
>>> VL=VB -desc
>>> print("o valor de R$ {:.2f} sofrerá um desconto de R$ {} e você pagará apenas R$ {}".format(VB,desc,VL))
o valor de R$ 1231.00 sofrerá um desconto de R$ 184.65 e você pagará apenas R$1046.35
>>>
```

**Fonte:** Elaborado pelo autor (2023).

**Desafio 8:** Crie um programa que some, subtraia, multiplique e divida duas frações quaisquer.

**Solução:** Veja a resposta na Figura 18.

**Figura 18: Programa usando operações com frações**



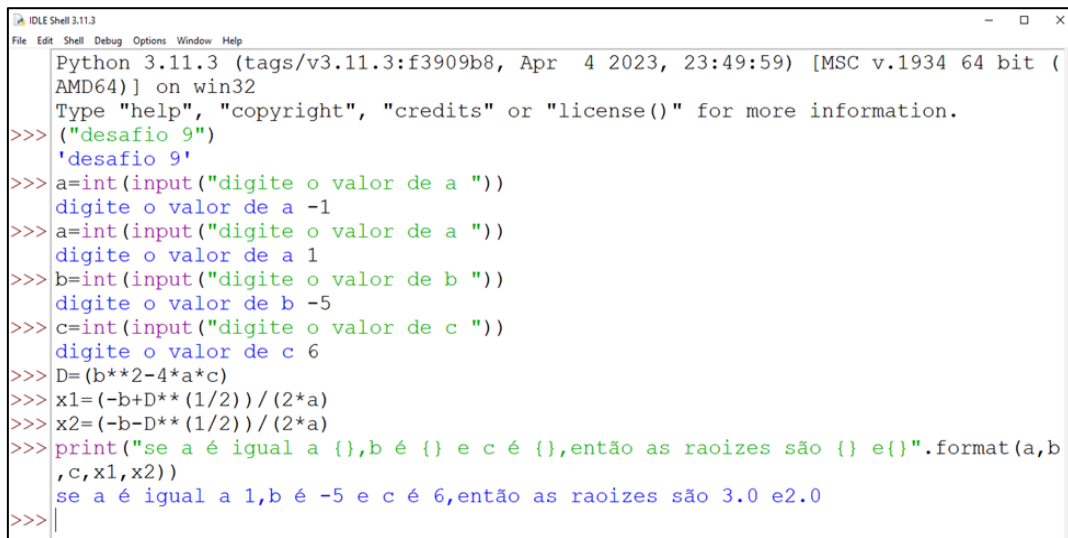
```
Python 3.11.3 (tags/v3.11.3:f3909b8, Apr 4 2023, 23:49:59) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> ("desafio 8")
'desafio 8'
>>> from fractions import Fraction
>>> a=Fraction(1,2)
>>> b=Fraction(3,4)
>>> print("a+b é igual a {}, a-b é igual a {}, a*b é igual a {} e a/b é igual a {}".format(a+b,a-b,a*b,a/b))
a+b é igual a 5/4, a-b é igual a -1/4, a*b é igual a 3/8 e a/b é igual a 2/3
>>>
```

**Fonte:** Elaborado pelo autor (2023).

**Desafio 9:** Crie um programa que resolva uma equação de segundo grau através da fórmula resolutiva.

**Solução:** veja a resposta na Figura 19.

**Figura 19: Programa resolvendo uma equação de segundo grau**



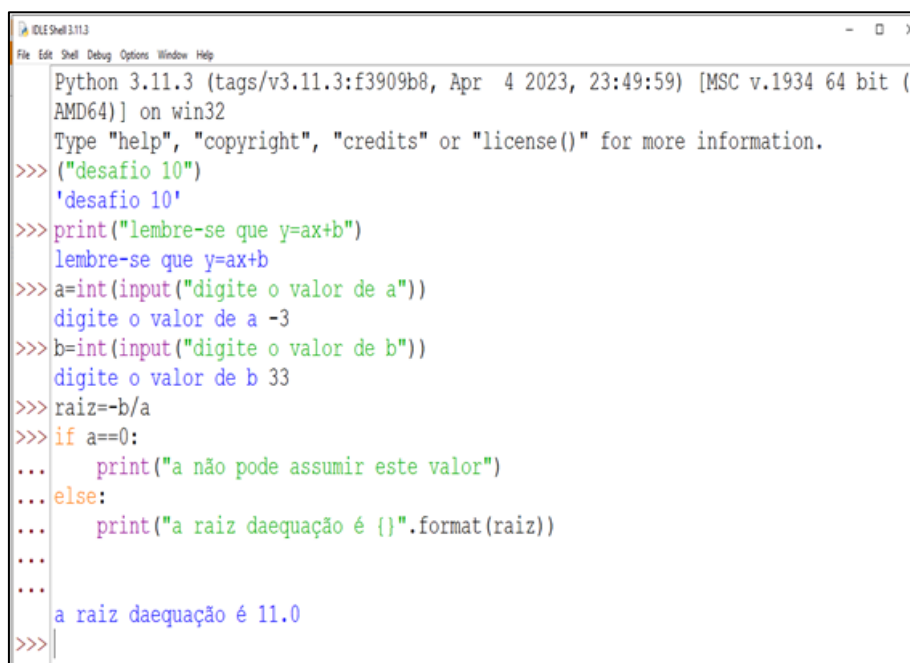
```
Python 3.11.3 (tags/v3.11.3:f3909b8, Apr 4 2023, 23:49:59) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> ("desafio 9")
'desafio 9'
>>> a=int(input("digite o valor de a "))
digite o valor de a -1
>>> a=int(input("digite o valor de a "))
digite o valor de a 1
>>> b=int(input("digite o valor de b "))
digite o valor de b -5
>>> c=int(input("digite o valor de c "))
digite o valor de c 6
>>> D=(b**2-4*a*c)
>>> x1=(-b+D**(1/2))/(2*a)
>>> x2=(-b-D**(1/2))/(2*a)
>>> print("se a é igual a {},b é {} e c é {},então as raios são {} e{}".format(a,b,c,x1,x2))
se a é igual a 1,b é -5 e c é 6,então as raios são 3.0 e2.0
>>>
```

**Fonte:** Elaborado pelo autor (2023).

**Desafio 10:** Crie um programa que resolva uma equação de primeiro grau utilizando os comandos *if* e *else* (condicionais).

**Solução:** veja a resposta na Figura 20.

**Figura 20: Programa usando equação de grau 1 e os comandos *if* e *else***



```
Python 3.11.3 (tags/v3.11.3:f3909b8, Apr  4 2023, 23:49:59) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> ("desafio 10")
'desafio 10'
>>> print("lembre-se que y=ax+b")
lembre-se que y=ax+b
>>> a=int(input("digite o valor de a"))
digite o valor de a -3
>>> b=int(input("digite o valor de b"))
digite o valor de b 33
>>> raiz=-b/a
>>> if a==0:
...     print("a não pode assumir este valor")
... else:
...     print("a raiz daequação é {}".format(raiz))
...
...
a raiz daequação é 11.0
>>>
```

**Fonte:** Elaborado pelo autor (2023).

### 3. CONCEITOS PRELIMINARES SOBRE EXPRESSÕES NUMÉRICAS

Neste capítulo apresentamos conceitos e definições preliminares necessários para o bom desenvolvimento do nosso estudo. As operações básicas serão tratadas no campo dos números reais, As principais referências utilizadas neste capítulo são FLEMMING e GONÇALVES (2006), LEITHOLD (2001), LIPSCHUTZ (1978) e LIMA (2017).

#### 3.1 Conjuntos Numéricos

Nesta seção vamos falar um pouco sobre conjuntos numéricos. O leitor familiarizado com o tema pode omitir a leitura desta seção.

#### 3.2 Números Naturais

Iniciaremos com os números inteiros positivos ou Naturais, os primeiros números conhecidos pela humanidade, de acordo com Fleming e Gonçalves (2006, p.2) que tem como essência os axiomas de Peano.

A lista dos referidos axiomas foi proposta pelo matemático Italiano Giuseppe Peano: (1858-1932).



### **Axiomas de Peano:**

- (A1) Todo número natural tem um único sucessor;
- (A2) Números naturais diferentes têm sucessores diferentes;
- (A3) Existe um único número natural, chamado ‘um’ e representado pelo símbolo ‘1’, que não é sucessor de nenhum outro;
- (A4) Seja  $X$  um conjunto de números naturais. Se 1 pertence a  $X$  e se, além disso, o sucessor de todo elemento de  $X$  ainda pertence a  $X$ , então  $X = \mathbb{N}$ .

Observa-se através dos axiomas de Peano que a ideia de sucessor se refere a operação de adição, ou seja, determinar o sucessor de um número natural  $n$  é adicionar uma unidade a este valor e utilizar a forma  $n + 1$ . Intuitivamente significa tomar o elemento da frente, o que vem logo depois na lista.

O símbolo  $\mathbb{N}$  em alusão a letra inicial da palavra natural, representa o conjunto dos números naturais, isto é,  $\mathbb{N} = \{1, 2, 3, \dots\}$ . O conjunto dos números naturais é infinito. A demonstração desse fato será omitida, pois não faz parte dos objetivos do nosso estudo.

### **3.3 Números Inteiros**

De acordo com os estudiosos Scalabrin, Lopes e Pozebon (2021) o surgimento do conjunto dos números naturais ( $\mathbb{N}$ ) e do conjunto dos números inteiros ( $\mathbb{Z}$ ) não aconteceu de forma cronológica, primeiro os naturais e depois os inteiros, apenas com a formalização do conjunto  $\mathbb{N}$  ficou mais coerente e fácil a formalização do conjunto  $\mathbb{Z}$ , onde os números negativos são simétricos em relação ao número 0.

Portanto os números  $-1, -2, -3, -4, \dots$  são os inteiros negativos. Tomando a união dos números naturais com os inteiros negativos e o zero (0), define-se o conjunto dos números inteiros que se simboliza por  $\mathbb{Z}$ , que provém da palavra ZAHL que significa número em Alemão, isto é  $\mathbb{Z} = \{\dots - 3; -2, -1, 0, +1, +2 + 3, \dots\}$ . O conjunto dos números inteiros é infinito.

### **3.4 Números Racionais**

Como um terceiro degrau na escada dos conjuntos numéricos surgem os números racionais cujo conjunto é simbolizado pela letra  $\mathbb{Q}$ .

**Definição 3.1:** Os números racionais, que podem ser escritos como frações, são números que podem ser escritos na forma  $\frac{m}{n}$ , com  $m$  e  $n$  inteiros e  $n \neq 0$ . Isto é,  $\mathbb{Q} = \left\{ \frac{m}{n}; m, n \in \mathbb{Z} \text{ e } n \neq 0 \right\}$ .

É salutar destacar que o conjunto  $\mathbb{Q}$  é fechado em relação as quatro operações básicas: adição, subtração, multiplicação e divisão. Ou seja, a adição, a subtração, a multiplicação e a divisão entre dois números racionais geram como resultado um número racional. O conjunto dos números racionais é infinito. A demonstração desse fato será omitida, uma vez que não faz parte dos objetivos do nosso estudo.

A seguir, vamos apresentar algumas propriedades do conjunto  $\mathbb{Q}$ .

**Propriedade 1.** (Igualdade de números racionais) Sejam  $\frac{a}{b}$  e  $\frac{c}{d}$  números racionais. Então  $\frac{a}{b} = \frac{c}{d}$  se, e somente se,  $a \cdot d = b \cdot c$ .

**Demonstração:** Considere  $a, b, c, d \in \mathbb{R}$ , tais que  $b \neq 0$  e  $d \neq 0$ .

i)  $\frac{a}{b} = \frac{c}{d} \Rightarrow \frac{a}{b} - \frac{c}{d} = 0 \Rightarrow \frac{ad-bc}{bd} = 0$ . Como  $bd \neq 0$ , então  $ad - bc = 0$ . O que finalmente acarreta em  $ad = bc$ .

$$\text{ii) } a \cdot d = b \cdot c \Rightarrow ad \cdot \frac{1}{bd} = bc \cdot \frac{1}{bd} \Rightarrow a \cdot \frac{1}{b} = c \cdot \frac{1}{d} = \frac{a}{b} = \frac{c}{d} \quad \blacksquare$$

**Propriedade 2.** Todo número racional pode ser representado na forma  $\frac{a}{b}$ , sendo  $b$ , um número natural.

**Demonstração.** Seja  $r$  um número racional, logo  $r = \frac{a}{b}$ , onde  $a, b \in \mathbb{Z}$  e  $b \neq 0$ .

i) Se  $b > 0$ ,  $b$  é natural.

ii) Se  $b < 0$ , então  $-b > 0$  e  $r = \frac{a}{b} = \frac{(-1) \cdot a}{(-1) \cdot b} = \frac{-a}{-b}$ , com  $-b \in \mathbb{N}$ . Aqui usamos o fato  $-b = (-1) \cdot b$ . Deixamos essa parte da demonstração a cargo do leitor. ■

**Propriedade 3:** Dados  $a, b \in \mathbb{Z}$ , se  $a \neq 0$  ou  $b \neq 0$  define-se como Máximo Divisor Comum, e denota-se por  $\text{MDC}(a, b)$  ou  $(a, b)$  ao maior inteiro  $d$  que divide  $a$  ( $d|a$ ) e divide  $b$  ( $d|b$ ).

**Prova:**

De fato, se  $d = (a, b)$ , então existem inteiros  $x$  e  $y$  tais que  $a = xd$  e  $b = yd$  e  $(x, y) = 1$ , portanto  $\frac{a}{b} = \frac{xd}{yd}$ , já que  $ayd = xdb$ . ■

**Propriedade 4.** Todo número racional  $\frac{a}{b} > 0$  pode ser escrito na forma  $\frac{x}{y}$ , sendo  $(x, y) = 1$ , isto é,  $x$  e  $y$  primos entre si. A notação  $(x, y)$  representa o máximo divisor comum entre  $x$  e  $y$ .

### 3.5 Números Irracionais: um pouco da história

Nesta seção vamos tratar um pouco da história sobre os números irracionais, uma vez que as demonstrações relacionadas aos irracionais fogem um pouco do tema do trabalho.

O surgimento dos números irracionais aconteceu desde a época de Pitágoras na Grécia antiga onde era costume associar segmentos geométricos a números e que existiam alguns comprimentos que não podiam ser expressos por um número inteiro. Os Pitagóricos chamavam estes comprimentos de “alogon” significando para eles “não racionais” como também “não podem ser falados ou divulgados”. Percebe-se que o fato da ocorrência desta descoberta deixou muitos integrantes da escola Pitagórica apavorados.

Sendo mais fiel à história, menciona-se o pitagórico Hipasus de Metapontum como o autor da descoberta da existência destes números que hoje denominamos de irracionais. De Hipasus de Metapontum (ou Crotona) aproximadamente contemporâneo de Filolaus, diz ter sido inicialmente um pitagórico, que foi expulso da confraria. Uma estória diz que os pitagóricos lhe erigiram um túmulo, como se estivesse morto, outra que sua apostasia foi punida pela morte num naufrágio, sendo esta a punição por ter declarado tal descoberta (Boyer 1974, p.53).

Portanto os Pitagóricos só conheciam as denominadas grandezas comensuráveis e a partir desta descoberta veio a ideia ou a definição das grandezas incomensuráveis. E foi estudando exatamente a figura geométrica do quadrado e a relação entre lado e diagonal que tal fato aconteceu.

Os segmentos comensuráveis e incomensuráveis são bem tratados e definidos por (LINO,2018) onde relata:

**Definição 3.2:** A medida de um segmento  $AB$ , simbolizado por  $\overline{AB}$ , é um número que deve exprimir quantas vezes o segmento  $AB$  contém o segmento “u”, fixado previamente, tomado como unidade de medida e unitário (“u” = 1).

Dado um inteiro positivo  $n$ , se for possível obter  $n - 1$  pontos intermediários  $A_1, A_2, \dots, A_{n-1}$  no segmento  $\overline{AB}$  tal que os segmentos sejam todos congruentes ao segmento unitário previamente definido, conclui-se que a medida do segmento  $\overline{AB}$  será  $n$ . Se o segmento  $\overline{AB}$  não conter uma quantidade inteira de vezes o segmento unitário, então define-se um outro segmento denominado de subunidade de “u”, digamos  $w = \frac{u}{2}$  para que se possa medir o segmento  $\overline{AB}$  por esta unidade. Agora suponha que o segmento  $\overline{AB}$  contenha três vezes o segmento  $w$ , ou seja,  $\overline{AB} = 3w = \frac{3u}{2}$ . Seja  $w$  um segmento que esteja contido  $n$  vezes em  $u$  e  $m$  vezes em  $\overline{AB}$ , com  $m$  e  $n$  inteiros e positivo. Assim pode-se concluir que  $\overline{AB} = m \cdot w$ ,  $u = n \cdot w$ ,  $\frac{\overline{AB}}{u} = \frac{mw}{nw} = \frac{m}{n}$  e como  $u=1$ , segue que  $\overline{AB} = \frac{m}{n}$ . Neste caso se diz que  $w$  é um submúltiplo comum de  $AB$  e  $u$ , ou ainda que  $AB$  e  $u$  são segmentos comensuráveis. A Negação dos argumentos anteriores define os segmentos incommensuráveis.

De acordo com (CAJORI, 2007) a descoberta da existência de grandezas incommensuráveis é a mais importante contribuição matemática atribuída aos pitagóricos. Eles provaram que a razão entre o lado e a diagonal de um quadrado não pode ser a razão de quaisquer dois números inteiros, sendo chamados segmentos dessa espécie de incommensuráveis e de irracional a razão desses segmentos.

Dentre os números irracionais destaca-se a raiz quadrada de 2 como sendo o primeiro irracional na história. Hoje somos conhecedores de que qualquer raiz quadrada de número que não seja quadrado perfeito é irracional. Além destas raízes o número pi ( $\pi$ ) definido como a razão do comprimento de uma circunferência pelo seu diâmetro e o número de Euler ( $e$ ) ou número de ouro são também famosos e irracionais.

De acordo com Oliveira (2009), as evidências sobre a utilização desse número indicam que a ideia de  $\pi$  já era usada a cerca de 4.000 anos atrás, sendo a relação constante entre a circunferência e o seu diâmetro percebido por muitas civilizações antigas.

Conforme Carvalho (2011), embora a ideia de  $\pi$  tenha sido usada durante a história por muitas civilizações, assim como por muitos matemáticos na modernidade, o motivo, para a escolha da 16ª letra do alfabeto grego ( $\pi$ ) para simbolizar o número irracional tratado, se inicia na antiguidade, onde o famoso matemático grego Arquimedes, no seu tratado “Da medida do Círculo”, refere-se ao comprimento da circunferência pela palavra grega  $\pi\epsilon\rho\acute{\iota}\mu\epsilon\tau\rho\varsigma$  que significava perímetro. Mais tarde, o uso do termo

περίμετρος para indicar o perímetro de um círculo de raio  $R$ , foi denotado de modo abreviado por  $\pi$ , por alguns matemáticos como William Oughtred e Isaac em 1706, O matemático galês William Jones (1675-1749) publicou a obra “A new introduction to Mathematics” na qual usa a letra, designando-a especificamente como a razão entre o perímetro do círculo e seu diâmetro, ao invés de simplesmente perímetro de um círculo. Entretanto, nem todos usavam essa notação. Em 1737 este termo ganhou destaque a partir do seu uso por Leonhard Euler, em sua obra séries infinitas, chamada “*Variae observationes circa*”.

De acordo com MACHADO (2013), na civilização babilônica existem evidências que direcionam a uma compreensão da essência do significado do  $\pi$  por esse povo. Uma dessas evidências foi uma tábua de barro encontrada em 1936 na cidade de Suza, no Iran. Essa tábua traz informações de que a razão entre o perímetro de um hexágono regular e o perímetro de uma circunferência circunscrita a ele era de  $24/25$ .

Por volta do ano 263 o matemático Liu Hui sugeriu que se inscrevesse um polígono de 96 ou 192 lados em uma circunferência e assim estimou o valor de  $\pi$  em 3,14. Depois utilizou um polígono de 3.072 lados e encontrou o valor para  $\pi$  como sendo 3,14159. Outro chinês que se destacou por apresentar outra aproximação para  $\pi$ , foi **Tsu Ch'ung-chih**. No final do século V, este matemático chegou a um valor ainda mais próximo, sendo obtido pela razão  $355/113 = 3,14159292$ , (GRILLI, et al. 2011).

Conforme estudamos anteriormente o surgimento do irracional  $\pi$  é desde a antiguidade, porém o descobrimento do irracional  $e$  aconteceu no período de 1453 a 1789, na idade moderna. Segundo (MAOR, 2008) a existência do número de **EULER** está nitidamente relacionada à utilização da matemática em cálculos envolvendo situações rotineiras e está associado aos juros compostos tratados na Matemática Financeira. Ainda de acordo com (MAOR, 1994, p.26) o valor aproximado do irracional  $e$  estaria relacionado ao estudo detalhado do Montante resultante da aplicação do capital de **\$1** aplicados a uma taxa de 100% a.a. (ao ano).

Como a taxa utilizada é de 100% teríamos a seguinte evolução:

1. capitalização anual, montante  $M = (1 + 1)^1 = 2$
2. capitalização semestral, montante  $M = \left(1 + \frac{1}{2}\right)^2 = \frac{9}{4} = 2,25$
3. capitalização ao trimestre,  $M = \left(1 + \frac{1}{4}\right)^4 = \left(\frac{5}{4}\right)^4 = 2,4414$

Continuando assim esse raciocínio verifica-se que podemos generalizar o cálculo do montante para a capitalização de  $n$  vezes ao ano obtendo a fórmula

$$M = \left(1 + \frac{1}{n}\right)^n. \text{ É possível mostrar que } \lim_{n \rightarrow \infty} \left(1 + \frac{1}{n}\right)^n = e, \text{ no entanto tal}$$

demonstração foge do objetivo do nosso estudo.

O estudo da evolução da capitalização em juros compostos foi apreciado em 1683 pelo matemático Suíço Jacob Bernoulli (1654-1705) que provou que o valor do número de EULER estava no intervalo (2,3). Coube ao matemático alemão Gottfried Leibniz a primeira aparição propriamente dita do número  $e$ , em 1690. Muitos outros matemáticos trataram e discutiram a existência do número irracional  $e$ , entre eles, um dos mais importantes matemáticos da história, o grande Leonhard Euler.

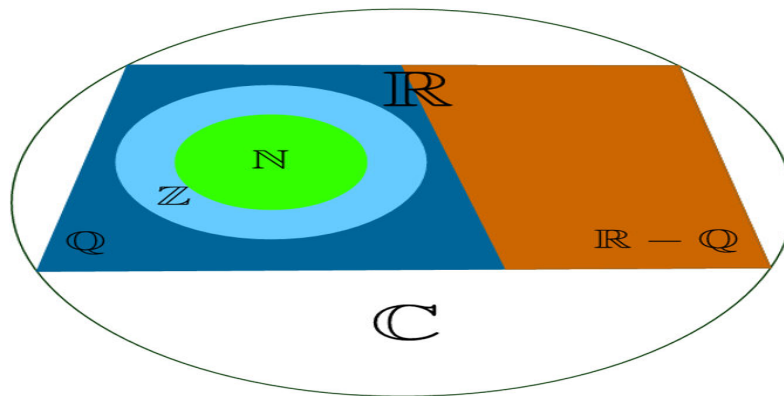
De acordo com (BOYER,2010 p.305) foi o matemático EULER que divulgou e padronizou a utilização do número  $e$ , através de uma exposição manuscrita dos resultados de experiências sobre disparo de canhões, em 1727 ou 1728, data não precisa, EULER utilizou o símbolo  $e$  mais de uma dúzia de vezes para representar a base do sistema de logaritmos naturais. Em uma carta endereçada a Goldbach em 1731, o matemático EULER utilizou o  $e$  na seguinte expressão: “aquele número cujo logaritmo hiperbólico é igual a 1” (BOYER, 2010, p.305).

Uma das obras mais conhecidas e importantes do matemático EULER é *Introductio in Analysin Infinitorum*, publicada inicialmente em 1748 onde o matemático utiliza a letra  $e$ , apresentando este número com 23 casas decimais ( $e = 2,71828182845904523536028$ ), utilizando a série infinita  $\sum \frac{1}{n!}$ .

### 3.6 Números Reais

Para finalizar a lista de conjuntos numéricos essenciais a este trabalho apresento o conjunto dos números Reais, denotado por  $\mathbb{R}$ , e que na verdade é dado pela união dos conjuntos  $\mathbb{Q}$  e  $\mathbb{Q}'$ . De forma rigorosa pode-se afirmar que:  $\mathbb{R} = \mathbb{Q} \cup \mathbb{Q}'$  e  $\mathbb{Q}' = \mathbb{R} - \mathbb{Q}$ . Assim dado um número qualquer, digamos  $n$ , este será racional ou irracional, ou seja,  $n \in \mathbb{Q}$  ou  $n \in \mathbb{Q}'$ .

**Figura 21: Conjuntos numéricos**



Fonte: <https://brasilecola.uol.com.br/o-que-e/matematica/o-que-sao-conjuntos-numericos.htm> em 13/09/2023

**Definição 3.3** (Adição e Multiplicação). No conjunto dos números reais  $\mathbb{R}$  estão definidas duas operações, as quais são a **adição** e a **multiplicação**. A adição associa a cada par de elementos  $x, y \in \mathbb{R}$  a sua soma  $x + y \in \mathbb{R}$ . A multiplicação faz corresponder a cada par de elementos o seu produto  $x \cdot y \in \mathbb{R}$ . Essas duas operações satisfazem os seguintes axiomas, dos quais seguem todas as regras de manipulação com os números reais.

#### I. Fechamento

Se  $a, b \in \mathbb{R}$ , então existe um e somente um número real denotado por  $a + b$ , chamado soma, e existe um e somente um número real, denotado por  $ab$  (ou  $a \cdot b$ ), chamado produto.

#### II. Comutatividade

Se  $a, b \in \mathbb{R}$ , então  $a + b = b + a$  e  $a \cdot b = b \cdot a$ .

#### III. Associatividade

Se  $a, b, c \in \mathbb{R}$ , então  $a + (b + c) = (a + b) + c$  e  $(ab) \cdot c = a(b \cdot c)$ .

#### IV. Distributividade

Se  $a, b, c \in \mathbb{R}$ , então  $a \cdot (b + c) = ab + ac$ .

#### V. Existência de elementos neutros

Existem 0 e 1 números reais tais que  $a + 0 = a$  e  $a \cdot 1 = a$ , para qualquer  $a \in \mathbb{R}$ .

#### VI. Existência de simétricos

Todo  $a \in \mathbb{R}$  tem um simétrico ou oposto denotado por  $-a$ , tal que  $a + (-a) = 0$ .

#### VII. Existência de Inversos

Todo  $a \in \mathbb{R}$ ,  $a \neq 0$ , tem um inverso denotado por  $1/a = \frac{1}{a} = a^{-1}$ , tal que  $a \cdot \frac{1}{a} = 1$ .

### VIII. Subtração

Se  $a, b \in \mathbb{R}$ , a diferença entre  $a$  e  $b$ , denotada por  $a - b$ , é definida por  $a - b = a + (-b)$ . Usou-se o item VI.

### IX. Divisão

Se  $a, b \in \mathbb{R}$  e  $b \neq 0$ , o quociente de  $a$  e  $b$  é definido por  $\frac{a}{b} = a \cdot \frac{1}{b}$ .

**Definição 3.4** (Subtração e Divisão). Devido a existência do simétrico, podemos definir a diferença entre  $x$  e  $y$  dada por  $x - y = x + (-y)$ . A operação que associa o par  $x, y \in \mathbb{R}$  ao número real  $x - y$  é chamada de subtração. Se  $y \neq 0$ , o produto  $x \cdot y^{-1} = \frac{x}{y}$  é chamado quociente de  $x$  por  $y$ . A operação que associa o par  $x, y \in \mathbb{R}$  ao número real  $\frac{x}{y}$  é chamada de divisão.

**Definição 3.5** (Potenciação). A operação potenciação é definida a partir da multiplicação de elementos ou fatores iguais. Dados  $a$  e  $b$  números inteiros e não conjuntamente nulos, define-se  $a^b$  como sendo  $a \cdot a \cdot a \cdot \dots \cdot a$  uma quantidade  $b$  de vezes, ou seja,  $a^b = a \cdot a \cdot a \cdot \dots \cdot a$ .

Denomina-se base o número  $a$ , expoente o número  $b$  e  $a^b$  será a potência ou o resultado da operação. Define-se também  $a^0 = 1$  e  $a^1 = a$ .

Algumas propriedades da potenciação:

$$P1: a^m \cdot a^n = a^{m+n}$$

$$P2: a^m : a^n = a^{m-n}$$

$$P3: (a^m)^\eta = a^{m\eta}$$

$$P4: (ab)^m = a^m \cdot b^m$$

$$P5: (a : b)^m = a^m : b^m$$

$$P6: a^{-n} = \frac{1}{a^n}$$

**Definição 3.6** (Radiciação): A radiciação é a operação inversa da potenciação. De modo geral pode-se escrever  $\sqrt[n]{a} = b$  se, e somente se,  $b^n = a$ , para  $n$  natural maior que ou igual a 1.

Algumas propriedades da radiciação:



$$P1: \quad \sqrt[n]{a^m} = a^{\frac{m}{n}}, \text{ em particular: } \sqrt[n]{a^n} = a^{\frac{n}{n}} = a^1 = a$$

$$P2: \quad \sqrt[n]{ab} = \sqrt[n]{a} \cdot \sqrt[n]{b}$$

$$P3: \quad \sqrt[n]{\frac{a}{b}} = \frac{\sqrt[n]{a}}{\sqrt[n]{b}}$$

$$P4: \quad \sqrt[n]{\sqrt[m]{a}} = \sqrt[nm]{a}$$

$$P5: \quad (\sqrt[n]{a})^k = \sqrt[n]{a^k}$$

Diante das definições tratadas nesta seção chega-se à conclusão de que se pode escrever algumas operações utilizando como ponto inicial a operação adição.

$$1) a - b = a + (-b)$$

$$2) a \times b = (b + \dots + b) \quad , a \text{ vezes.}$$

$$3) a^b = a \cdots a, b \text{ vezes.}$$

### 3.7 Expressões Numéricas

Iniciaremos esta seção com uma situação-problema do cotidiano que requer o uso de expressão numérica.

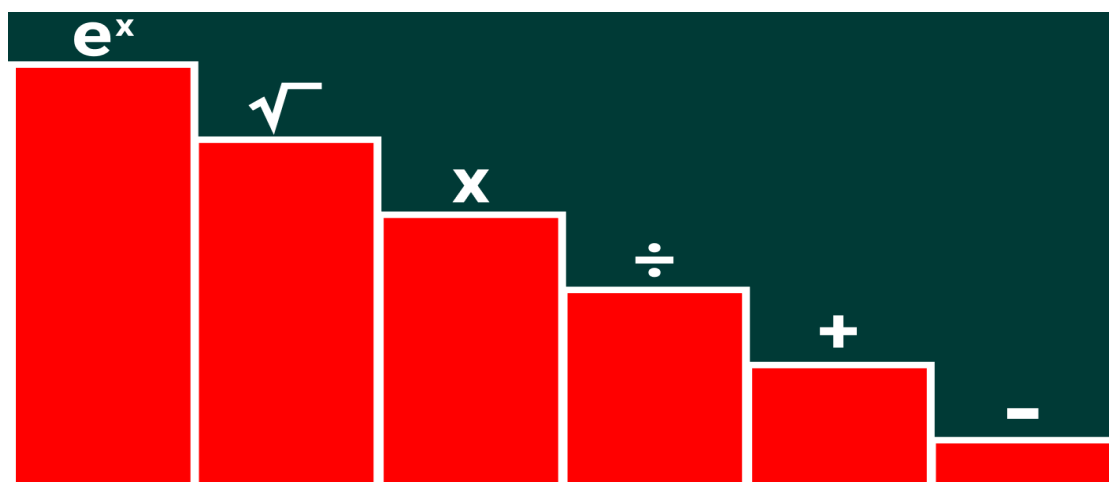
Qual foi a mesada do menino Jean, se sua mãe lhe deu uma nota de R\$ 10,00 e seu pai lhe deu 4 notas de R\$ 5,00? (RIBEIRO, 2010, p.53). Esse tipo de situação real, cotidiana considerada por muitos docentes como problema motivador costuma ser resolvido através das expressões numéricas. Tal situação-problema pode ser resolvida através da expressão matemática  $10 + 4 \times 5$  cujo resultado será R\$ 30,00. Esta expressão numérica é composta apenas de duas operações matemáticas básicas tratadas no Ensino Fundamental e de início resolve-se a multiplicação e depois a operação adição, que trataremos mais adiante.

De maneira explícita, pode-se dizer que a expressão numérica é uma forma de expressar, traduzir ou descrever matematicamente uma situação (RAMOS, 2002, p.21). Neste caso, o resultado da expressão numérica é a resposta a uma pergunta que envolve o problema em questão.

Definimos expressões numéricas como sequências de uma ou mais operações com números que devem ser resolvidas ou realizadas respeitando uma ordem de precedência, regras que definem a ordem que as operações serão realizadas, e que tem um único resultado. Uma expressão numérica é composta de números, operações e os sinais de associação conhecidos como parênteses, colchetes e chaves.

Quanto às operações matemáticas presentes em uma expressão numérica deve-se resolver ou operacionalizar da seguinte maneira: em primeiro lugar as radiciações e potenciações, depois as multiplicações e divisões e por último as adições e subtrações. Siga o sentido da esquerda para a direita, como se a expressão numérica fosse um pedaço de reta e o leitor a pessoa a resolver esta expressão que caminha pausadamente do sentido oeste para o sentido leste.

**Figura 22: Ordem de precedência**



Fonte: matematicabasica.net. Acesso em 18/06/2023.

Um fato importante que se deve mencionar é a questão do aparecimento concomitante de parêntesis ( ), colchetes [ ] e chaves { }. Alguns livros didáticos relatam que a sequência correta para resolução da expressão numérica é a seguinte: resolve-se logo o que está entre parêntesis, depois o que está entre os colchetes e por último o que está inserido nas chaves.

$$7 - \{2 + [5 - (2^3 + 1)]\} - 7.$$

No entanto, perceba que este mesmo exemplo pode ser escrito alternando os parêntesis, colchetes e chaves e o resultado será o mesmo. Veja a seguir a nova situação:

$$7 - (2 + [5 - \{2^3 + 1\}]) - 7.$$

As duas expressões são iguais numericamente, de modo que o correto é resolver da parte mais interna a parte mais externa da expressão numérica, ou seja, resolve-se de dentro para fora.

O entendimento e a aprendizagem na resolução das expressões numéricas são bem mais simples quando se ensina com foco apenas no conjunto dos números naturais ( $\mathbb{N}$ ), pois não surgem os números negativos. A dificuldade maior é bem mais nítida quando eles aparecem. Dessa forma, acreditamos que seja interessante trabalhar primeiramente no contexto dos números naturais. Uma vez que os alunos estejam familiarizados, seguimos aumentando o nível para as expressões numéricas, envolvendo os números inteiros e racionais.

É comum os alunos se confundirem com os sinais (teoria dos números inteiros) que envolvem as regras matemáticas. Em razão disso ressaltamos a importância do manuseio das regras para encontrar a solução. Para Alves (2012) essa dificuldade de se trabalhar com os sinais e as regras para solucionar pequenos cálculos envolvendo os números inteiros acontece no ensino fundamental e também no ensino médio. Enumeramos a seguir uma pequena amostra de expressões matemáticas que são denominadas de expressões numéricas contendo parêntesis, colchetes e chaves.

$$\text{i) } (2 + 4) \times 3 + 6 \div 2 + 32 - ([10 - 2] + 2 \times 3)$$

$$\text{ii) } 4 \times 5 - 6 \times 2 + 4 \div 2 - (4 - 3) + [5 \times 2]$$

$$\text{iii) } (20 \div 10) - 3 \times 20 - [6 \times 3] - ([7 - 5] \times 2)$$

$$\text{iv) } 7 - \{2 + [5 - (2^3 + 1)]\} - 7.$$

$$\text{v) } 4 - \sqrt{4} + 4 - 4 - 4 \times 4$$

Em Matemática podemos utilizar modelos para traduzir, descrever e expressar situações do cotidiano, as quais podem ser representadas por expressões que envolvam uma ou mais operações matemáticas e que podem ou não estar agrupadas por sinais de associação, resultando em uma quantidade numérica (FREITAS, 2014).

Os enunciados de problemas de expressões numéricas trazem em si propostas de pensamento em estruturas de operações, que se diversificam na redação do problema, e envolvem passos que podem levar à construção correta da escrita da expressão na forma algébrica (forma simbólica da expressão numérica), bem como de sua resolução.

As expressões numéricas, além de aglutinar as operações conhecidas pelos estudantes em problemas, iniciam um caráter simbólico no qual as operações são escritas de forma algébrica e resolvidas como tal.

## **4 SEQUÊNCIA DIDÁTICA COM USO DO PYTHON PARA EXPRESSÕES NUMÉRICAS**

Neste capítulo apresentamos uma sequência didática para trabalhar expressões numéricas com o uso da linguagem de programação Python.

A expressão sequência didática é utilizada desde a década de 1980, nas pesquisas em Didática da Matemática que incluem pesquisa experimental. É amplamente empregada pelos pesquisadores e estudiosos da Didática da Matemática francesa. Por estar relacionado a uma Sequência de experimentos, é bastante aplicada nos trabalhos baseados na Engenharia Didática, citado por Artigue (1988 apud PAIS, 2001) assim a define:

Sequência Didática é um conjunto de aulas planejadas e analisadas previamente com a finalidade de observar situações de aprendizagem, envolvendo os conceitos previstos na pesquisa didática [...] tal como acontece na execução de todo projeto, é preciso estar atento durante as sessões ao maior número de informações que podem contribuir no desvelamento do fenômeno investigado.

E ainda de acordo com Zabala (1998, p.18) sequência didática é “um conjunto de atividades ordenadas, estruturadas e articuladas para a realização de certos objetivos educacionais, que têm um princípio e um fim conhecido tanto pelos professores como pelos alunos”. Assim conclui-se que a definição de sequência didática se refere à organização de uma sequência de aulas, geralmente planejadas para pesquisas relacionadas à Didática, podendo ser também uma produção para o próprio ensino.

### **4.1 Sequência didática proposta**

Nesta seção foi proposta uma sequência didática para a resolução de expressões numéricas voltada para o professor de Matemática que esteja trabalhando o referido conteúdo, estudado no Ensino Fundamental, mas com o uso da linguagem de programação Python.

#### **Detalhamento da Sequência Didática**

**1. Unidade Temática (BNCC):** Números

**2. Modalidade/Nível de Ensino:** 8º e 9º ano do ensino fundamental.

**3. Objeto de Conhecimento (BNCC):** Operações de adição, subtração, multiplicação e divisão.

**4. Habilidades (BNCC): EF07MA04** – Resolver e elaborar problemas que envolvam operações.

**5. Objetivos/Expectativas de Aprendizagem:** trabalhar o raciocínio lógico do aluno; desenvolver a capacidade de efetuar o cálculo das quatro operações básicas, num primeiro momento sem o uso de calculadoras, através do estudo de expressões numéricas; desenvolver a agilidade e a competência para calcular; introduzir o aluno ao mundo da programação em Python.

**6. Estratégia de ensino e materiais utilizados:** aulas expositivas, atividade individual, quadro e pincel, caderno, lápis, borracha, caneta, livro didático, projetor portátil, computador.

**7. Duração das atividades:** 4 aulas de 50 minutos podendo haver um ajuste conforme necessidade da turma.

#### Quadro 6-sequência didática

<b><u>Sequência didática proposta</u></b>		
<b><u>Etapas</u></b>	<b><u>Descrição</u></b>	<b><u>Tempo</u></b>
<b>1<sup>a</sup></b>	<b>Problema gerador.</b>	<b>50 min</b>
<b>2<sup>a</sup></b>	<b>Resolução analítica de algumas expressões numéricas.</b>	<b>50 min</b>
<b>3<sup>a</sup></b>	<b>Resolução das expressões numéricas através da linguagem de programação Python.</b>	<b>50 min</b>

**Fonte:** Elaborado pelo autor (2023).

A sequência didática está estruturada em três etapas. Na primeira etapa foi proposto o uso de um problema gerador para introduzir o tema de uma maneira que o aluno seja membro ativo na construção da aprendizagem. Nesta etapa orienta-se que o professor não explique o conteúdo, mas apenas apresente o problema gerador e divida a turma em grupos, de no máximo 5 alunos, para discussão e resolução do problema. Em seguida, cada grupo deve socializar a sua resposta com o restante da turma. A partir da discussão iniciada pelos alunos, sugere-se que o professor faça uma reflexão geral e explique o conteúdo de maneira leve.

Na segunda etapa da sequência foi proposta a resolução de algumas expressões numéricas, as quais são tratadas do ponto de vista matemático e com o intuito de fixação do conteúdo. Nesta etapa as expressões numéricas podem ser apresentadas de

forma direta, isto é, não necessariamente precisam ser problemas interpretativos. Para que o professor possa trabalhar uma quantidade maior de exercícios, uma vez que o objetivo aqui é a fixação do conteúdo. Sugerimos que o professor resolva uma ou duas questões e, em seguida, aplique algumas atividades para os alunos resolverem individualmente na sala de aula.

Na terceira, e última etapa, foi proposta a linguagem de programação *Python* para resolver as mesmas expressões numéricas propostas na segunda etapa. Acredita-se que o uso do Python irá diversificar a aula e apresentar novas ferramentas para tornar as aulas mais dinâmicas, além de introduzi-los no mundo da linguagem de programação. A ideia é que o aluno atue como um professor que está ensinando o seu “aluno” (o computador) a resolver uma expressão numérica através da ferramenta Python.

Dependendo do nível de conhecimento da turma, sobre linguagem de programação em Python, pode ser necessária uma aula para introduzir tal conteúdo. Deixamos essa etapa a critério do professor. Apresentamos na Seção 2.4 alguns desafios para que o aluno fique familiarizado com tal linguagem de programação.

Pode-se observar que a sequência didática proposta nesta seção tem um caráter generalista, de forma que possa ser aplicada de várias maneiras distintas. Para isso, basta mudar o problema gerador e os exercícios propostos. Além disso, a estrutura da sequência aqui proposta pode ser utilizada para trabalhar outros conteúdos matemáticos. Por exemplo, solução de equações do primeiro e segundo graus, entre outros.

Na seção a seguir apresentou-se um caso particular da sequência didática proposta, onde foi apresentado um problema gerador específico e alguns exercícios envolvendo o objeto de estudo desta pesquisa.

## **4.2 Um caso particular para a sequência didática proposta**

A seguir apresentou-se casos particulares do problema gerador e atividades propostas, envolvendo expressões numéricas, para aplicar a sequência didática sugerida na Seção 4.1.

De acordo com Vygotsky (1991) a atuação do professor é de suma importância já que ele exerce o papel de mediador da aprendizagem do aluno. Certamente é muito importante para o aluno a qualidade de mediação exercida pelo professor, pois desse processo dependerão os avanços e as conquistas do aluno em relação à aprendizagem na escola. Assim todo e qualquer mecanismo didático e eficaz que estimule

o interesse ou a motivação do aluno pelo conteúdo apresentado é essencial ao processo ensino aprendizagem. Por isso resolvemos utilizar uma situação-problema que traz um ambiente próprio, próximo, corriqueiro ou familiar do aluno.

### **Primeira Etapa: Situação-problema (problema gerador)**

A cidade de Quixeramobim, no interior do Ceará, produziu o maior queijo coalho artesanal do mundo, pesando 1.480 quilos. O recorde foi apresentado pela prefeitura do município na noite de 11 de agosto de 2022, durante a 4ª edição da Copa Leite, evento que reúne pecuaristas e produtores de laticínios.

**Figura 23: Imagem do maior queijo do mundo**



**Fonte:** <https://g1.globo.com/ce/ceara/noticia/2022/11/12/cidade-do-ceara-produz-o-maior-queijo-coalho-do-mundo-pesando-1480-kg.ghtml>

Na fabricação do queijo artesanal são utilizados basicamente os seguintes ingredientes: leite, sal e coagulante. Para cada 1 kg de queijo são necessários 10 litros de leite, 100 gramas de sal e 1 colher de sopa (15 ml) de coagulante ou coalho industrializado. Sabendo que atualmente os preços dos ingredientes são:

O preço médio de 1 ml do coagulante é de R\$ 0,16.

O preço médio de 1 litro de leite é de R\$ 2,00.

O preço médio de 100 gramas de sal é de R\$ 0,13.

Determine o custo total da produção do maior queijo artesanal do mundo em Quixeramobim/CE ocorrida em 2020.

Para isto, precisamos resolver a expressão numérica a seguir.

$$\begin{aligned} 1480 \times \{(10 \times 2) + 0,13 + (15 \times 0,16)\} = \\ 1480 \times \{20 + 0,13 + 2,40\} = \\ 1480 \times 22,53 = \\ \text{R\$ } 33.344,40. \end{aligned}$$

## **Segunda Etapa – Resolução de expressões numéricas utilizando a linguagem matemática**

### **Momento Inicial**

Inicialmente proponha a resolução das expressões a seguir que são consideradas simples, para que o aluno se sinta à vontade e que crie autoestima para a resolução de expressões mais complexas.

**Atividade 1:**  $(4 \times 5) - 2$ .

**Solução:** Primeiro a multiplicação  $(4 \times 5)$  obtendo o resultado 20. Em seguida diante da nova sentença matemática fechada,  $20 - 2$ , efetuar a subtração encontrando o número 18 como resultado.

**Atividade 2:**  $4 \times (5 - 2)$ .

**Solução:** Lembre-se que primeiro é preciso resolver a expressão dentro dos parênteses, neste caso o professor efetuará a subtração  $(5 - 2)$  encontrando o número 3 como resultado. Em seguida efetuar o cálculo,  $4 \times 3$ , encontrando o valor final igual a 12, fruto da multiplicação.

### **Momento Intermediário**

Neste momento o professor deve aumentar o nível de dificuldade das expressões e propor a resolução das expressões numéricas listadas a seguir.

**Atividade 3:**  $2 + (3 - 2) - (2 - 3) - 1$ .



**Solução:** O professor deve resolver ou eliminar os dois parêntesis presentes na expressão, um de cada vez, com tranquilidade ficando com a nova sentença:  $2 + 1 - (-1) - 1$ . Agora, o professor deve trabalhar o resultado de  $-(-1)$  mostrando aos alunos que o primeiro sinal de menos representa em matemática o “oposto de”, portanto  $-(-1) = +1 = 1$ . E assim surge a sentença:  $2 + 1 + 1 - 1$ . Finalmente, o professor resolve a sentença seguindo as operações da esquerda para a direita, encontrando o resultado 3.

**Atividade 4:**  $4 - \{3 + [2^3 - 5 + 1 - 2] - 1\} + 7$ .

**Solução:** O professor deve eliminar os colchetes e depois as chaves, da região interna para a região externa. Eliminando os colchetes:  $[2^3 - 5 + 1 - 2]$  ele resolverá de imediato a potenciação e depois as demais operações ficando no estágio:  $[8 - 5 + 1 - 2] = 2$ . Resultando na sentença  $4 - \{3 + 2 - 1\} + 7$ . Elimina-se as chaves cujo resultado é igual a 4, obtendo uma nova sentença:  $4 - 4 + 7 = 0 + 7 = 7$ .

**Atividade 5:**  $(3 - 6) + \{3 + (2 \times 0) - [2^3 - 3^2] - 1\}$ .

**Solução:** O professor pode resolver logo as potenciações presentes no interior dos colchetes  $[2^3 - 3^2]$  obtendo  $8 - 9 = -1$ . Portanto, a sentença passa a ser:  $(3 - 6) + \{3 + (2 \times 0) - (-1) - 1\}$ . Agora, o professor pode eliminar todos os parêntesis presentes na expressão e também calcular o valor de  $-(-1) = 1$ . A nova sentença passa a ser a seguinte:  $-3 + \{3 + 0 + 1 - 1\}$ . Finalmente, eliminando as chaves, o professor e os alunos encontram o resultado igual a 3. Eis a nova sentença,  $-3 + 3$ , cujo resultado é igual a 0.

**Atividade 6:**  $3^0 - \{5 - [(10^1 - 3^2) + 8]\} - (-2)$ .

**Solução:** O professor calcula a potência  $3^0$ , elimina os parêntesis, calculando as potências internas e faz o cálculo já bem conhecido pela turma de  $-(-2) = 2$ , resultando na nova sentença:  $1 - \{5 - [1 + 8]\} + 2$ . Agora eliminamos os colchetes e por último as chaves resultando a nova sentença matemática:  $1 - \{5 - 9\} + 2 = 1 - \{-4\} + 2 = 1 + 4 + 2 = 7$

**Atividade 7:**  $\{(4 + 2^0) + [(-2 - 3) - (12 \times 100)] + (10^2 \div 5^2)\} - 5$ .

**Solução:** O professor deve inicialmente eliminar os quatro pares de parêntesis presentes na expressão numérica, calculando as potências, a soma, a multiplicação e a divisão, restando a nova sentença:  $\{5 + [-5 - 1200] + 4\} - 5$ . Eliminando os colchetes o professor

ficará diante da seguinte sentença:  $\{5 - 1205 + 4\} - 5$ . Por fim eliminando as chaves o professor chegará ao resultado igual a  $-1201$ .

**Terceira Etapa: Resolução das expressões numéricas, propostas na 2ª etapa, utilizando a linguagem de programação Python.**

Nesta etapa, o professor pode utilizar o modo interativo do Python e executar o IDLE (3.11.64 bit), já devidamente instalado em seu computador ou simplesmente utilizar o modo online, sem precisar sobrecarregar a memória de seu instrumento computacional.

**Atividade 1:**  $(4 \times 5) - 2$ .

**Solução:** Observe no quadro a seguir o comparativo entre a linguagem matemática e a linguagem Python.

**Quadro 7-comparativo entre linguagens**

Expressão numérica	Linguagem Python
$(4 \times 5) - 2$	$(4 * 5) - 2$

**Fonte:** Elaborado pelo autor (2023).

**Figura 24: Imagem do resultado da expressão no Python**

```

Python 3.11.3 (tags/v3.11.3:f3909b8, Apr 4 2023, 23:49:59) [MS
C v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more inf
ormation.
>>> ('atividade 1: Resolva a expressão (4x5)-2 .')
'atividade 1: Resolva a expressão (4x5)-2 .'
>>>
>>> (4*5)-2
18
>>>

```

**Fonte:** Elaborado pelo autor (2023).

**Atividade 2:**  $4 \times (5 - 2)$ .

**Solução:** Veja o quadro 8:

**Quadro 8-comparativo entre linguagens**

Expressão numérica	Linguagem Python
$4 \times (5 - 2)$	$4 * (5 - 2)$

**Fonte:** Elaborado pelo autor (2023).

**Figura 25: Imagem do resultado da expressão no Python**

```

Python 3.11.3 (tags/v3.11.3:f3909b8, Apr 4 2023, 23:49:59) [MS
C v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more inf
ormation.
>>> ('Atividade 2: Resolva a expressão 4x(5-2) ')
'Atividade 2: Resolva a expressão 4x(5-2) '
>>>
>>> 4*(5-2)
12
>>>

```

**Fonte:** Elaborado pelo autor (2023).

Na primeira expressão numérica, deve-se efetuar a multiplicação  $4 \times 5$ , que se encontra dentro do parêntesis e em seguida efetua-se a subtração  $20 - 2$ , encontrando o resultado igual a 18. Na segunda expressão, calcula-se de imediato a subtração  $(5 - 2)$  e em seguida opera-se com  $4 \times 3$ , resultando em 12. Nestes dois casos, a posição do parêntesis é de suma importância, pois dá resultados desiguais e, portanto, tem-se expressões numéricas diferentes.

Observe que se não utilizássemos parêntesis, teríamos o seguinte resultado:

**Quadro 9-comparativo entre linguagens**

Expressão numérica	Linguagem Python
$4 \times 5 - 2$	$4 * 5 - 2$

**Fonte:** Elaborado pelo autor (2023).

**Figura 26: Imagem do resultado da expressão no Python**



```

Python 3.11.3 (tags/v3.11.3:f3909b8, Apr 4 2023, 23:49:59) [MS
C v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more inf
ormation.
>>> ('Comentário: resolva a expressão 4x5 -2 ')
'Comentário: resolva a expressão 4x5 -2 '
>>>
>>> 4*5-2
18
>>>
  
```

**Fonte:** Elaborado pelo autor (2023).

O resultado foi o mesmo encontrado na expressão  $(4 \times 5) - 2 = 18$ . Isso demonstra que a operação multiplicação tem preferência em relação à operação subtração.

**Atividade 3:**  $2 + (3 - 2) - (2 - 3) - 1$ .

**Solução:** Veja o quadro 10:

**Quadro 10-comparativo entre linguagens**

Expressão numérica	Linguagem Python
$2 + (3 - 2) - (2 - 3) - 1$	$2 + (3 - 2) - (2 - 3) - 1$

Fonte: Elaborado pelo autor (2023).

**Figura 27: Imagem do resultado da expressão no Python**

```

Python 3.11.3 (tags/v3.11.3:f3909b8, Apr  4 2023, 23:49:59) [MS
C v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more inf
ormation.
>>>
>>> 2+(3-2)-(2-3)-1
3
>>>

```

Fonte: Elaborado pelo autor (2023).

**Atividade 4:**  $4 - \{3 + [2^3 - 5 + 1 - 2] - 1\} + 7$ .

**Solução:** Veja o quadro 11:

**Quadro 11-comparativo entre linguagens**

Expressão numérica	Linguagem Python
$4 - \{3 + [2^3 - 5 + 1 - 2] - 1\} + 7$ .	$4 - ((2 ** 3) - 5 + 1 - 2) - 1) + 7$

Fonte: Elaborado pelo autor (2023).

**Figura 28: Imagem do resultado da expressão no Python**

The screenshot shows a Python IDLE Shell window titled 'IDLE Shell 3.11.3'. The window contains the following text:

```
Python 3.11.3 (tags/v3.11.3:f3909b8, Apr 4 2023, 23:49:59) [MS
C v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more inf
ormation.
>>> 4-(3+((2**3)-5+1-2)-1))+7
7
>>>
```

**Fonte:** Elaborado pelo autor (2023).

Na expressão anterior, verifica-se que existem quatro pares de parêntesis ou exatamente oito parêntesis, que devem ser utilizados na linguagem Python nessa exata quantidade para que não ocorra erro de sintaxe ou erro de resultado. Caso ocorra erro no resultado, deve-se contar os parêntesis e verificar se a quantidade de parêntesis abertos dá exatamente igual a quantidade de parêntesis fechados, ou seja, a quantidade de parêntesis do tipo (“**aberto à direita**” deve ser igual a quantidade de parêntesis do tipo) “**aberto à esquerda**”.

**Atividade 5:**  $(3 - 6) + \{3 + (2 \times 0) - [2^3 - 3^2] - 1\}$ .

**Solução:** Veja o quadro 12:

**Quadro 12-comparativo entre linguagens**

Expressão numérica	Linguagem Python
$(3 - 6) + \{3 + (2 \times 0) - [2^3 - 3^2] - 1\}$ .	$(3 - 6) + (3 + (2 * 0) - ((2 ** 3) - (3 ** 2)) - 1)$

**Fonte:** Elaborado pelo autor (2023).

**Figura 29: imagem do resultado da expressão no Python**

```

Python 3.11.3 (tags/v3.11.3:f3909b8, Apr 4 2023, 23:49:59) [MS
C v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more inf
ormation.
>>>
>>> (3-6)+(3+(2*0)-((2**3)-(3**2))-1)
0
>>>

```

**Fonte:** Elaborado pelo autor (2023).

**Atividade 6:**  $3^0 - \{5 - [(10^1 - 3^2) + 8]\} - (-2)$ .

**Solução:** Veja o quadro 13:

**Quadro 13-comparativo entre linguagens**

Expressão numérica	Linguagem Python
$3^0 - \{5 - [(10^1 - 3^2) + 8]\} - (-2)$	$(3 ** 0) - (5 - ((10 ** 1) - (3 ** 2) + 8)) - (-2)$

**Fonte:** Elaborado pelo autor (2023).

**Figura 30: Imagem do resultado da expressão no Python**

```

Python 3.11.3 (tags/v3.11.3:f3909b8, Apr 4 2023, 23:49:59) [MS
C v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more inf
ormation.
>>>
>>> (3**0)-(5-((10**1)-(3**2)+8))-(-2)
7
>>>

```

**Fonte:** Elaborado pelo autor (2023).

**Atividade 7:**  $\{(4 + 2^0) + [(-2 - 3) - (12 \times 100)] + (10^2 \div 5^2)\} - 5$ .

**Solução:** Veja o quadro 14:

**Quadro 14-comparativo entre linguagens**

Expressão numérica	Linguagem Python
$\{(4 + 2^0) + [(-2 - 3) - (12 \times 100)] + (10^2 \div 5^2)\} - 5$	$(4 + (2 ** 0)) + ((-2 - 3) - (12 * 100) + \frac{10 ** 2}{5 ** 2}) - 5$

**Fonte:** Elaborado pelo autor (2023).

**Figura 31: imagem do resultado da expressão no Python**

```

Python 3.11.3 (tags/v3.11.3:f3909b8, Apr 4 2023, 23:49:59) [MS
C v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more inf
ormation.
>>>
>>> (5-1205+4)-5
-1201
>>> (5+(-5-1200)+4)-5
-1201
>>> (4+(2**0))+((-2-3)-(12*100)+(10**2)/(5**2))-5
-1201.0
>>>
  
```

**Fonte:** Elaborado pelo autor (2023).

### 4.3 Resultados da aplicação da Sequência Didática em sala de aula

Nesta seção apresentou-se os resultados obtidos a partir da aplicação da Sequência Didática proposta na Seção 4.2 em sala de aula. Como universo da pesquisa utilizou-se 10 alunos da turma de 9º ano do CEJA (Centro Educacional de Jovens e Adultos) - Alfredo Almeida Machado, da cidade de Quixeramobim no Ceará, com estudantes na faixa etária de 16 e 30 anos. Com essa aplicação, pretendemos, investigar



se o uso da linguagem de programação em *Python* pode ser potencializadora do processo de aprendizagem em matemática.

A atividade foi organizada para desenvolver conteúdos de expressão numérica com o auxílio de Programação em *Python*. Como não são solicitados conhecimentos prévios para os participantes, os conhecimentos básicos da área de linguagem de programação e o conteúdo de expressão numérica, precisaram ser explicados, exemplificados e exercitados durante a oficina. Foi utilizada uma aula de 50 minutos para introduzir a linguagem de programação Python.

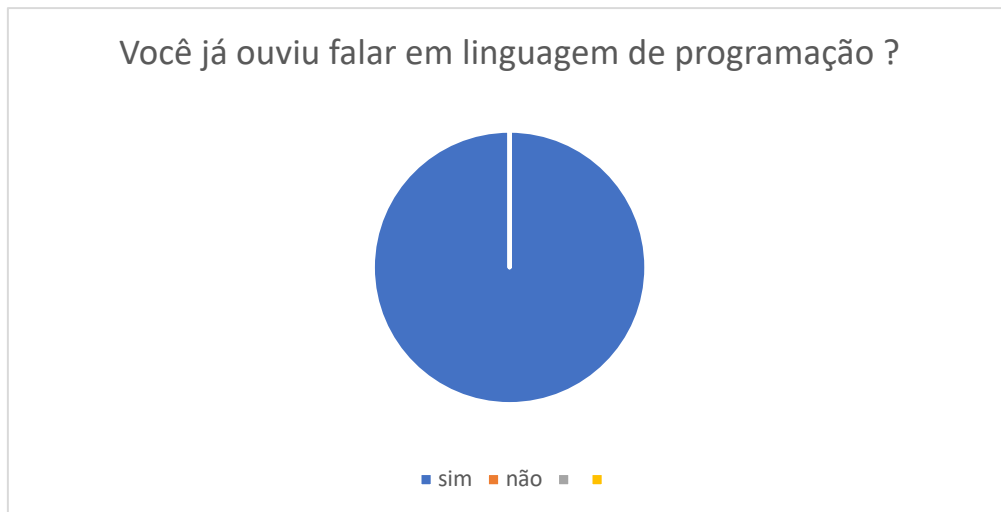
Foi aplicado um questionário com o objetivo de coletar feedback dos alunos sobre a sequência didática aplicada. Um total de 6 alunos realizaram a sequência didática. A seguir apresentamos os resultados baseado no questionário aplicado.

**Figura 32: Você gosta de matemática?**



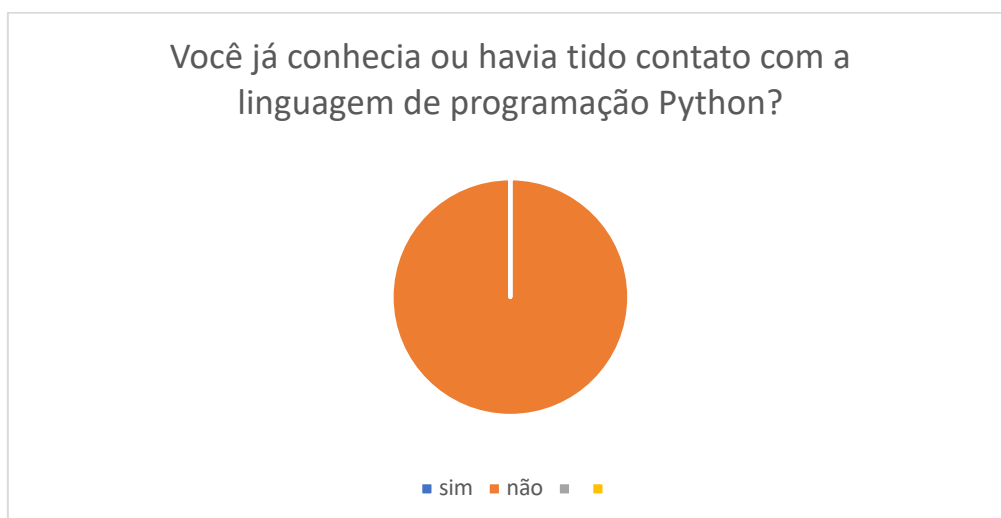
**Fonte:** Elaborado pelo autor (2023).

**Figura 33: Você já ouviu falar em linguagem de programação?**



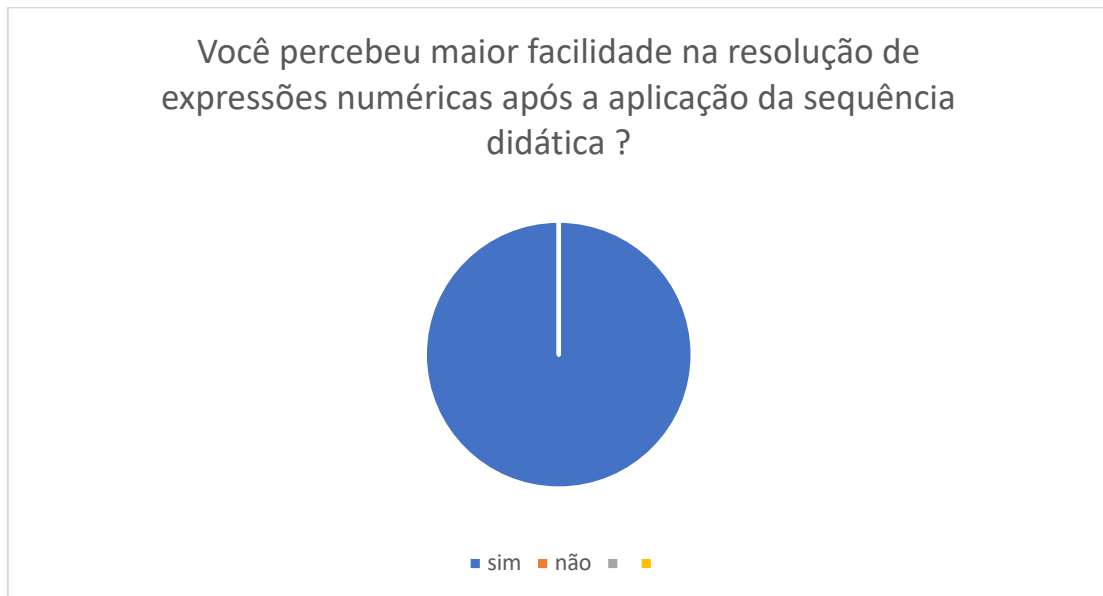
**Fonte:** Elaborado pelo autor (2023).

**Figura 34: Você já conhecia ou havia tido contato com a linguagem de programação Python?**



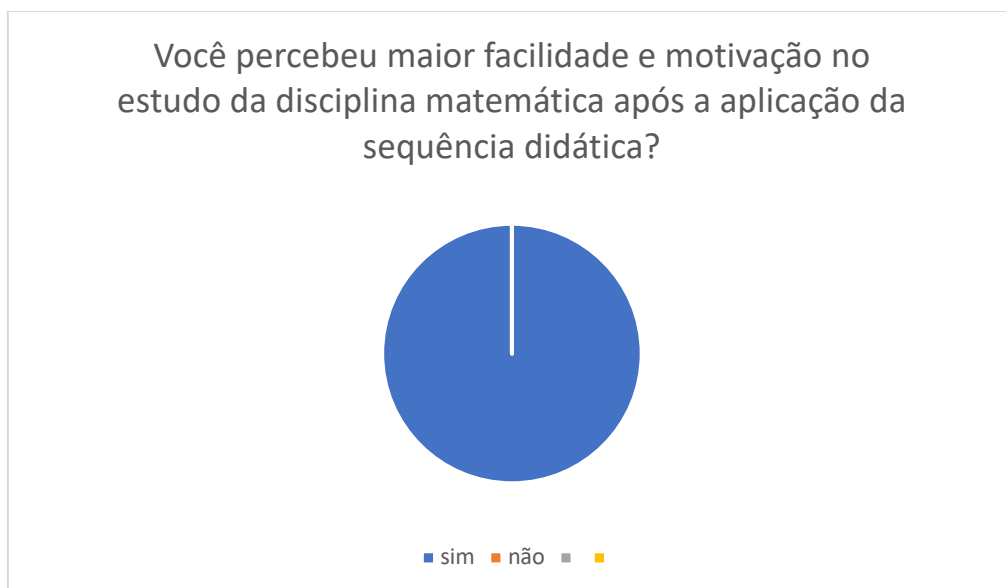
**Fonte:** Elaborado pelo autor (2023).

**Figura 35: Você percebeu maior facilidade na resolução de expressões numéricas após a aplicação da sequência didática?**



**Fonte:** Elaborado pelo autor (2023).

**Figura 36: Você percebeu maior facilidade e motivação no estudo da disciplina matemática após a aplicação da sequência didática?**



**Fonte:** Elaborado pelo autor (2023).

## 5 CONSIDERAÇÕES FINAIS

Aponte aqui, algumas sugestões de como tornar o aprendizado do aluno mais significativo e interessante, por meio da Linguagem de Programação *Python* que proporciona ao professor um subsídio a mais para avaliar a aprendizagem dos discentes e abrilhantar sua criatividade em suas aulas.

Analisou-se com essa pesquisa que a Linguagem *Python* uma vez aplicada em sala de aula com intencionalidade e aliada a um bom planejamento pedagógico, tornou-se um ótimo recurso capaz de beneficiar todas as partes envolvidas no processo educacional.

Foi possível perceber que ao aplicar esta linguagem nas aulas de expressões numéricas, ocorreu momentos de espontaneidade e interação, dando abertura a questionamentos e situações importantes de observações, de uma maneira interativa e que ao mesmo tempo, contribuiu no processo de avaliação.

Outro aspecto observado foi que o ensino se tornou prazeroso aos olhos dos alunos, apresentou uma linguagem simples e de fácil entendimento, fazendo com que eles participassem efetivamente do processo educacional, Estabelecendo uma relação entre o cognitivo e o afetivo dos envolvidos.

Os alunos mostraram-se estimulados e interessados. Era algo novo, atrativo, que chamou a atenção deles. Observaram-se mudanças significativas ao longo das atividades e no final a maioria conseguiu entender e realizaram as atividades propostas. Logo, notou-se que proporcionou uma aprendizagem diferenciada, quase lúdica, ampliando as experiências do alunado, dando a eles a oportunidade de conhecer a realidade da matemática de uma maneira mais concreta, palpável, criativa e de serem mais versáteis.

Devido à riqueza desse material, há possibilidade de criar atividades para diversos conteúdos em matemática com foco na Educação Básica, modificando seus níveis de dificuldade e deste modo o educador, investigando os níveis de conhecimento dos alunos, consegue criar novos contextos para descobertas. Portanto esse trabalho pode servir como um trampolim para o professor aprimorar e disseminar o conhecimento da linguagem matemática utilizando a linguagem de programação *Python* e para que os alunos fiquem mais familiarizados, próximos e com menos ojeriza da rainha das ciências, a nossa matemática.

## REFERÊNCIAS BIBLIOGRÁFICAS

AÇÃO EDUCATIVA; INSTITUTO PAULO MONTENEGRO. INAF Brasil 2018: Resultados preliminares. São Paulo: Ação Educativa / IPM, 2018. Disponível em: . Acesso em: 30 jul. 2023

ALVES, EVANILSON LANDIM. **Menos com menos é menos ou é mais?** Resolução de problemas de multiplicação e divisão de números inteiros por alunos do ensino regular e da educação de jovens e adultos. Recife. Dissertação (Em Educação Matemática e Tecnológica). Universidade Federal de Pernambuco, 2012.

ALVES, L. L. **A importância da matemática nos anos iniciais.** In: XXII Erematsul, 2016, Curitiba. Anais do XXII EREMATSUL. Curitiba: UNIANDRADE, 2016.

BECHARA, Evanildo. **Moderna gramática portuguesa.** 37ª ed. revista e ampliada conforme o novo Acordo Ortográfico. Rio de Janeiro, Nova Fronteira, 2009.

BETTEGA, Maria H.S. **Educação continuada na era digital.** 2ed. São Paulo: Cortez, 2010.

BLOCH, Marc. **Apologia da história ou o ofício do historiador.** Rio de Janeiro: Jorge Zahar, 2001.

BNCC e **programação de computadores: qual a relação?** Instituto Ayrton Senna. São Paulo, 2020.

BORBA, MARCELO DE CARVALHO; SILVA, RICARDO SCUCUGLIA DA; GADANIDIS, GEORGE. **Fases das tecnologias digitais em Educação Matemática: sala de aula e internet em movimento.** 2. ed. Belo Horizonte: Autêntica, 2018. 155 p.

BOYER, Carl Benjamin. **História da matemática.** São Paulo: Edgard Blucher Ltda., 1974. :

BRASIL. Ministério da Educação. **Base Nacional Comum Curricular** Brasília: MEC/SEB, 2017. Disponível em: <Disponível em: [http://basenacionalcomum.mec.gov.br/images/BNCC\\_EI\\_EF\\_110518\\_versaofinal\\_site.pdf](http://basenacionalcomum.mec.gov.br/images/BNCC_EI_EF_110518_versaofinal_site.pdf)> Acesso em: 30 jul.. 2023.

BRASIL. Ministério da Educação. **Parâmetros Curriculares Nacionais.** Brasília: MEC/SEF, 1998.

CAJORI, F. **Uma história da Matemática.** (2007). Rio de Janeiro: Ciência Moderna.

CAMPOS, CELSO RIBEIRO; WODEWOTZKI, MARIA LÚCIA Lorenzetti; JACOBINI, OTÁVIO ROBERTO. **Educação Estatística: teoria e prática em ambientes de modelagem matemática.** 2. ed. Belo Horizonte: Authentic, 2018. 143 p.

CARLOS, J. **Ordem de precedência.** 2023; imagem.

CARVALHO, Sônia Pinto de. **Área e o perímetro de um círculo.** 1º Colóquio de Matemática da Região Sudeste. Minas Gerais, 2011

CHARLES, R. SEVERANCE. **Python para todos.** Campina Grande, 2009.

D'AMBROSIO, Ubiratan. **Da realidade à ação: reflexões sobre Educação e Matemática**. São Paulo: Summus: Unicamp, 1986.

EVES, Howard. **Introdução à História da Matemática** 3. ed. Campinas: Editora da UNICAMP, 1994 (edição 2002). Tradutor: Hygino H. Domingues. Original: An introduction on the history of mathematics (1964).

FERRARI, F; CECHINEL, **Introdução a algoritmos e programação**. Bagé, 2008.

FLEMING, M.D; GONÇALVES, M.B: **Cálculo A**; Florianópolis, Editora Pearson Education, 2006.

FLICK, UWE. **Desenho da pesquisa qualitativa**. Tradução Roberto Cataldo Costa. Porto Alegre: Artmed, 2009.

FREITAS, H. S. **Expressões numéricas e suas abordagens em livros didáticos de matemática do 6º ano do ensino fundamental adotados por uma escola pública de Cuiabá-MT**. 2014. 111 f. Dissertação (Mestrado em Educação) – Universidade Federal de Mato Grosso, Cuiabá, 2014.

GIL, A. C. **Métodos e Técnicas de Pesquisa Social**. 6ª.ed. São Paulo: Atlas, 2011.

Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (Inep). **Censo da Educação Superior 2020**: notas estatísticas. Brasília, DF: Inep, 2022.

KENSKI, V. M. Educação e tecnologias: o novo ritmo da informação. Campinas: Editora Papirus, 2012. 141p.

LEITHOLD, LOUIS: **O Cálculo com geometria Analítica**; São Paulo, Editora Harbra Ltda, 1994.

LIMA, ELON LAGES. **Números e funções reais**-Rio de Janeiro, 2013 (Coleção Profmat).

Lima, Elon L. "O que é o número  $\pi$ ?". RPM 6, 1985, pp. 18-20.

LINGUAGEM. In: DICIONÁRIO Etimológico. Disponível em: [Etimologia e origem das palavras - Dicionário Etimológico \(dicionarioetimologico.com.br\)](http://Etimologia e origem das palavras - Dicionário Etimológico (dicionarioetimologico.com.br)) . Acesso em: 15 de ago. de 2023

LINO, PAULO: **Segmentos comensuráveis e incommensuráveis**, 2018; disponível em [Segmentos Comensuráveis e Incommensuráveis – Fatos Matemáticos \(wordpress.com\)](http://Segmentos Comensuráveis e Incommensuráveis – Fatos Matemáticos (wordpress.com))

LIPSCHUTZ, SEYMOUR: **Teoria dos Conjuntos**. São Paulo, Editora McGRAW-HILL DO BRIL, LTDA; 1978.

LOPES, A; POZEBON, S; SCALABRIN, T: **Constituição dos números inteiros: um movimento histórico**, 2021; disponível em [\\*Microsoft Word - 077 - Thanize Bortolini Scalabrin.doc \(ufpel.edu.br\)](http://*Microsoft Word - 077 - Thanize Bortolini Scalabrin.doc (ufpel.edu.br))

LUZ, Bruno Wallace Da Silva. SABIÃO, Roseline Martins. A evolução no ensino da matemática e a importância de se conhecer sua história. Revista Científica

Multidisciplinar Núcleo do Conhecimento. Ano 04, Ed. 08, Vol. 01, pp. 127-139. Agosto de 2019

MACHADO, Djeison. **Propostas Didáticas para o Ensino do Número  $\pi$** . 2013. 66f. Trabalho de Conclusão de Curso (Licenciatura em Matemática) - Universidade Federal de Santa Catarina, Florianópolis, 2013.

MAIOR, E. **e: a história de um número**. 5. ed. Trad. Jorge Calife. Rio de Janeiro: Editora Record, 2008.

MARCONDES, G. A. B. **Matemática com Python – Um guia prático**. São Paulo, 2018.

Melo, L. A. S. (2013). **Dois Jogos de Linguagem: a Informática e a Matemática na Aprendizagem de Função quadrática**. (Dissertação de Mestrado em Educação em Ciências e Matemáticas). Universidade Federal do Pará, Belém.

MENEZES, N. N. C. **Introdução à programação com Python**—2ª edição: Algoritmos e lógica de programação para iniciantes. [S.l.]: Novatec Editora, 2016.

OLIVEIRA, J. C.; GOMES, C. C. **Números Irracionais e Transcendentes**. 2009. 61 f. TCC (Professor Especialista em Matemática) - Universidade Federal de Santa Catarina e Universidade Virtual do Maranhão, Imperatriz, 2009

OLIVEIRA, RAMON DE..**Informática Educativa**. São Paulo: Papirus, 2012.

PAIS, L.C. **Didática da Matemática: uma análise da influência francesa**. 2ª. ed. Belo Horizonte: Autêntica, 2001.

PAPERT, S. **A máquina das crianças: Repensando a escola na era da informática**. ed. revis. Porto Alegre: artmed, 2008.

PARMEGIANI, R. **Contextualizando o Ensino das expressões numéricas no Ensino Fundamental**; Caxias do Sul: 2011.

Python org. Interactive Python Online. 2018. Disponível em: <https://www.python.org/shell/> . Acessado em 06/12/2022 às 20:00.

RAMOS, Luzia Faraco. **O que fazer primeiro?** 18. ed. São Paulo: Ática, 2002.

RENISCK, M. LearnToCode, CodetoLearn. MIT, 2012.

RIBEIRO, F.D. **Jogos e modelagem na educação matemática**. São Paulo: Saraiva, 2010.

ROSSUM, Guido. Disponível em: [gvanrossum.github.io/pics.html](https://github.com/gvanrossum/pics.html)

SÁ, Adriana Lourenço; MACHADO, Marília Costa. *O uso do software GeoGebra no estudo de funções*. XIV EVIDOSOL e XI CILTEC online, junho 2017. Disponível em: <https://eventos.textolivre.org/moodle/course/view.php?id=12>. Acesso em: 01 ago. 2023.

SEVERANCE, C. **Python for Informatics: Exploring Information**. Amazon CreateSpace 2009-2013.

SCALABRIN, T.B; **Sobre Ensino e aprendizagem de números negativos o que apontam as pesquisas**. Santa Maria, 2023.

SCHERER, D.; DUBOIS, P.; SHERWOOD, B. V **Python: 3D Interactive Scientific Graphics for Students**. Warner Hall: Carnegie Mellon University. 2000.

SUMMERFIELD, M. **Programação em Python 3: Uma introdução completa à linguagem Python**. [S.l.]: Alta Books, 2012. (Biblioteca do Programador). [https://download.inep.gov.br/publicacoes/institucionais/estatisticas\\_e\\_indicadores/notas\\_estatisticas\\_censo\\_da\\_educacao\\_superior\\_2020.pdf](https://download.inep.gov.br/publicacoes/institucionais/estatisticas_e_indicadores/notas_estatisticas_censo_da_educacao_superior_2020.pdf) – acesso em 15/05/2023 às 16:40:23.

ESTEVAM, Carvalho: **Top Programming Languages, 2022**. 09 nov 2022. Disponível em [Top Programming Languages 2022 \(dio.me\)](https://topprogramminglanguages.com/)

VENTURA, Felipe. **Python e Java empatam em ranking de linguagens de programação**. Tecnoblog. [S.l], 2020.

VYGOTSKY, L. S. **A formação social da mente**, 1991: o desenvolvimento dos processos

ZABALA, Antoni. **A prática educativa** – como ensinar. Porto Alegre: Artmed, 1998.



**ANEXO I: Aplicação da sequência didática proposta****Nome do aluno:****Segmento/etapa/turma:****1ª etapa: Situação-problema (problema gerador)**

A cidade de Quixeramobim, no interior do Ceará, produziu o maior queijo coalho artesanal do mundo, pesando 1.480 quilos. O recorde foi apresentado pela prefeitura do município na noite de 11 de agosto de 2022, durante a 4ª edição da Copa Leite, evento que reúne pecuaristas e produtores de laticínios.



Fonte: <https://g1.globo.com/ce/ceara/noticia/2022/11/12/cidade-do-ceara-produz-o-maior-queijo-coalho-do-mundo-pesando-1480-kg.ghml>

Na fabricação do queijo artesanal são utilizados basicamente os seguintes ingredientes: leite, sal e coagulante. Para cada 1 kg de queijo são necessários 10 litros de leite, 100 gramas de sal e 1 colher de sopa (15 ml) de coagulante ou coalho industrializado. Sabendo que atualmente os preços dos ingredientes são:

O preço médio de 1 ml do coagulante é de R\$ 0,16.

O preço médio de 1 litro de leite é de R\$ 2,00.

O preço médio de 100 gramas de sal é de R\$ 0,13.

Determine o custo total da produção do maior queijo artesanal do mundo em Quixeramobim/CE ocorrida em 2020.

**2ª etapa – Resolução de expressões numéricas utilizando a linguagem matemática**

**Atividade 1:**  $(4 \times 5) - 2$ .

**Atividade 2:**  $4 \times (5 - 2)$ .

**Atividade 3:**  $2 + (3 - 2) - (2 - 3) - 1$ .

**Atividade 4:**  $4 - \{3 + [2^3 - 5 + 1 - 2] - 1\} + 7$ .

**Atividade 5:**  $(3 - 6) + \{3 + (2 \times 0) - [2^3 - 3^2] - 1\}$ .

**Atividade 6:**  $3^0 - \{5 - [(10^1 - 3^2) + 8]\} - (-2)$ .

**Atividade 7:**  $\{(4 + 2^0) + [(-2 - 3) - (12 \times 100)] + (10^2 \div 5^2)\} - 5$ .

**3ª etapa: Resolução das expressões numéricas, propostas na 2ª etapa, utilizando a linguagem de programação Python.**

Agora você irá utilizar o modo interativo do Python e executar o IDLE (3.11.64 bit) já devidamente instalado no computador ou simplesmente utilizar o modo online sem precisar sobrecarregar a memória de seu instrumento computacional, para resolver as atividades propostas na segunda etapa.

**ANEXO II: Questionário utilizado para coletar o nível de satisfação dos alunos com a atividade proposta na sequência didática.**

### Questionário

No âmbito da dissertação de mestrado em matemática (Profmat) pela Universidade Rural do Semiárido (UFERSA) em Mossoró/RN com o título "A linguagem Python como certificação da solução de uma expressão numérica" o presente questionário tem por objetivo recolher informação dos alunos do CEJA a respeito da aplicação da sequência didática proposta na dissertação.

Solicita-se a sua colaboração na resolução individual das questões contidas neste questionário, garantindo-se que os dados e resultados serão tratados de forma totalmente profissional e anônima.

Obrigado.

01. Você gosta de matemática?

( ) Sim      ( ) NÃO

02. Você já ouviu falar em linguagem de programação?

( ) Sim      ( ) Não

03. Você já conhecia ou havia tido contato com a linguagem de programação Python?

( ) Sim      ( ) Não

04. Você percebeu maior facilidade na resolução de expressões numéricas após a aplicação da sequência didática?

( ) Sim      ( ) Não

05. Você percebeu maior facilidade e motivação no estudo da disciplina matemática após a aplicação da sequência didática?

( ) Sim      ( ) Não