

CENTRO UNIVERSITÁRIO FEI
JULIA VITÓRIA DE LIMA

**CALCULADORA EM PYTHON COM INTERFACE GRÁFICA PARA OS
CÁLCULOS ANALÍTICOS E VETORIAIS**

São Bernardo do Campo

2022

JULIA VITÓRIA DE LIMA

**CALCULADORA EM PYTHON COM INTERFACE GRÁFICA PARA OS
CÁLCULOS ANALÍTICOS E VETORIAIS**

Relatório Final de Iniciação Científica
apresentado ao Centro Universitário FEI,
como parte dos requisitos do Programa
PIBIC-FEI. Orientado pelo Prof. Dr.
Ariovaldo José de Almeida.

São Bernardo do Campo

2022

RESUMO

O curso de Engenharia oferece, em seu ciclo básico, diversas disciplinas voltadas para as ciências exatas. Em específico a disciplina de Geometria Analítica e Vetorial, que versa sobre a análise de vetores, planos e superfícies esféricas. Contemplando a disciplina de forma geral, sabe-se que uma grande porcentagem de alunos tem dificuldade com os cálculos e acabam tendo dificuldade de progredir no curso, gerando um baixo desempenho e um número alto de reprovações e dependências.

Nota-se também que alunos em períodos avançados voltam a sentir essas dificuldades, como por exemplo na disciplina de Eletromagnetismo, na área de Engenharia Elétrica.

Dessa forma, visando atenuar o problema dos alunos com estes cálculos, sugere-se a criação de uma calculadora, na qual existiriam recursos que facilitariam a interpretação e a análise do aluno, dispondo dos métodos de resolução e demonstração da figura em tese. Tais recursos seriam desenvolvidos com base em um amplo conhecimento de Python, em que os dados inseridos serão analisados e manipulados com funções específicas. O Python é uma linguagem de programação de alto nível, a qual os alunos têm o primeiro contato no ciclo básico dos cursos de Engenharia e Computação.

Palavras-chave:

1. Geometria Analítica e Vetorial.
2. Eletromagnetismo.
3. Calculadora.

LISTA DE ILUSTRAÇÕES

Figura 1 – Interface do WolframAlpha.....	8
Figura 2 – Interface do Symbolab.....	9
Figura 3 – Interface do 3D do Geogebra	10
Figura 4 – Demonstração do resultado dos dados inseridos na aplicação para vetores linearmente dependentes.....	18
Figura 5 – Demonstração do resultado dos dados inseridos na aplicação para vetores linearmente independentes	18
Figura 6 – Demonstração do resultado dos dados inseridos para cálculo das propriedades dos vetores na aplicação.	19
Figura 7 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código elaborado para dependência linear	19
Figura 8 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para mudança de base	22
Figura 9 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para mudança de base	22
Figura 10 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para mudança de base	23
Figura 11 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para produto escalar entre dois vetores.....	23
Figura 12 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para produto escalar entre três vetores.	24
Figura 13 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para produto vetorial	25
Figura 14 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para produto misto.....	25
Figura 15 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para a equação da reta vetorial	27
Figura 16 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para a equação da reta paramétrica	27
Figura 17 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para a equação da reta simétrica.....	28

Figura 18 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para verificação das retas reversas.....	30
Figura 19 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para verificação das retas paralelas	31
Figura 20 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para verificação das retas concorrentes	31
Figura 21 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para a equação do plano vetorial.....	33
Figura 22 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para a equação do plano paramétrica.....	33
Figura 23 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para a equação do plano geral ou cartesiana	33
Figura 24 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para posição relativa entre reta e plano (reta contida)	36
Figura 25 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para posição relativa entre reta e plano (reta paralela)	36
Figura 26 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para posição relativa entre reta e plano (reta concorrente e não perpendicular)	37
Figura 27 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para posição relativa entre reta e plano (reta concorrente e perpendicular).....	37
Figura 28 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para posição relativa entre planos (coincidentes)	39
Figura 29 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para posição relativa entre planos (paralelos distintos)	39
Figura 30 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para posição relativa entre planos (secantes não perpendiculares)	40
Figura 31 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para posição relativa entre planos (secantes perpendiculares)	40
Figura 32 – Linhas de código utilizadas para elaboração da tela inicial.....	41
Figura 33 – Linhas de código utilizadas para elaboração da tela inicial.....	41
Figura 34 – Demonstração da interface gráfica da tela inicial da calculadora.....	42
Figura 35 – Demonstração da interface gráfica da tela inicial da calculadora.....	42
Figura 36 – Demonstração do terminal de comando para produto escalar.....	43

SUMÁRIO

1	INTRODUÇÃO	7
2	REVISÃO BIBLIOGRÁFICA	8
2.1	WOLFRAMALPHA	8
2.2	SYMBOLAB	9
2.3	GEOGEBRA	9
2.4	PYTHON	10
2.4.1	TKINTER	10
2.5	VISUAL STUDIO CODE	11
2.6	REPLIT	11
3	METODOLOGIA	12
3.1	CONTEÚDO PROGRAMÁTICO	12
3.1.1	DEPENDÊNCIA LINEAR	12
3.1.2	MUDANÇA DE BASE	13
3.1.3	PRODUTO ESCALAR	13
3.1.4	PRODUTO VETORIAL	13
3.1.5	PRODUTO MISTO	14
3.1.6	RETAS	14
3.1.7	RETAS – POSIÇÃO RELATIVA	14
3.1.8	PLANOS	15
3.1.9	PLANOS – POSIÇÃO RELATIVA	15
3.2	PROGRAMAÇÃO	17
4	RESULTADOS PARCIAIS E DISCUSSÃO	18
4.1	PROGRAMAÇÃO	18
4.2	INTERFACE GRÁFICA	40
5	CRONOGRAMA	44
	REFERÊNCIAS	45

1 INTRODUÇÃO

A Engenharia é uma área ampla e multifocal, com ênfase na área das Ciências Exatas, campo em que se encontram disciplinas como: Cálculo Diferencial e Integral, Física, Geometria Analítica e Vetorial, entre outras. É de conhecimento geral que uma calculadora convencional pode atuar com excelência na resolução de problemas nas disciplinas de Cálculo e Física. Contudo, observando de outra perspectiva, é possível visualizar que tal método não é eficaz no que tange a disciplina de Geometria Analítica e Vetorial, no ciclo básico de Engenharia, e Eletromagnetismo, no sexto semestre de Engenharia Elétrica.

Nota-se, de forma geral, a dificuldade dos discentes do Centro Universitário FEI em resolver cálculos destas disciplinas, visto que as fórmulas são complicadas e demandam muito tempo de resolução, dificultando o êxito nas atividades avaliativas.

Por conseguinte, considerando a necessidade dos alunos, vê-se como alternativa a criação de um recurso capaz de realizar estes cálculos e ajudá-los a entender e visualizar o conteúdo de forma dinâmica e didática, em que possa haver a interação e a experiência do aluno com o que é lecionado. Outrossim, analisando o melhor método de criar este recurso, vê-se uma possibilidade interessante em inovar em algo que os alunos já têm acesso e convivem diariamente, que seria uma calculadora. O desenvolvimento seria feito com base na linguagem de programação Python, em que existem diversas possibilidades e funções capazes de suprir as necessidades de uma calculadora para cálculos vetoriais, como posição relativa entre planos, equações gerais de planos e retas, mudança de base, produtos escalares, vetoriais, mistos, entre outros.

2 REVISÃO BIBLIOGRÁFICA

Os cálculos vetoriais e analíticos possuem diversos processos para que se obtenha um resultado, seja a resolução de matrizes e determinantes, seja a utilização de fórmulas específicas. Dada a complexidade dos cálculos, vê-se a necessidade de haver um método de verificação da exatidão dos valores finais, visto que a calculadora deve ser precisa e exibir apenas valores corretos.

Para tanto, dispõe-se de softwares e aplicativos matemáticos capazes de suprir tais necessidades, cujas interfaces podem ter graus diferentes de dificuldade e aplicação. Em nosso projeto, selecionou-se três recursos, são eles: WolframAlpha, Symbolab e Geogebra. Além da linguagem de programação Python, uma de suas bibliotecas, Tkinter, para elaboração da interface gráfica e dois ambientes de estruturação de código, o Visual Studio Code e o Replit.

2.1 WOLFRAMALPHA

A interface deste site (<https://www.wolframalpha.com/>) é bem simples, dinâmica e ampla, dispondo de diversas funções voltadas tanto para a matemática, como para química e biologia. O processo é simples, basta digitar o que se deseja calcular e o site fornece a resposta detalhada, com passo a passo disponível. Utilizada na disciplina de Cálculo Diferencial e Integral I, será um recurso importante para a realização dos cálculos geométricos, como volume de um tetraedro, no conteúdo de produto misto.

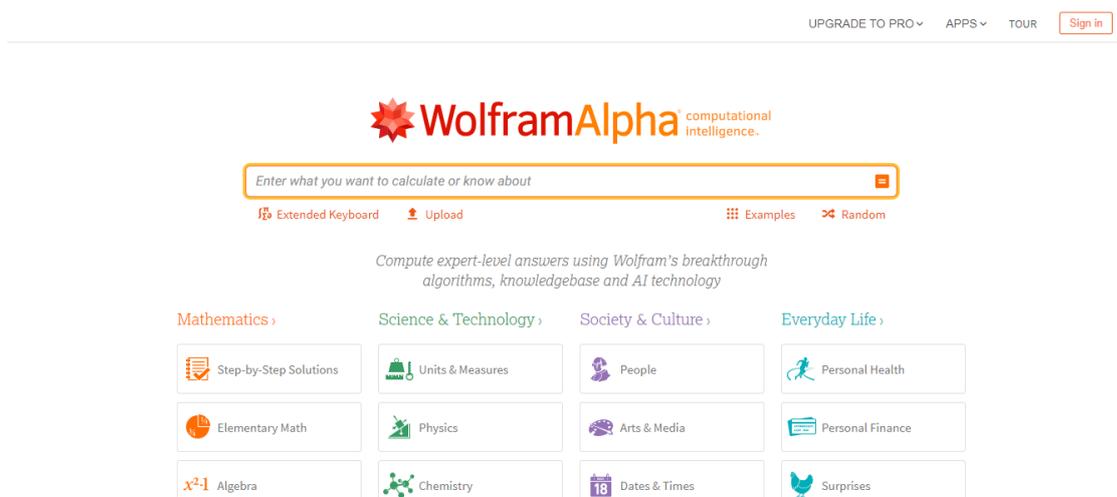


Figura 1 – Interface do WolframAlpha.

Fonte: Autor, 2021.

2.2 SYMBOLAB

O site (<https://pt.symbolab.com/>) trabalha de forma similar ao WolframAlpha, contudo, este possui ênfase em matemática. Ele fornece diversos meios de realizar os cálculos, assim como dispõe de gráficos, tabelas e solucionador geométrico. Tal recurso também será importante para a análise dos resultados obtidos.

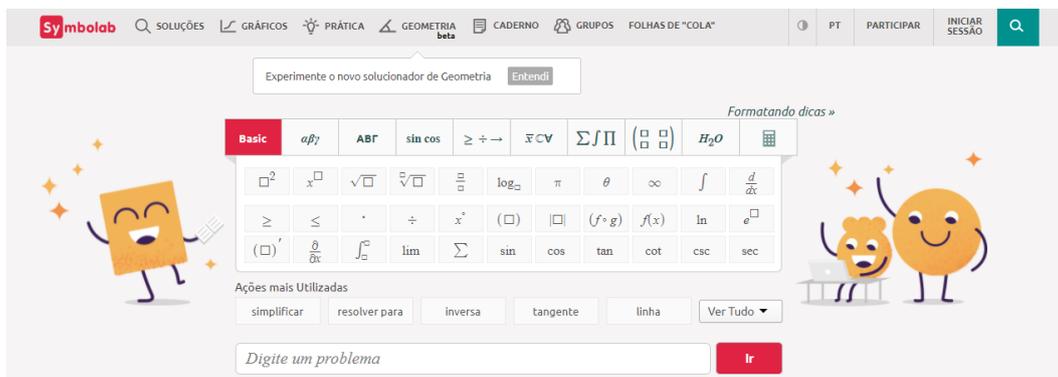


Figura 2 – Interface do Symbolab.

Fonte: Autor, 2021.

2.3 GEOGEBRA

O Geogebra é um software de geometria cujas funções variam entre elaboração de gráficos, cálculos de álgebra e o processo de equacionar planos e vetores, sendo um recurso essencial para o desenvolvimento do projeto devido à facilidade de executar as figuras geométricas e comparar com os resultados obtidos pela calculadora, como por exemplo as equações de retas, planos e as posições relativas. Suas funções são abordadas no ciclo básico de Engenharia, na disciplina de Laboratório de Matemática, em que são lecionadas todas as facilidades que este método dispõe. Levando em consideração a necessidade de trabalhar com figuras geométricas dinâmicas, o Geogebra será utilizado na função 3D, para que o software permita movimentar a ilustração obtida.

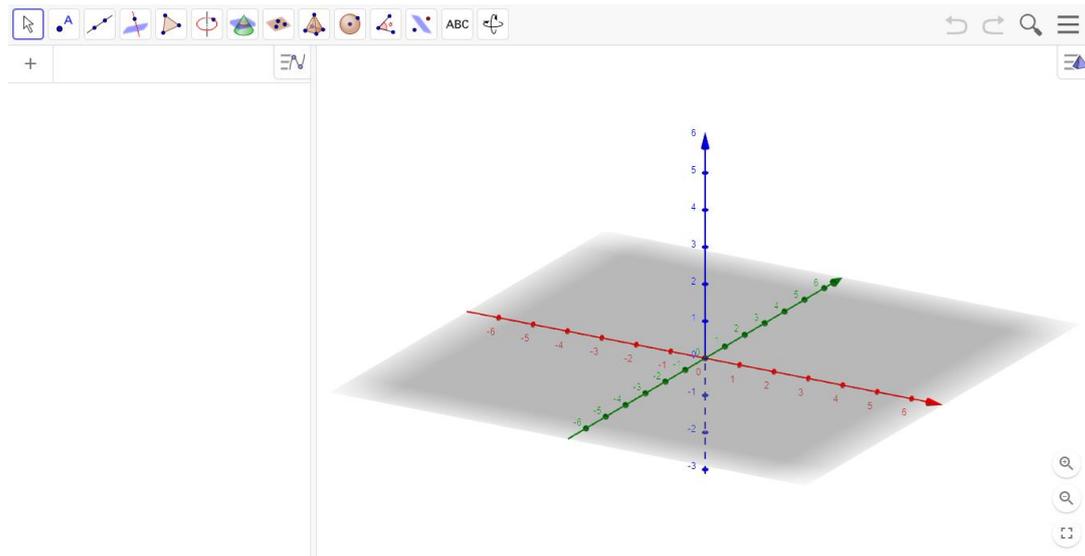


Figura 3 – Interface 3D do Geogebra.

Fonte: Autor, 2021.

2.4 PYTHON

O Python é uma linguagem de programação de alto nível, considerada a mais popular segundo a TIOBE, um índice da comunidade de programação, que busca elencar, de acordo com o número de pesquisas sobre a linguagem, qual é a mais procurada e utilizada pelos programadores. Devido à sua sintaxe simples e suas diversas funcionalidades, o Python torna possível executar diferentes estruturas de dados, realizando programas complexos, mas com códigos simples. Isso se dá, principalmente, devido às bibliotecas, que oferecem estruturas prontas e que facilitam a etapa de desenvolvimento da programação.

Essa linguagem “traz características que possibilitam escrever o mesmo requisito em menos linhas de código que o necessário em outras linguagens de programação e hoje, além de adotado na construção de soluções web, também está sendo muito utilizado em aplicações que lidam com processamento de texto, machine learning e recomendação de conteúdo.” (DEVMÉDIA, 2021)

2.4.1 TKINTER

O Tkinter é uma biblioteca da linguagem Python, cujo objetivo é o desenvolvimento de interfaces gráficas. O Python contém tal biblioteca já instalada, portanto, qualquer computador que tenha o interpretador da linguagem instalado, é possível criar interfaces gráficas usando o Tkinter, com exceção de algumas distribuições Linux, exigindo que seja feita o download do módulo separadamente. Neste trabalho, o

Tkinter tem como objetivo exibir uma tela para que o usuário escolha qual cálculo deseja realizar e, em um próximo momento, mostre o resultado daquilo que foi inserido.

2.5 VISUAL STUDIO CODE

A plataforma Visual Studio Code é um editor de código fonte criado pela Microsoft capaz de suportar diversas linguagens de programação, especialmente as mais conhecidas como C++, Java, XML e, principalmente a linguagem Python, abordada neste projeto de pesquisa. Por ter uma ampla paleta de recursos e comandos, permite a elaboração de códigos simples e complexos, além da instalação de bibliotecas específicas para o desenvolvimento dos códigos. É a plataforma ideal para a etapa de construção do código e complementação com a interface gráfica, visto que permite ajustes e personalizações para que o resultado obtido seja o mais próximo do idealizado.

2.6 REPLIT

Este recurso é uma plataforma de programação online (<https://replit.com/>), em que é possível desenvolver os códigos de acordo com a linguagem de programação desejada. É uma opção viável para a elaboração de projetos pois permite que os códigos sejam unificados ou separados em pastas, tornando a apresentação mais simples. Além disso, a plataforma é conhecida como o ambiente de programação online mais simples e dinâmico.

3 METODOLOGIA

Para a realização deste trabalho de pesquisa, foram utilizados diversos recursos, entre eles o livro Vetores e Geometria Analítica (LORETO, A. C. D. C; JUNIOR, A. P. L., 2014), que será a base para a coleta de fórmulas e informações acerca das demonstrações que a calculadora precisa oferecer, além do livro Fundamentos da Física – Eletromagnetismo (HALLIDAY, David; RESNICK, Robert; WALKER, Jearl, 2016), em que seriam coletados os dados de análise vetorial. Outrossim, por meio da linguagem de programação Python, serão desenvolvidas funções que obedecem a procedimentos específicos para demonstrar os resultados dos dados inseridos. O material base para a implementação da linguagem de programação será o livro Introdução à programação com Python: algoritmos e lógica de programação para iniciantes (MENEZES, N. N. C., 2010), assim como o site Curso em Vídeo (CURSO EM VÍDEO, 2021), em que é disponibilizado um material didático rico e completo que abrange todas as funcionalidades dessa linguagem. Por outro lado, a referência para a execução da interface gráfica é o livro Python para Desenvolvedores (BORGES, Luiz Eduardo, 2014).

A escolha da linguagem de programação se dá devido à facilidade de trabalhar com este recurso, visto que o Python é um código limpo, simples e legível em diversos tipos de aplicações, seja em programas complexos, seja em programas compactos (TECNOBLOG, 2021).

3.1 CONTEÚDO PROGRAMÁTICO

Para desenvolver este projeto de pesquisa, utilizou-se o conteúdo programático da disciplina de Geometria Analítica e Vetorial, presente na grade curricular básica do curso de Engenharia. Dentre a gama de assuntos abordados, selecionou-se alguns elementares para inserir na calculadora, que tornariam a experiência de diversos alunos mais simples e diversa, auxiliando em diferentes exercícios com abordagens diferentes. Os principais tópicos são Dependência Linear, Mudança de Base, Produto Escalar, Produto Vetorial, Produto Misto, Retas e Posição Relativa Entre Retas, Planos e Posição Relativa Entre Planos.

3.1.1 DEPENDÊNCIA LINEAR

A teoria de dependência linear se baseia no princípio de que, um conjunto de vetores é linearmente dependente um do outro se um for combinação linear dos outros.

Tal teoria varia de acordo com quantos vetores compõe o conjunto, visto que, no caso de um vetor, caso este seja nulo, será linearmente dependente, caso este não seja, será linearmente independente. No que se refere à dois e três vetores, são linearmente dependentes se um for(em) combinação linear do(s) outro(s), contudo, se este não for o caso, são linearmente independentes. Por fim, no caso de quatro ou mais vetores, estes serão linearmente dependentes em um espaço vetorial tridimensional, sendo coplanares e colineares.

3.1.2 MUDANÇA DE BASE

O conteúdo teórico de mudança de base considera o cálculo de um determinado vetor descrito em uma outra base, visando tornar a obtenção das coordenadas deste o mais simples possível. Para tanto, é necessário conhecer o vetor U em uma base E , e um vetor V na base F , o meio de conexão entre eles é um sistema cujas coordenadas em E são descritas por valores de F . Realizando dois cálculos distintos envolvendo matrizes e determinantes, é possível obter as novas coordenadas de U e de V em suas respectivas novas bases.

3.1.3 PRODUTO ESCALAR

Conforme a teoria, sabe-se que o produto entre dois vetores tem como resultado um número, ou seja, um número real pode ser obtido. Considerando dois vetores com coordenadas u_1, u_2, u_3 e v_1, v_2, v_3 , ambos em relação à uma base ortonormal B , é possível realizar uma multiplicação dos termos, sendo as coordenadas do primeiro multiplicadas de acordo com a sua respectiva posição na segunda, e, com essa multiplicação, obter um escalar.

3.1.4 PRODUTO VETORIAL

O produto vetorial pode ser descrito como o produto entre dois vetores, cujo resultado é um vetor. Considerando dois vetores com coordenadas u_1, u_2, u_3 e v_1, v_2, v_3 , ambos em relação à uma base ortonormal B , é possível realizar o cálculo de um determinante entre essas coordenadas e as coordenadas do espaço vetorial tridimensional $(\hat{i}, \hat{j}, \hat{k})$. Dessa forma, o resultado obtido terá valor respectivo à cada uma das posições do espaço vetorial tridimensional.

3.1.5 PRODUTO MISTO

O conteúdo de produto misto pode ser simplificado como sendo uma mistura entre os produtos escalar e vetorial, em que o resultado é um número. Considerando três vetores com coordenadas u_1, u_2, u_3 ; v_1, v_2, v_3 e w_1, w_2, w_3 , sendo estes em relação à uma base ortonormal B, é possível realizar o cálculo do determinante entre as coordenadas, obtendo assim o resultado escalar (número). Por meio do produto misto também é possível obter o volume de algumas formas geométricas, por exemplo o paralelepípedo, tetraedro e prisma retangular.

3.1.6 RETAS

A abordagem teórica define que uma reta é descrita como sendo um ponto e um vetor não nulo paralelo a ela. Consideram-se dois pontos distintos, sendo um deles um ponto qualquer (x, y, z) e o outro com coordenadas conhecidas (x_1, y_1, z_1) , além disso, considera-se também um vetor diretor $\vec{U} = (u_1, u_2, u_3)$ em relação à uma base ortonormal e o escalar α . Com estes dados, é possível elaborar a equação vetorial da reta, em que:

$$r: (x, y, z) = (x_1, y_1, z_1) + \alpha(u_1, u_2, u_3)$$

3.1.7 RETAS – POSIÇÃO RELATIVA

Complementando a teoria de retas, analisa-se como é o comportamento entre estas, ou seja, a posição relativa. As retas podem ser concorrentes, paralelas ou reversas. Para realizar a análise da posição, são necessários os vetores diretores das duas retas, assim como o vetor originado pelos dois pontos conhecidos, criando-se um vetor \overrightarrow{AB} . As retas concorrentes são as que estão no mesmo plano e tem um ponto de intersecção. As retas paralelas são as retas cujos vetores diretores são paralelos. As retas reversas não são coplanares. O conjunto entre os vetores diretores e este vetor \overrightarrow{AB} é linearmente dependente, logo, o determinante entre eles é igual a 0, fazendo com que os vetores não sejam paralelos; as paralelas são as que estão no mesmo plano, apenas. O conjunto entre os vetores diretores e este vetor \overrightarrow{AB} é linearmente dependente, logo, o determinante entre eles é igual a 0, e, já que não há um ponto de encontro, os vetores são paralelos; as reversas estão em planos diferentes. O conjunto entre os vetores diretores e este vetor \overrightarrow{AB} é linearmente independente, logo, o determinante entre eles é diferente de 0, fazendo com que os vetores sejam não coplanares.

3.1.8 PLANOS

Os planos podem ser definidos, de acordo com a teoria, como uma superfície que contém pontos, retas e vetores. O plano possui equações e um sistema de coordenadas, cujos dados é possível obter a partir de um ponto e dois vetores. Além disso, o plano possui um vetor normal em relação a ele, que pode ser descrito como um vetor ortogonal a todos os vetores paralelos a esse plano; o produto escalar entre ele e qualquer vetor paralelo ao plano é 0. Considera-se um ponto com coordenadas conhecidas (x_0, y_0, z_0) , os dois coeficientes escalares α e β , além de dois vetores diretores $\vec{U} = (u_1, u_2, u_3)$ e $\vec{V} = (v_1, v_2, v_3)$ em relação à uma base ortonormal. Com estes dados, é possível elaborar a equação vetorial do plano, em que:

$$p: (x, y, z) = (x_0, y_0, z_0) + \alpha(u_1, u_2, u_3) + \beta(v_1, v_2, v_3)$$

Tal equação também pode gerar uma equação paramétrica do plano, em que se coloca em evidência cada uma das coordenadas cartesianas x, y e z.

$$p: \begin{cases} x = x_0 + u_1\alpha + v_1\beta \\ y = y_0 + u_2\alpha + v_2\beta \\ z = z_0 + u_3\alpha + v_3\beta \end{cases}$$

Também é possível obter a equação geral ou cartesiana do plano, cujo formato é:

$$ax + by + cz + d = 0$$

Para tanto, é necessário obter um terceiro vetor originado pela diferença entre um ponto qualquer (x, y, z) e um dos pontos conhecidos. Por fim, utilizando os conteúdos teóricos de produtos escalares, vetoriais, mistos e determinantes, obtém-se a equação geral.

3.1.9 PLANOS – POSIÇÃO RELATIVA

Complementando a teoria de planos, analisa-se como estes se comportam entre si e em relação as retas. O conteúdo inicia com a abordagem das posições relativas entre as retas e planos, as quais se dividem em quatro tópicos principais, são eles:

- Reta contida no plano: Tal situação ocorre quando o produto escalar entre o vetor normal ao plano e o vetor diretor da reta é igual à 0. Além disso, o ponto da reta deve, obrigatoriamente, pertencer ao plano.

- Reta paralela ao plano: Tal situação ocorre quando o produto escalar entre o vetor normal ao plano e o vetor diretor da reta é igual à 0. Neste caso, o ponto não pode pertencer ao plano.
- Reta concorrente ao plano, mas não perpendicular: Tal situação ocorre quando o produto escalar entre o vetor normal ao plano e o vetor diretor da reta é diferente de 0 e, portanto, os vetores serão linearmente independentes. Além disso, o ponto de intersecção entre o plano e a reta deve pertencer ao plano.
- Reta concorrente e perpendicular ao plano: Tal situação ocorre quando o vetor normal ao plano e o vetor diretor da reta são paralelos e, dessa forma, linearmente dependentes. Outrossim, o ponto de intersecção entre o plano e a reta deve pertencer ao plano.

Já no que tange as posições relativas entre dois planos, existem cinco tópicos principais, aos quais são determinados com base na análise das duas equações gerais de planos distintos:

- Coincidentes: Este caso ocorre quando as coordenadas a, b, c e d de um dos planos são divididas pelas coordenadas a_1, b_1, c_1 e d_1 do outro plano e originam valores iguais e diferentes de 0. Além disso, os vetores normais de ambos os planos são linearmente dependentes.
- Paralelos distintos: Este caso ocorre quando as coordenadas a, b e c de um dos planos são divididas pelas coordenadas a_1, b_1 e c_1 do outro plano e originam valores iguais e diferentes de 0, com exceção da divisão entre d e d_1 , cujo resultado é diferente de 0 e do resultado obtido pelas demais divisões.
- Secantes não perpendiculares: Tal situação ocorre quando o produto escalar entre os vetores normais dos planos é diferente de 0, assim como ambos são linearmente independentes. Os planos, neste caso, não fazem 90° entre si.
- Secantes perpendiculares: Tal situação ocorre quando o produto escalar entre os vetores normais dos planos é igual a 0, assim como ambos são linearmente independentes. Os planos, neste caso, fazem 90° entre si.
- Intersecção entre dois planos: Neste caso, os dois planos originam uma reta que pode ser descrita com base na equação destes.

3.2 PROGRAMAÇÃO

Com base no conteúdo programático, iniciou-se um roteiro com todos os processos que os programas deveriam seguir para cumprir com suas respectivas funções e realizar os cálculos. Fazendo uso da plataforma Replit, realizou-se, de forma separada, a elaboração dos programas dos conteúdos propostos, sendo eles: Dependência Linear, Mudança de Base, Produto Escalar, Produto Vetorial, Produto Misto, Equação e Posição Relativa de Retas e Posição Relativa e Equação de Planos. Optou-se pelo desenvolvimento separado para que os testes realizados fossem simples e objetivos, capazes de captar as possíveis falhas na construção do código.

A plataforma Visual Studio Code foi utilizada para sintetizar as programações realizadas. Inicialmente, esquematizou-se por meio de diagramas e fluxogramas quais funções e em qual ordem o aplicativo iria desempenhar. Em seguida, iniciou-se a etapa de programação, para tornar possível uma conexão entre as funções, e testes, nas quais os códigos foram estruturados e adaptados, além de testados, a cada alteração.

4 RESULTADOS PARCIAIS E DISCUSSÃO

4.1 PROGRAMAÇÃO

A calculadora projetada para realizar cálculos de Geometria Analítica e Eletromagnetismo pode ter sua eficácia verificada em alguns casos específicos por meio de exercícios propostos que, ao utilizá-la para resolver, pode-se comparar com os valores obtidos em cálculos manuais. Considerando as etapas do projeto já realizadas, têm-se a programação finalizada, o que permite testar valores e apurar a qualidade do material elaborado.

Os exemplos realizados neste tópico foram retirados do material didático auxiliar da disciplina de Geometria Analítica e Vetorial.

Os casos em que se pode analisar sua exatidão, são:

1) Dependência Linear;

No que se refere à dependência linear em relação à um vetor, há somente duas hipóteses. Caso ele seja nulo, ele é linearmente dependente, ou seja, é um ponto. Confirmando essa premissa, vê-se na aplicação desenvolvida o resultado emitido.



```

1 v = int(input("Digite quantos vetores serão analisados: "))
2
3 if v == 1:
4     ve = int(input("Digite o vetor: "))
5     if ve == 0:
6         print("0 vetor é linearmente dependente")
7     if ve != 0:
8         print("0 vetor é linearmente independente")
9

```

```

Digite quantos vetores serão analisados: 1
Digite o vetor: 0
0 vetor é linearmente dependente
>

```

Figura 4 – Demonstração do resultado dos dados inseridos na aplicação para vetores linearmente dependentes.

Fonte: Autor, 2022.

Por outro lado, caso ele não seja nulo, ele é linearmente independente, ou seja, é uma reta.



```

1 v = int(input("Digite quantos vetores serão analisados: "))
2
3 if v == 1:
4     ve = int(input("Digite o vetor: "))
5     if ve == 0:
6         print("0 vetor é linearmente dependente")
7     if ve != 0:
8         print("0 vetor é linearmente independente")
9

```

```

Digite quantos vetores serão analisados: 1
Digite o vetor: 1
0 vetor é linearmente independente
>

```

Figura 5 – Demonstração do resultado dos dados inseridos na aplicação para vetores linearmente independentes.

Fonte: Autor, 2022.

Em relação à dois vetores, propõe-se o cálculo da dependência linear deste conjunto de vetores.

$$\{(1,2,-1), (-2,-4,2)\}$$

$$\begin{cases} 1\alpha = -2 \\ 2\alpha = -4 \\ -1\alpha = 2 \end{cases} \rightarrow \begin{cases} \alpha = \frac{-2}{1} \\ \alpha = \frac{-4}{2} \\ \alpha = \frac{2}{-1} \end{cases} \rightarrow \begin{cases} \alpha = -2 \\ \alpha = -2 \\ \alpha = -2 \end{cases}$$

Conforme visto por meio do cálculo manual acima, sabe-se que o resultado deve apresentar o resultado de que os vetores são linearmente dependentes, uma vez que os três alfas possuem o mesmo valor. Portanto, o cálculo utilizando a programação executada deve mostrar os mesmos valores.

```
Digite quantos vetores serão analisados:2
Qual a coordenada u1 do vetor diretor u? 1
Qual a coordenada u2 do vetor diretor u? 2
Qual a coordenada u3 do vetor diretor u? -1
Qual a coordenada v1 do vetor diretor v? -2
Qual a coordenada v2 do vetor diretor v? -4
Qual a coordenada v3 do vetor diretor v? 2
Os vetores são linearmente dependentes
```

Figura 6 – Demonstração do resultado dos dados inseridos para cálculo das propriedades dos vetores na aplicação.

Fonte: Autor, 2022.

Agora, realizando o cálculo de um outro exemplo cujo resultado deve apresentar que os vetores são linearmente independentes, utiliza-se o conjunto de vetores abaixo:

$$\{(1,-2,3), (2,1,0)\}$$

$$\begin{cases} 1\alpha = 2 \\ -2\alpha = 1 \\ 3\alpha = 0 \end{cases} \rightarrow \begin{cases} \alpha = \frac{2}{1} \\ \alpha = \frac{1}{-2} \\ \alpha = \frac{0}{3} \end{cases} \rightarrow \begin{cases} \alpha = 2 \\ \alpha = -\frac{1}{2} \\ \alpha = 0 \end{cases}$$

Cujo resultado no simulador apresenta igualdade com o calculado manualmente.

```
Digite quantos vetores serão analisados: 2
Qual a coordenada u1 do vetor diretor u? 1
Qual a coordenada u2 do vetor diretor u? -2
Qual a coordenada u3 do vetor diretor u? 3
Qual a coordenada v1 do vetor diretor v? 2
Qual a coordenada v2 do vetor diretor v? 1
Qual a coordenada v3 do vetor diretor v? 0
Os vetores são linearmente independentes
```

Figura 7 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código elaborado para dependência linear.

Fonte: Autor, 2022.

No caso de três vetores, propõe-se o cálculo dos conjuntos de vetores abaixo, em que o primeiro será linearmente dependente e o segundo linearmente independente:

$$\{(-1,0,1), (1,2,-1), (0,2,0)\}$$

$$\Delta = \begin{vmatrix} -1 & 0 & 1 \\ 1 & 2 & -1 \\ 0 & 2 & 0 \end{vmatrix} = 2 - 2 = 0$$

$$\{(1,2,-1), (0,1,0), (0,0,1)\}$$

$$\Delta = \begin{vmatrix} 1 & 2 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix} = 1$$

Por fim, como é conhecido na teoria, conjuntos com quatro vetores sempre serão linearmente dependentes em um espaço vetorial tridimensional, conforme pode-se verificar no simulador.

2) Mudança de Base;

Visto que é dado um vetor U em uma base E , e um outro vetor V em base F , deseja-se escrever esse vetor U em base F , assim como o vetor V na base E , verificando por meio do exemplo abaixo:

$$\begin{aligned} \vec{U} &= 2\vec{e}_1 - \vec{e}_2 + 3\vec{e}_3 \\ \vec{V} &= 3\vec{f}_1 + 3\vec{f}_2 + 2\vec{f}_3 \end{aligned} \quad \left\{ \begin{array}{l} \vec{e}_1 = 2\vec{f}_1 + \vec{f}_3 \\ \vec{e}_2 = \vec{f}_1 - \vec{f}_2 \\ \vec{e}_3 = 1\vec{f}_1 + 2\vec{f}_2 + \vec{f}_3 \end{array} \right.$$

Calculando o vetor U na base F :

$$\vec{U}_f = M_{F \rightarrow E} * \vec{U}_E$$

$$\vec{U}_f = \begin{bmatrix} 2 & 1 & 1 \\ 0 & -1 & 2 \\ 1 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 2 \\ -1 \\ 3 \end{bmatrix} = \begin{bmatrix} 4 - 1 + 3 \\ 0 + 1 + 6 \\ 2 + 0 + 3 \end{bmatrix} = \begin{bmatrix} 6 \\ 7 \\ 5 \end{bmatrix}$$

Logo, o vetor \vec{U} na base F pode ser representado por:

$$\vec{U}_f = 6\vec{f}_1 + 7\vec{f}_2 + 5\vec{f}_3$$

Calculando o vetor V na base E :

$$M_{F \rightarrow E} = \begin{bmatrix} 2 & 1 & 1 \\ 0 & -1 & 2 \\ 1 & 0 & 1 \end{bmatrix}$$

$$M_{E \rightarrow F} = M_{F \rightarrow E}^{-1} = \frac{1}{\Delta} * [coef(m)]^t$$

Encontrando Δ :

$$\Delta = \begin{vmatrix} 2 & 1 & 1 & 2 & 1 \\ 0 & -1 & 2 & 0 & -1 \\ 1 & 0 & 1 & 1 & 0 \end{vmatrix} = -2 + 2 - 0 + 1 = 1$$

Calculando o $coef(m)$:

$$Coef(m) = \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix}$$

$$A_{ij} = (-1)^{i+j} * D_{ij}$$

$$C_{11} = (-1)^{1+1} * \begin{bmatrix} -1 & 2 \\ 0 & 1 \end{bmatrix} = (1) * (-1) = -1$$

$$C_{12} = (-1)^{1+2} * \begin{bmatrix} 0 & 2 \\ 1 & 1 \end{bmatrix} = (-1) * (-2) = 2$$

$$C_{13} = (-1)^{1+3} * \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} = (1) * (1) = 1$$

$$C_{21} = (-1)^{2+1} * \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} = (-1) * (1) = -1$$

$$C_{22} = (-1)^{2+2} * \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} = (1) * (1) = 1$$

$$C_{23} = (-1)^{2+3} * \begin{bmatrix} 2 & 1 \\ 1 & 0 \end{bmatrix} = (-1) * (-1) = 1$$

$$C_{31} = (-1)^{3+1} * \begin{bmatrix} 1 & 1 \\ -1 & 2 \end{bmatrix} = (1) * (3) = 3$$

$$C_{32} = (-1)^{3+2} * \begin{bmatrix} 2 & 1 \\ 0 & 2 \end{bmatrix} = (-1) * (4) = -4$$

$$C_{33} = (-1)^{3+3} * \begin{bmatrix} 2 & 1 \\ 0 & -1 \end{bmatrix} = (1) * (-2) = -2$$

Logo:

$$Coef(m) = \begin{bmatrix} -1 & 2 & 1 \\ -1 & 1 & 1 \\ 3 & -4 & -2 \end{bmatrix}$$

$$Coef(m)^t = \begin{bmatrix} -1 & -1 & 3 \\ 2 & 1 & -4 \\ 1 & 1 & -2 \end{bmatrix}$$

$$M_{F \rightarrow E}^{-1} = \frac{1}{\Delta} * [coef(m)]^t = \frac{1}{1} * \begin{bmatrix} -1 & -1 & 3 \\ 2 & 1 & -4 \\ 1 & 1 & -2 \end{bmatrix} = \begin{bmatrix} -1 & -1 & 3 \\ 2 & 1 & -4 \\ 1 & 1 & -2 \end{bmatrix}$$

$$\vec{V} = \begin{bmatrix} -1 & -1 & 3 \\ 2 & 1 & -4 \\ 1 & 1 & -2 \end{bmatrix} * \begin{bmatrix} 3 \\ 3 \\ 2 \end{bmatrix} = \begin{bmatrix} -3 - 3 + 6 \\ 6 + 1 - 8 \\ 3 + 3 - 4 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix}$$

Logo, o vetor \vec{V} na base E pode ser representado por:

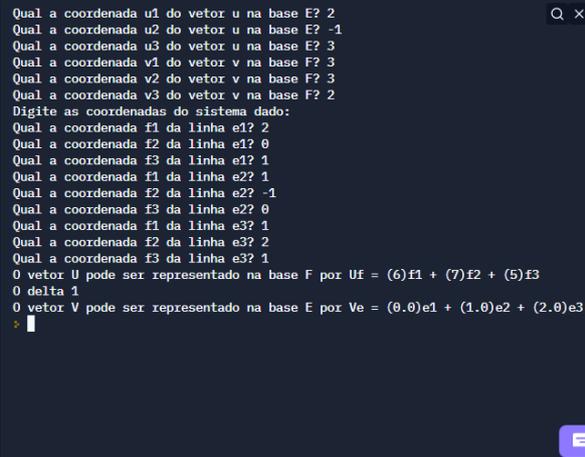
$$\vec{V}_e = 0\vec{e}_1 + 1\vec{e}_2 + 2\vec{e}_3$$

Realizando o cálculo por meio do código elaborado, é possível verificar a coerência dos resultados obtidos:

```

1  vetoru1 = int(input("Qual a coordenada u1 do vetor u na base E? "))
2  vetoru2 = int(input("Qual a coordenada u2 do vetor u na base E? "))
3  vetoru3 = int(input("Qual a coordenada u3 do vetor u na base E? "))
4  vetorv1 = int(input("Qual a coordenada v1 do vetor v na base F? "))
5  vetorv2 = int(input("Qual a coordenada v2 do vetor v na base F? "))
6  vetorv3 = int(input("Qual a coordenada v3 do vetor v na base F? "))
7
8  print("Digite as coordenadas do sistema dado:")
9  e1f1 = int(input("Qual a coordenada f1 da linha e1? "))
10 e1f2 = int(input("Qual a coordenada f2 da linha e1? "))
11 e1f3 = int(input("Qual a coordenada f3 da linha e1? "))
12 e2f1 = int(input("Qual a coordenada f1 da linha e2? "))
13 e2f2 = int(input("Qual a coordenada f2 da linha e2? "))
14 e2f3 = int(input("Qual a coordenada f3 da linha e2? "))
15 e3f1 = int(input("Qual a coordenada f1 da linha e3? "))
16 e3f2 = int(input("Qual a coordenada f2 da linha e3? "))
17 e3f3 = int(input("Qual a coordenada f3 da linha e3? "))
18
19 uf1a = e1f1*vetoru1
20 uf1b = e1f2*vetoru1
21 uf1c = e1f3*vetoru1
22 uf2a = e2f1*vetoru2
23 uf2b = e2f2*vetoru2
24 uf2c = e2f3*vetoru2
25 uf3a = e3f1*vetoru3
26 uf3b = e3f2*vetoru3
27 uf3c = e3f3*vetoru3

```



```

Qual a coordenada u1 do vetor u na base E? 2
Qual a coordenada u2 do vetor u na base E? -1
Qual a coordenada u3 do vetor u na base E? 3
Qual a coordenada v1 do vetor v na base F? 3
Qual a coordenada v2 do vetor v na base F? 3
Qual a coordenada v3 do vetor v na base F? 2
Digite as coordenadas do sistema dado:
Qual a coordenada f1 da linha e1? 2
Qual a coordenada f2 da linha e1? 0
Qual a coordenada f3 da linha e1? 1
Qual a coordenada f1 da linha e2? 1
Qual a coordenada f2 da linha e2? -1
Qual a coordenada f3 da linha e2? 0
Qual a coordenada f1 da linha e3? 1
Qual a coordenada f2 da linha e3? 2
Qual a coordenada f3 da linha e3? 1
0 vetor U pode ser representado na base F por Uf = (6)f1 + (7)f2 + (5)f3
0 delta 1
0 vetor V pode ser representado na base E por Ve = (0.0)e1 + (1.0)e2 + (2.0)e3

```

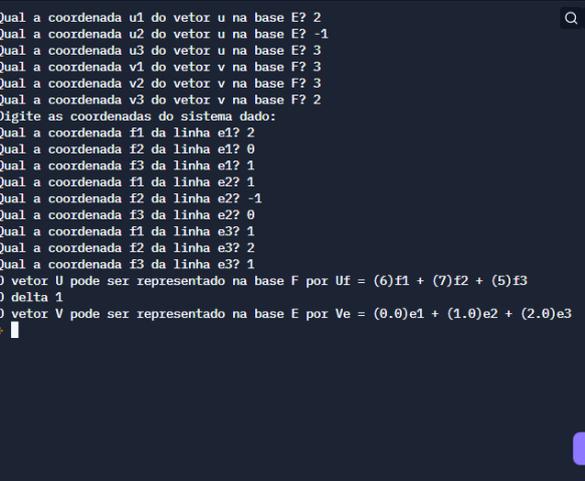
Figura 8 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para mudança de base.

Fonte: Autor, 2022.

```

29 uf1 = uf1a+uf2a+uf3a
30 uf2 = uf1b+uf2b+uf3b
31 uf3 = uf1c+uf2c+uf3c
32
33 print("0 vetor U pode ser representado na base F por Uf = ({}f1
+ ({}f2 + ({}f3)".format(uf1,uf2,uf3))
34
35 termo1 = e3f1 * e2f2
36 termo2 = e3f2 * e1f1
37 termo3 = e2f1 * e1f2
38 termo4 = e1f1 * e2f2
39 termo5 = e1f2 * e3f2
40 termo6 = e3f1 * e1f2
41
42 delta = termo1 + termo2 + termo3 + termo4 + termo5 + termo6
43
44 print("0 delta {}".format(delta))
45
46 #Cofator
47 coef1 = ((-1)**2)*((e2f2*e3f3)-(e3f2*e2f3))
48 coef2 = ((-1)**3)*((e1f2*e3f3)-(e3f2*e1f3))
49 coef3 = ((-1)**4)*((e1f2*e2f3)-(e2f2*e1f3))
50 coef4 = ((-1)**3)*((e2f1*e3f3)-(e3f1*e2f3))
51 coef5 = ((-1)**4)*((e1f1*e3f3)-(e3f1*e1f3))
52 coef6 = ((-1)**5)*((e1f1*e2f3)-(e2f1*e1f3))
53 coef7 = ((-1)**4)*((e2f1*e3f2)-(e3f1*e2f2))
54 coef8 = ((-1)**5)*((e1f1*e3f2)-(e3f1*e1f2))
55 coef9 = ((-1)**6)*((e1f1*e2f2)-(e2f1*e1f2))

```



```

Qual a coordenada u1 do vetor u na base E? 2
Qual a coordenada u2 do vetor u na base E? -1
Qual a coordenada u3 do vetor u na base E? 3
Qual a coordenada v1 do vetor v na base F? 3
Qual a coordenada v2 do vetor v na base F? 3
Qual a coordenada v3 do vetor v na base F? 2
Digite as coordenadas do sistema dado:
Qual a coordenada f1 da linha e1? 2
Qual a coordenada f2 da linha e1? 0
Qual a coordenada f3 da linha e1? 1
Qual a coordenada f1 da linha e2? 1
Qual a coordenada f2 da linha e2? -1
Qual a coordenada f3 da linha e2? 0
Qual a coordenada f1 da linha e3? 1
Qual a coordenada f2 da linha e3? 2
Qual a coordenada f3 da linha e3? 1
0 vetor U pode ser representado na base F por Uf = (6)f1 + (7)f2 + (5)f3
0 delta 1
0 vetor V pode ser representado na base E por Ve = (0.0)e1 + (1.0)e2 + (2.0)e3

```

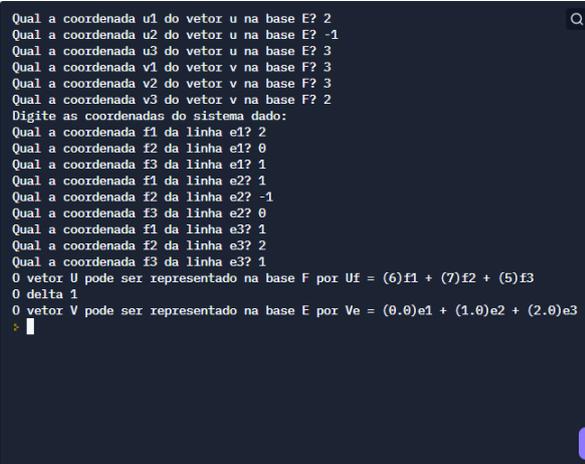
Figura 9 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para mudança de base.

Fonte: Autor, 2022.

```

57 m1 = 1/delta * coef1
58 m2 = 1/delta * coef2
59 m3 = 1/delta * coef3
60 m4 = 1/delta * coef4
61 m5 = 1/delta * coef5
62 m6 = 1/delta * coef6
63 m7 = 1/delta * coef7
64 m8 = 1/delta * coef8
65 m9 = 1/delta * coef9
66
67 dado1 = m1 * vetorv1
68 dado2 = m4 * vetorv2
69 dado3 = m7 * vetorv3
70 dado4 = m2 * vetorv1
71 dado5 = m5 * vetorv2
72 dado6 = m8 * vetorv3
73 dado7 = m3 * vetorv1
74 dado8 = m6 * vetorv2
75 dado9 = m9 * vetorv3
76
77 linha1 = dado1+dado2+dado3
78 linha2 = dado4+dado5+dado6
79 linha3 = dado7+dado8+dado9
80
81 print("O vetor V pode ser representado na base E por Ve = ({}e1
+ ({}e2 + ({}e3".format(linha1,linha2,linha3))

```



```

Qual a coordenada u1 do vetor u na base E? 2
Qual a coordenada u2 do vetor u na base E? -1
Qual a coordenada u3 do vetor u na base E? 3
Qual a coordenada v1 do vetor v na base F? 3
Qual a coordenada v2 do vetor v na base F? 3
Qual a coordenada v3 do vetor v na base F? 2
Digite as coordenadas do sistema dado:
Qual a coordenada f1 da linha e1? 2
Qual a coordenada f2 da linha e1? 0
Qual a coordenada f3 da linha e1? 1
Qual a coordenada f1 da linha e2? 1
Qual a coordenada f2 da linha e2? -1
Qual a coordenada f3 da linha e2? 0
Qual a coordenada f1 da linha e3? 1
Qual a coordenada f2 da linha e3? 2
Qual a coordenada f3 da linha e3? 1
O vetor u pode ser representado na base F por Uf = (6)f1 + (7)f2 + (5)f3
0 delta 1
O vetor v pode ser representado na base E por Ve = (0.0)e1 + (1.0)e2 + (2.0)e3

```

Figura 10 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para mudança de base.

Fonte: Autor, 2022.

3) Produto Escalar;

Realizando o cálculo do produto escalar, é possível obter um resultado numérico.

Verificando tal premissa, analisa-se o exemplo abaixo entre os vetores:

$$\vec{U} = 2\vec{i} - 3\vec{j} + \vec{k}$$

$$\vec{V} = -1\vec{i} - 2\vec{j} - 3\vec{k}$$

$$\vec{U} * \vec{V} = (2 * (-1)) + ((-3) * (-2)) + (1 * (-3))$$

$$\vec{U} * \vec{V} = -2 - 6 - 3$$

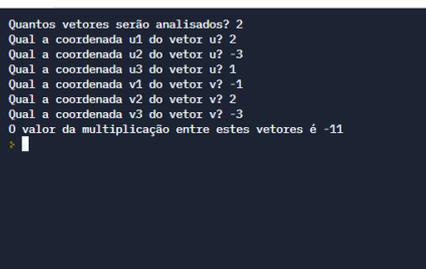
$$\vec{U} * \vec{V} = -11$$

Conforme vê-se na execução do código abaixo, em que é realizada a multiplicação entre dois vetores, a aplicação está coerente em relação ao cálculo feito:

```

1 qnt = int(input("Quantos vetores serão analisados? "))
2 if qnt == 2:
3     vetoru1 = int(input("Qual a coordenada u1 do vetor u? "))
4     vetoru2 = int(input("Qual a coordenada u2 do vetor u? "))
5     vetoru3 = int(input("Qual a coordenada u3 do vetor u? "))
6     vetorv1 = int(input("Qual a coordenada v1 do vetor v? "))
7     vetorv2 = int(input("Qual a coordenada v2 do vetor v? "))
8     vetorv3 = int(input("Qual a coordenada v3 do vetor v? "))
9
10    result1 = vetoru1*vetorv1
11    result2 = vetoru2*vetorv2
12    result3 = vetoru3*vetorv3
13
14    total = result1 + result2 + result3
15
16    print("O valor da multiplicação entre estes vetores é {}".format(total))
17

```



```

Quantos vetores serão analisados? 2
Qual a coordenada u1 do vetor u? 2
Qual a coordenada u2 do vetor u? -3
Qual a coordenada u3 do vetor u? 1
Qual a coordenada v1 do vetor v? -1
Qual a coordenada v2 do vetor v? -2
Qual a coordenada v3 do vetor v? -3
O valor da multiplicação entre estes vetores é -11

```

Figura 11 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para produto escalar entre dois vetores.

Fonte: Autor, 2022.

Já em relação à multiplicação entre três vetores:

$$\vec{U} = 2\vec{i} + 3\vec{j} + 9\vec{k}$$

$$\vec{V} = 6\vec{i} + 7\vec{j} + 2\vec{k}$$

$$\vec{W} = 3\vec{i} + 7\vec{j} + 4\vec{k}$$

$$\vec{U} * \vec{V} = (2 * 6) + (3 * 7) + (9 * 2)$$

$$\vec{U} * \vec{V} = 12 + 21 + 18$$

$$\vec{U} * \vec{V} = 51$$

$$\vec{U} * \vec{W} = (2 * 3) + (3 * 7) + (9 * 4)$$

$$\vec{U} * \vec{W} = 6 + 21 + 36$$

$$\vec{U} * \vec{W} = 63$$

```

18 if qnt == 3:
19     vetoru1 = int(input("Qual a coordenada u1 do vetor u? "))
20     vetoru2 = int(input("Qual a coordenada u2 do vetor u? "))
21     vetoru3 = int(input("Qual a coordenada u3 do vetor u? "))
22     vetov1 = int(input("Qual a coordenada v1 do vetor v? "))
23     vetov2 = int(input("Qual a coordenada v2 do vetor v? "))
24     vetov3 = int(input("Qual a coordenada v3 do vetor v? "))
25     vetow1 = int(input("Qual a coordenada w1 do vetor w? "))
26     vetow2 = int(input("Qual a coordenada w2 do vetor w? "))
27     vetow3 = int(input("Qual a coordenada w3 do vetor w? "))
28
29     result1 = vetoru1*vetov1
30     result2 = vetoru2*vetov2
31     result3 = vetoru3*vetov3
32
33     rt1 = result1 * vetow1
34     rt2 = result2 * vetow2
35     rt3 = result3 * vetow3
36
37     total = rt1 + rt2 + rt3
38
39     print("O valor da multiplicação entre estes vetores é {}".format(total))

```

```

Quantos vetores serão analisados? 3
Qual a coordenada u1 do vetor u? 2
Qual a coordenada u2 do vetor u? 3
Qual a coordenada u3 do vetor u? 9
Qual a coordenada v1 do vetor v? 6
Qual a coordenada v2 do vetor v? 7
Qual a coordenada v3 do vetor v? 2
Qual a coordenada w1 do vetor w? 3
Qual a coordenada w2 do vetor w? 7
Qual a coordenada w3 do vetor w? 4
O valor da multiplicação entre estes vetores é 255

```

Figura 12 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para produto escalar entre três vetores.

Fonte: Autor, 2022.

4) Produto Vetorial;

Realizando o cálculo do produto vetorial, é possível obter um resultado no formato de vetor. Verificando tal premissa, analisa-se o exemplo abaixo entre os vetores:

$$\vec{U} = 2\vec{i} - 3\vec{j} + \vec{k}$$

$$\vec{V} = -1\vec{i} + 2\vec{j} - 3\vec{k}.$$

$$\vec{U} \wedge \vec{V} = \begin{bmatrix} \vec{i} & \vec{j} & \vec{k} & \vec{i} & \vec{j} \\ 2 & -3 & 1 & 2 & -3 \\ -1 & 2 & -3 & -1 & 2 \end{bmatrix} = 9\vec{i} - 1\vec{j} + 4\vec{k} - 3\vec{k} - 2\vec{i} + 6\vec{j}$$

$$\vec{U} \wedge \vec{V} = 7\vec{i} + 5\vec{j} + \vec{k}$$

Conforme vê-se na execução do código abaixo, em que é realizada a multiplicação entre dois vetores, é possível visualizar no programa a execução do exemplo dado:

```

1  vetoru1 = int(input("Qual a coordenada u1 do vetor u? "))
2  vetoru2 = int(input("Qual a coordenada u2 do vetor u? "))
3  vetoru3 = int(input("Qual a coordenada u3 do vetor u? "))
4  vetorv1 = int(input("Qual a coordenada v1 do vetor v? "))
5  vetorv2 = int(input("Qual a coordenada v2 do vetor v? "))
6  vetorv3 = int(input("Qual a coordenada v3 do vetor v? "))
7
8  i = ((vetoru2 * vetorv3)-(vetoru3 * vetorv2))
9  j = ((vetoru3 * vetorv1)-(vetoru1 * vetorv3))
10 k = ((vetoru1 * vetorv2)-(vetoru2 * vetorv1))
11
12 print("O resultado da multiplicação é ({}i + ({}j + ({}k)".format(i,j,k))
--

```

Qual a coordenada u1 do vetor u? 2
Qual a coordenada u2 do vetor u? -3
Qual a coordenada u3 do vetor u? 1
Qual a coordenada v1 do vetor v? -1
Qual a coordenada v2 do vetor v? 2
Qual a coordenada v3 do vetor v? -3
O resultado da multiplicação é (7)i + (5)j + (1)k

Figura 13 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para produto vetorial.

Fonte: Autor, 2022.

5) Produto Misto;

Realizando o cálculo do produto misto, é possível obter um resultado numérico.

Verificando tal premissa, analisa-se o exemplo abaixo entre os vetores:

$$\vec{U} = \vec{i} - \vec{j} + 3\vec{k}$$

$$\vec{V} = -\vec{i} + 2\vec{j} + 5\vec{k}$$

$$\vec{W} = -2\vec{i} + \vec{j} - \vec{k}$$

$$[\vec{U}, \vec{V}, \vec{W}] = \begin{vmatrix} \vec{U} \\ \vec{V} \\ \vec{W} \end{vmatrix} = \begin{bmatrix} 1 & -1 & 3 & 1 & -1 \\ -1 & 2 & 5 & -1 & 2 \\ -2 & 1 & -1 & -2 & 1 \end{bmatrix} = -2 + 10 - 3 + 12 - 5 + 1$$

$$[\vec{U}, \vec{V}, \vec{W}] = 23 - 10 = 13$$

```

1  vetoru1 = int(input("Qual a coordenada u1 do vetor u? "))
2  vetoru2 = int(input("Qual a coordenada u2 do vetor u? "))
3  vetoru3 = int(input("Qual a coordenada u3 do vetor u? "))
4  vetorv1 = int(input("Qual a coordenada v1 do vetor v? "))
5  vetorv2 = int(input("Qual a coordenada v2 do vetor v? "))
6  vetorv3 = int(input("Qual a coordenada v3 do vetor v? "))
7  vetorw1 = int(input("Qual a coordenada w1 do vetor w? "))
8  vetorw2 = int(input("Qual a coordenada w2 do vetor w? "))
9  vetorw3 = int(input("Qual a coordenada w3 do vetor w? "))
10
11 linha = (vetoru1*vetorv2*vetorw3) + (vetoru2*vetorv3*vetorw1) +
(vetoru3*vetorv1*vetorw2) - (vetoru3*vetorv2*vetorw1) -
(vetoru1*vetorv3*vetorw2) - (vetoru2*vetorv1*vetorw3)
12
13 print("O valor do produto misto entre estes vetores é {}".format(linha))
--

```

Qual a coordenada u1 do vetor u? 1
Qual a coordenada u2 do vetor u? -1
Qual a coordenada u3 do vetor u? 3
Qual a coordenada v1 do vetor v? -1
Qual a coordenada v2 do vetor v? 2
Qual a coordenada v3 do vetor v? 5
Qual a coordenada w1 do vetor w? -2
Qual a coordenada w2 do vetor w? 1
Qual a coordenada w3 do vetor w? -1
O valor do produto misto entre estes vetores é 13

Figura 14 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para produto misto.

Fonte: Autor, 2022.

6) Retas;

No âmbito das equações de retas, têm-se três possíveis resultados, são eles: as equações vetoriais, paramétricas ou simétricas. Exemplificando cada um dos casos, deseja-se, por meio das coordenadas de um vetor e de um ponto, ou de dois pontos conhecidos, escrever as equações da reta.

Supondo o caso de o enunciado ter proposto um vetor \vec{V} de coordenadas $(-2,-7,1)$ e um ponto A $(3,5,-2)$, determinam-se as equações da reta:

Em relação à equação vetorial, é possível obtê-la a partir da substituição dos valores conhecidos das coordenadas do ponto e do vetor.

$$t: P = A + \alpha \vec{V}$$

$$t: (x, y, z) = (3, 5, -2) + \alpha(-2, -7, 1)$$

Já a equação paramétrica busca separar os termos da equação vetorial em relação às coordenadas do ponto P, conforme vê-se abaixo:

$$t: (x, y, z) = (3, 5, -2) + \alpha(-2, -7, 1)$$

$$t: \begin{cases} x = 3 - 2\alpha \\ y = 5 - 7\alpha \\ z = -2 + \alpha \end{cases}$$

Por fim, a equação simétrica é dada por meio do processo de separação da equação paramétrica, deixando o coeficiente de cada uma das equações em evidência. De acordo com a teoria, os três coeficientes possuem o mesmo valor, logo, ao igualar os coeficientes, é possível igualar as equações isoladas, tal que:

$$t: \begin{cases} x = 3 - 2\alpha \\ y = 5 - 7\alpha \\ z = -2 + \alpha \end{cases}$$

$$t: \begin{cases} \alpha = -\frac{x+3}{2} \\ \alpha = -\frac{y+5}{7} \\ \alpha = z+2 \end{cases}$$

$$t: \left\{ -\frac{x+3}{2} = \frac{y+5}{7} = z+2 \right\}$$

Testando esse procedimento na aplicação realizada, é possível conferir os resultados e verificar a precisão da calculadora nas imagens abaixo, as quais demonstram, respectivamente, o cálculo da equação vetorial, paramétrica e simétrica.

```

1 er = int(input("São conhecidas as coordenadas do ponto e do vetor
diretor? "))
2
3 if er == 1:
4     pont1r = int(input("Qual a coordenada p1 do ponto? "))
5     pont2r = int(input("Qual a coordenada p2 do ponto? "))
6     pont3r = int(input("Qual a coordenada p3 do ponto? "))
7     vetor1r = int(input("Qual a coordenada u1 do vetor diretor u? "))
8     vetor2r = int(input("Qual a coordenada u2 do vetor diretor u? "))
9     vetor3r = int(input("Qual a coordenada u3 do vetor diretor u? "))
10
11 e = int(input("Qual equação se deseja calcular? "))
12 if e == 1:
13     print("A equação vetorial da reta é t: X = {} +
({})a".format((pont1r,pont2r,pont3r),(vetor1r,vetor2r,vetor3r)))
14
15 if e == 2:
16     print("A equação paramétrica da reta é: ")
17     print("r: x = {} + ({}a".format((pont1r),(vetor1r)))
18     print("r: y = {} + ({}a".format((pont2r),(vetor2r)))
19     print("r: z = {} + ({}a".format((pont3r),(vetor3r)))
20
21 if e == 3:
22     print("A equação simétrica da reta é: ")
23     print("t: x - ({})/({}) = y - ({})/({}) = z -
({})/({})".format((pont1r),(vetor1r),(pont2r),(vetor2r),(pont3r),
({})/({}))

```

São conhecidas as coordenadas do ponto e do vetor diretor? 1
Qual a coordenada p1 do ponto? 3
Qual a coordenada p2 do ponto? 5
Qual a coordenada p3 do ponto? -2
Qual a coordenada u1 do vetor diretor u? -2
Qual a coordenada u2 do vetor diretor u? -7
Qual a coordenada u3 do vetor diretor u? 1
Qual equação se deseja calcular? 1
A equação vetorial da reta é t: X = (3, 5, -2) + ((-2, -7, 1))a

Figura 15 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para a equação da reta vetorial.

Fonte: Autor, 2022.

```

1 er = int(input("São conhecidas as coordenadas do ponto e do vetor
diretor? "))
2
3 if er == 1:
4     pont1r = int(input("Qual a coordenada p1 do ponto? "))
5     pont2r = int(input("Qual a coordenada p2 do ponto? "))
6     pont3r = int(input("Qual a coordenada p3 do ponto? "))
7     vetor1r = int(input("Qual a coordenada u1 do vetor diretor u? "))
8     vetor2r = int(input("Qual a coordenada u2 do vetor diretor u? "))
9     vetor3r = int(input("Qual a coordenada u3 do vetor diretor u? "))
10
11 e = int(input("Qual equação se deseja calcular? "))
12 if e == 1:
13     print("A equação vetorial da reta é t: X = {} +
({})a".format((pont1r,pont2r,pont3r),(vetor1r,vetor2r,vetor3r)))
14
15 if e == 2:
16     print("A equação paramétrica da reta é: ")
17     print("r: x = {} + ({}a".format((pont1r),(vetor1r)))
18     print("r: y = {} + ({}a".format((pont2r),(vetor2r)))
19     print("r: z = {} + ({}a".format((pont3r),(vetor3r)))
20
21 if e == 3:
22     print("A equação simétrica da reta é: ")
23     print("t: x - ({})/({}) = y - ({})/({}) = z -
({})/({})".format((pont1r),(vetor1r),(pont2r),(vetor2r),(pont3r),
({})/({}))

```

São conhecidas as coordenadas do ponto e do vetor diretor? 1
Qual a coordenada p1 do ponto? 3
Qual a coordenada p2 do ponto? 5
Qual a coordenada p3 do ponto? -2
Qual a coordenada u1 do vetor diretor u? -2
Qual a coordenada u2 do vetor diretor u? -7
Qual a coordenada u3 do vetor diretor u? 1
Qual equação se deseja calcular? 2
A equação paramétrica da reta é:
r: x = 3 + (-2)a
r: y = 5 + (-7)a
r: z = -2 + (1)a

Figura 16 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para a equação da reta paramétrica.

Fonte: Autor, 2022.

```

1  ex = int(input("São conhecidas as coordenadas do ponto e do vetor
diretor? "))
2
3  if ex == 1:
4      pont1r = int(input("Qual a coordenada p1 do ponto? "))
5      pont2r = int(input("Qual a coordenada p2 do ponto? "))
6      pont3r = int(input("Qual a coordenada p3 do ponto? "))
7      vetor1r = int(input("Qual a coordenada u1 do vetor diretor u? "))
8      vetor2r = int(input("Qual a coordenada u2 do vetor diretor u? "))
9      vetor3r = int(input("Qual a coordenada u3 do vetor diretor u? "))
10
11     e = int(input("Qual equação se deseja calcular? "))
12     if e == 1:
13         print("A equação vetorial da reta é t: X = {} +
({})a".format((pont1r,pont2r,pont3r),(vetor1r,vetor2r,vetor3r)))
14
15     if e == 2:
16         print("A equação paramétrica da reta é: ")
17         print("r: x = {} + ({}a".format((pont1r),(vetor1r)))
18         print("r: y = {} + ({}a".format((pont2r),(vetor2r)))
19         print("r: z = {} + ({}a".format((pont3r),(vetor3r)))
20
21     if e == 3:
22         print("A equação simétrica da reta é: ")
23         print("t: x - ({})/( {}) = y - ({})/( {}) = z -
({})/( {})" .format((pont1r),(vetor1r),(pont2r),(vetor2r),(pont3r),
(vetor3r)))

```

São conhecidas as coordenadas do ponto e do vetor diretor? 1
Qual a coordenada p1 do ponto? 3
Qual a coordenada p2 do ponto? 5
Qual a coordenada p3 do ponto? -2
Qual a coordenada u1 do vetor diretor u? -2
Qual a coordenada u2 do vetor diretor u? -7
Qual a coordenada u3 do vetor diretor u? 1
Qual equação se deseja calcular? 3
A equação simétrica da reta é:
t: x - (3)/(-2) = y - (5)/(-7) = z - (-2)/(1)

Figura 17 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para a equação da reta simétrica.

Fonte: Autor, 2022.

No entanto, caso sejam dados dois pontos, A e B, o vetor \vec{V} será igual ao vetor \overrightarrow{AB} , que é calculado subtraindo as coordenadas do ponto inicial em relação ao ponto final, de acordo com o cálculo abaixo:

Supondo os pontos A = (3,5,-2) e B = (1,-2,-1):

$$\overrightarrow{AB} = B - A = (1 - 3, -2 - 5, -1 - (-2)) = (-2, -7, 1)$$

Após calcular o valor de \overrightarrow{AB} , a calculadora repete o mesmo processo visto no caso anterior.

7) Retas – Posição Relativa;

As retas, quando analisadas entre si e em relação à um plano, podem ter características diferentes, sendo: concorrentes, paralelas ou reversas, de acordo com o conteúdo programático visto anteriormente.

Para verificar a posição relativa entre as retas, são propostos 3 casos com valores de vetores e pontos diferentes, visando analisar o comportamento de cada uma das situações.

Dadas duas retas, $r: (x, y, z) = (2, 7, 1) + \alpha(1, -1, 1)$ e $s: (x, y, z) = (2, 1, 0) + \beta(1, 4, -6)$, é possível obter os valores dos pontos (2,7,1) e (2,1,0) das retas, respectivamente, r e s.

Encontrando um vetor \overrightarrow{AB} entre elas, calcula-se:

$$\overrightarrow{AB} = B - A = (2 - 2, 1 - 7, 0 - 1) = (0, -6, -1)$$

Em seguida, realizando o cálculo do determinante entre os vetores das retas e o vetor \overrightarrow{AB} obtido, é possível encontrar um valor menor que zero, o que faz com que a reta seja reversa, conforme pode-se ver abaixo:

$$\begin{vmatrix} 1 & -1 & 1 & 1 & -1 \\ 1 & 4 & -6 & 1 & 4 \\ 0 & -6 & -1 & 0 & -6 \end{vmatrix} = -4 + 0 - 6 - 0 - 36 - 1 = -47$$

Em outro caso, dadas duas retas, $r: (x, y, z) = (2, 7, 5) + \alpha(-2, 6, -4)$ e $s: (x, y, z) = (1, 2, 1) + \beta(1, -3, 2)$, é possível obter os valores dos pontos $(2, 7, 5)$ e $(1, 2, 1)$ das retas, respectivamente, r e s.

Encontrando um vetor \overrightarrow{AB} entre elas, calcula-se:

$$\overrightarrow{AB} = B - A = (1 - 2, 2 - 7, 1 - 5) = (-1, -5, -4)$$

Em seguida, realizando o cálculo do determinante entre os dois vetores das retas e o vetor \overrightarrow{AB} obtido, é possível encontrar um valor igual a zero, o que torna necessária a verificação em relação ao paralelismo.

$$\begin{vmatrix} -2 & 6 & -4 & -2 & 6 \\ 1 & -3 & 2 & 1 & -3 \\ -1 & -5 & -4 & -1 & -5 \end{vmatrix} = -24 - 12 + 20 + 12 - 20 + 24 = 0$$

Realizando a divisão entre as coordenadas dos vetores \vec{U} e \vec{V} , é possível obter que:

$$\frac{\vec{U}}{\vec{V}} = \frac{-2}{1} = \frac{6}{-3} = \frac{-4}{2} = -2$$

Portanto, são paralelos. Dessa forma, as retas são paralelas.

Por fim, dadas duas retas, $r: (x, y, z) = (1, 1, -1) + \alpha(-1, 3, 2)$ e $s: (x, y, z) = (3, 2, 1) + \beta(-2, -1, -2)$, é possível obter os valores dos pontos $(1, 1, -1)$ e $(3, 2, 1)$ das retas, respectivamente, r e s.

Encontrando um vetor \overrightarrow{AB} entre elas, calcula-se:

$$\overrightarrow{AB} = B - A = (3 - 1, 2 - 1, 1 - (-1)) = (2, 1, 2)$$

Em seguida, realizando o cálculo do determinante entre os dois vetores das retas e o vetor \overrightarrow{AB} obtido, é possível encontrar um valor igual a zero, o que torna necessária a verificação em relação ao paralelismo.

$$\begin{vmatrix} -1 & 3 & 2 \\ -2 & -1 & -2 \\ 2 & 1 & 2 \end{vmatrix} \begin{vmatrix} -1 & 3 \\ -2 & -1 \\ 2 & 1 \end{vmatrix} = 2 - 12 - 4 + 4 - 2 + 12 = 0$$

Realizando a divisão entre as coordenadas dos vetores \vec{U} e \vec{V} , é possível obter que:

$$\frac{\vec{U}}{\vec{V}} = \frac{-1}{-2} \neq \frac{3}{-1} \neq \frac{2}{-2} = -2$$

Portanto, não são paralelos. Dessa forma, as retas são concorrentes.

Verificando estes cálculos na aplicação realizada, é possível obter a confirmação da veracidade dos resultados obtidos para as retas reversas, paralelas e concorrentes, respectivamente.

```

1 pont1ra = int(input("Qual a coordenada p1 do ponto A? "))
2 pont2ra = int(input("Qual a coordenada p2 do ponto A? "))
3 pont3ra = int(input("Qual a coordenada p3 do ponto A? "))
4 vetoru1r = int(input("Qual a coordenada u1 do vetor diretor u? "))
5 vetoru2r = int(input("Qual a coordenada u2 do vetor diretor u? "))
6 vetoru3r = int(input("Qual a coordenada u3 do vetor diretor u? "))
7 pont1rb = int(input("Qual a coordenada p1 do ponto B? "))
8 pont2rb = int(input("Qual a coordenada p2 do ponto B? "))
9 pont3rb = int(input("Qual a coordenada p3 do ponto B? "))
10 vetorv1r = int(input("Qual a coordenada u1 do vetor diretor v? "))
11 vetorv2r = int(input("Qual a coordenada u2 do vetor diretor v? "))
12 vetorv3r = int(input("Qual a coordenada u3 do vetor diretor v? "))
13
14 paba = pont1rb - pont1ra
15 pabb = pont2rb - pont2ra
16 pabc = pont3rb - pont3ra
17
18 linha = (vetoru1r*vetorv2r*pabc) + (vetoru2r*vetorv3r*paba) +
          (vetoru3r*vetorv1r*pabb) - (vetoru3r*vetorv2r*paba) -
          (vetoru1r*vetorv3r*pabb) - (vetoru2r*vetorv1r*pabc)
19
20 if linha == 0:
21
22     if vetoru1r/vetorv1r == vetoru2r/vetorv2r == vetoru3r/vetorv3r:
23         print("As retas são paralelas")
24     else:

```

```

Qual a coordenada p1 do ponto A? 2
Qual a coordenada p2 do ponto A? 7
Qual a coordenada p3 do ponto A? 1
Qual a coordenada u1 do vetor diretor u? 1
Qual a coordenada u2 do vetor diretor u? -1
Qual a coordenada u3 do vetor diretor u? 1
Qual a coordenada p1 do ponto B? 2
Qual a coordenada p2 do ponto B? 1
Qual a coordenada p3 do ponto B? 0
Qual a coordenada u1 do vetor diretor v? 1
Qual a coordenada u2 do vetor diretor v? 4
Qual a coordenada u3 do vetor diretor v? -6
As retas são reversas

```

Figura 18 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para verificação das retas reversas.

Fonte: Autor, 2022.

```

1 pont1ra = int(input("Qual a coordenada p1 do ponto A? "))
2 pont2ra = int(input("Qual a coordenada p2 do ponto A? "))
3 pont3ra = int(input("Qual a coordenada p3 do ponto A? "))
4 vetoru1r = int(input("Qual a coordenada u1 do vetor diretor u? "))
5 vetoru2r = int(input("Qual a coordenada u2 do vetor diretor u? "))
6 vetoru3r = int(input("Qual a coordenada u3 do vetor diretor u? "))
7 pont1rb = int(input("Qual a coordenada p1 do ponto B? "))
8 pont2rb = int(input("Qual a coordenada p2 do ponto B? "))
9 pont3rb = int(input("Qual a coordenada p3 do ponto B? "))
10 vetoru1r = int(input("Qual a coordenada u1 do vetor diretor v? "))
11 vetoru2r = int(input("Qual a coordenada u2 do vetor diretor v? "))
12 vetoru3r = int(input("Qual a coordenada u3 do vetor diretor v? "))
13
14 paba = pont1rb - pont1ra
15 pabb = pont2rb - pont2ra
16 pabc = pont3rb - pont3ra
17
18 linha = (vetoru1r*vetoru2r*pabc) + (vetoru2r*vetoru3r*paba) +
(vetoru3r*vetoru1r*pabb) - (vetoru3r*vetoru2r*paba) -
(vetoru1r*vetoru3r*pabb) - (vetoru2r*vetoru1r*pabc)
19
20 if linha == 0:
21
22     if vetoru1r/vetoru1r == vetoru2r/vetoru2r == vetoru3r/vetoru3r:
23         print("As retas são paralelas")
24     else:

```

```

Qual a coordenada p1 do ponto A? 2
Qual a coordenada p2 do ponto A? 7
Qual a coordenada p3 do ponto A? 5
Qual a coordenada u1 do vetor diretor u? -2
Qual a coordenada u2 do vetor diretor u? 6
Qual a coordenada u3 do vetor diretor u? -4
Qual a coordenada p1 do ponto B? 1
Qual a coordenada p2 do ponto B? 2
Qual a coordenada p3 do ponto B? 1
Qual a coordenada u1 do vetor diretor v? 1
Qual a coordenada u2 do vetor diretor v? -3
Qual a coordenada u3 do vetor diretor v? 2
As retas são paralelas

```

Figura 19 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para a verificação das retas paralelas.

Fonte: Autor, 2022.

```

1 pont1ra = int(input("Qual a coordenada p1 do ponto A? "))
2 pont2ra = int(input("Qual a coordenada p2 do ponto A? "))
3 pont3ra = int(input("Qual a coordenada p3 do ponto A? "))
4 vetoru1r = int(input("Qual a coordenada u1 do vetor diretor u? "))
5 vetoru2r = int(input("Qual a coordenada u2 do vetor diretor u? "))
6 vetoru3r = int(input("Qual a coordenada u3 do vetor diretor u? "))
7 pont1rb = int(input("Qual a coordenada p1 do ponto B? "))
8 pont2rb = int(input("Qual a coordenada p2 do ponto B? "))
9 pont3rb = int(input("Qual a coordenada p3 do ponto B? "))
10 vetoru1r = int(input("Qual a coordenada u1 do vetor diretor v? "))
11 vetoru2r = int(input("Qual a coordenada u2 do vetor diretor v? "))
12 vetoru3r = int(input("Qual a coordenada u3 do vetor diretor v? "))
13
14 paba = pont1rb - pont1ra
15 pabb = pont2rb - pont2ra
16 pabc = pont3rb - pont3ra
17
18 linha = (vetoru1r*vetoru2r*pabc) + (vetoru2r*vetoru3r*paba) +
(vetoru3r*vetoru1r*pabb) - (vetoru3r*vetoru2r*paba) -
(vetoru1r*vetoru3r*pabb) - (vetoru2r*vetoru1r*pabc)
19
20 if linha == 0:
21
22     if vetoru1r/vetoru1r == vetoru2r/vetoru2r == vetoru3r/vetoru3r:
23         print("As retas são paralelas")
24     else:

```

```

Qual a coordenada p1 do ponto A? 1
Qual a coordenada p2 do ponto A? 1
Qual a coordenada p3 do ponto A? -1
Qual a coordenada u1 do vetor diretor u? -1
Qual a coordenada u2 do vetor diretor u? 3
Qual a coordenada u3 do vetor diretor u? 2
Qual a coordenada p1 do ponto B? 3
Qual a coordenada p2 do ponto B? 2
Qual a coordenada p3 do ponto B? 1
Qual a coordenada u1 do vetor diretor v? -2
Qual a coordenada u2 do vetor diretor v? -1
Qual a coordenada u3 do vetor diretor v? -2
As retas são concorrentes

```

Figura 20 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para a verificação das retas concorrentes.

Fonte: Autor, 2022.

8) Planos;

Assim como as retas, as equações de plano têm três possíveis resultados, são eles: as equações vetoriais, paramétricas ou gerais/cartesianas. Exemplificando cada um dos casos, deseja-se, por meio das coordenadas de dois vetores e de um ponto, ou de um vetor e dois pontos conhecidos, escrever as equações do plano.

Supondo o caso de o enunciado ter proposto um vetor \vec{U} de coordenadas (2,-3,7), um vetor \vec{V} de coordenadas (-3,2,5) e um ponto A (1,-3,4), determinam-se as equações do plano:

Em relação à equação vetorial, é possível obtê-la a partir da substituição dos valores conhecidos das coordenadas do ponto e dos vetores.

$$t: P = A + \alpha \vec{U} + \beta \vec{V}$$

$$t: (x, y, z) = (1, -3, 4) + \alpha(2, -3, 7) + \beta(-3, 2, 5)$$

Já a equação paramétrica busca separar os termos da equação vetorial em relação às coordenadas do ponto P, conforme vê-se abaixo:

$$t: (x, y, z) = (1, -3, 4) + \alpha(2, -3, 7) + \beta(-3, 2, 5)$$

$$t: \begin{cases} x = 1 + 2\alpha - 3\beta \\ y = -3 - 3\alpha + 2\beta \\ z = 4 + 7\alpha + 5\beta \end{cases}$$

No entanto, o procedimento para a equação geral ou cartesiana do plano diverge em relação ao método proposto para as retas. Utilizando os mesmos dois vetores conhecidos e o mesmo ponto, implementa-se um novo ponto P, cujas coordenadas são (x, y, z) e, por meio do ponto A dado, calcula-se um vetor entre eles.

$$\overrightarrow{AX} = X - A = (x, y, z) - (1, -3, 4) = (x - 1, y + 3, z - 4) \text{ e determinante} = 0.$$

Calcula-se então o determinante entre este vetor \overrightarrow{AX} obtido e os dois pontos dados anteriormente.

$$\begin{vmatrix} x-1 & y+3 & z-4 \\ 2 & -3 & 7 \\ -3 & 2 & 5 \end{vmatrix} \begin{vmatrix} x-1 & y+3 \\ 2 & -3 \\ -3 & 2 \end{vmatrix} = 0$$

$$-15(x - 1) - 21(y + 3) + 4(z - 4) - 9(z - 4) - 14(x - 1) - 10(y + 3) = 0$$

Tal resultado, ao ser simplificado, dará uma equação geral do plano igual à $-29x - 31y - 5z - 44 = 0$.

Testando esse procedimento na aplicação realizada, é possível conferir os resultados e verificar a precisão da calculadora nas imagens abaixo, as quais demonstram, respectivamente, o cálculo da equação vetorial, paramétrica e geral ou cartesiana do plano.

```

1 e = int(input("Qual equação se deseja calcular? "))
2 if e == 1:
3     pont1 = int(input("Qual a coordenada p1 do ponto? "))
4     pont2 = int(input("Qual a coordenada p2 do ponto? "))
5     pont3 = int(input("Qual a coordenada p3 do ponto? "))
6     vetor1 = int(input("Qual a coordenada u1 do vetor u? "))
7     vetor2 = int(input("Qual a coordenada u2 do vetor u? "))
8     vetor3 = int(input("Qual a coordenada u3 do vetor u? "))
9     vetorv1 = int(input("Qual a coordenada v1 do vetor v? "))
10    vetorv2 = int(input("Qual a coordenada v2 do vetor v? "))
11    vetorv3 = int(input("Qual a coordenada v3 do vetor v? "))
12    print("A equação vetorial do plano é p = {} + α({} +
    β({})).format((pont1,pont2,pont3),(vetor1,vetor2,vetor3),
    (vetorv1,vetorv2,vetorv3))
13
14 if e == 2:
15     pont1 = int(input("Qual a coordenada p1 do ponto? "))
16     pont2 = int(input("Qual a coordenada p2 do ponto? "))
17     pont3 = int(input("Qual a coordenada p3 do ponto? "))
18     vetor1 = int(input("Qual a coordenada u1 do vetor u? "))
19     vetor2 = int(input("Qual a coordenada u2 do vetor u? "))
20     vetor3 = int(input("Qual a coordenada u3 do vetor u? "))
21     vetorv1 = int(input("Qual a coordenada v1 do vetor v? "))
22     vetorv2 = int(input("Qual a coordenada v2 do vetor v? "))
23     vetorv3 = int(input("Qual a coordenada v3 do vetor v? "))
24     print("A equação paramétrica do plano é: ")

```

```

Qual equação se deseja calcular? 1
Qual a coordenada p1 do ponto? 1
Qual a coordenada p2 do ponto? -3
Qual a coordenada p3 do ponto? 4
Qual a coordenada u1 do vetor u? 2
Qual a coordenada u2 do vetor u? -3
Qual a coordenada u3 do vetor u? 7
Qual a coordenada v1 do vetor v? -3
Qual a coordenada v2 do vetor v? 2
Qual a coordenada v3 do vetor v? 5
A equação vetorial do plano é p = (1, -3, 4) + α((2, -3, 7)) + β((-3, 2, 5))

```

Figura 21 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para a equação do plano vetorial.

Fonte: Autor, 2022.

```

1 e = int(input("Qual equação se deseja calcular? "))
2 if e == 1:
3     pont1 = int(input("Qual a coordenada p1 do ponto? "))
4     pont2 = int(input("Qual a coordenada p2 do ponto? "))
5     pont3 = int(input("Qual a coordenada p3 do ponto? "))
6     vetor1 = int(input("Qual a coordenada u1 do vetor u? "))
7     vetor2 = int(input("Qual a coordenada u2 do vetor u? "))
8     vetor3 = int(input("Qual a coordenada u3 do vetor u? "))
9     vetorv1 = int(input("Qual a coordenada v1 do vetor v? "))
10    vetorv2 = int(input("Qual a coordenada v2 do vetor v? "))
11    vetorv3 = int(input("Qual a coordenada v3 do vetor v? "))
12    print("A equação vetorial do plano é p = {} + α({} +
    β({})).format((pont1,pont2,pont3),(vetor1,vetor2,vetor3),
    (vetorv1,vetorv2,vetorv3))
13
14 if e == 2:
15     pont1 = int(input("Qual a coordenada p1 do ponto? "))
16     pont2 = int(input("Qual a coordenada p2 do ponto? "))
17     pont3 = int(input("Qual a coordenada p3 do ponto? "))
18     vetor1 = int(input("Qual a coordenada u1 do vetor u? "))
19     vetor2 = int(input("Qual a coordenada u2 do vetor u? "))
20     vetor3 = int(input("Qual a coordenada u3 do vetor u? "))
21     vetorv1 = int(input("Qual a coordenada v1 do vetor v? "))
22     vetorv2 = int(input("Qual a coordenada v2 do vetor v? "))
23     vetorv3 = int(input("Qual a coordenada v3 do vetor v? "))
24     print("A equação paramétrica do plano é: ")

```

```

Qual equação se deseja calcular? 2
Qual a coordenada p1 do ponto? 1
Qual a coordenada p2 do ponto? -3
Qual a coordenada p3 do ponto? 4
Qual a coordenada u1 do vetor u? 2
Qual a coordenada u2 do vetor u? -3
Qual a coordenada u3 do vetor u? 7
Qual a coordenada v1 do vetor v? -3
Qual a coordenada v2 do vetor v? 2
Qual a coordenada v3 do vetor v? 5
A equação paramétrica do plano é:
r: x = 1 + α(2) + β(-3)
r: y = -3 + α(-3) + β(2)
r: z = 4 + α(7) + β(5)

```

Figura 22 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para a equação do plano paramétrica.

Fonte: Autor, 2022.

```

1 e = int(input("Qual equação se deseja calcular? "))
2 if e == 1:
3     pont1 = int(input("Qual a coordenada p1 do ponto? "))
4     pont2 = int(input("Qual a coordenada p2 do ponto? "))
5     pont3 = int(input("Qual a coordenada p3 do ponto? "))
6     vetor1 = int(input("Qual a coordenada u1 do vetor u? "))
7     vetor2 = int(input("Qual a coordenada u2 do vetor u? "))
8     vetor3 = int(input("Qual a coordenada u3 do vetor u? "))
9     vetorv1 = int(input("Qual a coordenada v1 do vetor v? "))
10    vetorv2 = int(input("Qual a coordenada v2 do vetor v? "))
11    vetorv3 = int(input("Qual a coordenada v3 do vetor v? "))
12    print("A equação vetorial do plano é p = {} + α({} +
    β({})).format((pont1,pont2,pont3),(vetor1,vetor2,vetor3),
    (vetorv1,vetorv2,vetorv3))
13
14 if e == 2:
15     pont1 = int(input("Qual a coordenada p1 do ponto? "))
16     pont2 = int(input("Qual a coordenada p2 do ponto? "))
17     pont3 = int(input("Qual a coordenada p3 do ponto? "))
18     vetor1 = int(input("Qual a coordenada u1 do vetor u? "))
19     vetor2 = int(input("Qual a coordenada u2 do vetor u? "))
20     vetor3 = int(input("Qual a coordenada u3 do vetor u? "))
21     vetorv1 = int(input("Qual a coordenada v1 do vetor v? "))
22     vetorv2 = int(input("Qual a coordenada v2 do vetor v? "))
23     vetorv3 = int(input("Qual a coordenada v3 do vetor v? "))
24     print("A equação paramétrica do plano é: ")

```

```

Qual equação se deseja calcular? 3
São dois vetores e um ponto? 1
Qual a coordenada p1 do ponto? 1
Qual a coordenada p2 do ponto? -3
Qual a coordenada p3 do ponto? 4
Qual a coordenada u1 do vetor u? 2
Qual a coordenada u2 do vetor u? -3
Qual a coordenada u3 do vetor u? 7
Qual a coordenada v1 do vetor v? -3
Qual a coordenada v2 do vetor v? 2
Qual a coordenada v3 do vetor v? 5
O vetor normal é (-29,31,5)
A equação é (-29)x + (31)y + (5)z + -182 = 6

```

Figura 23 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para a equação do plano geral ou cartesiana.

Fonte: Autor, 2022.

9) Planos – Posição Relativa;

Os planos, quando analisadas entre si e em relação à uma reta, podem ter características diferentes. Em relação à posição relativa entre planos, existem quatro casos, de forma que os planos podem ser coincidentes, paralelos distintos, secantes não perpendiculares e secantes perpendiculares, ou ainda, pode haver uma intersecção entre os planos. Já no que tange a posição relativa entre planos e retas, existem as retas contidas no plano, as retas paralelas ao plano, as retas concorrentes e perpendiculares e as retas concorrentes e não perpendiculares.

Iniciando as análises pela posição relativa entre a reta e o plano, propõe-se alguns exemplos:

Dado um plano $\pi : x + y - 2z - 1 = 0$ e uma reta $r : X = (1,2,1) + \alpha(1, 1, 1)$, verifica-se a posição relativa entre eles.

O ponto da reta e o vetor diretor são, respectivamente, $(1,2,1)$ e $(1,1,1)$ e o vetor normal do plano é $(1,1,-2)$.

Calculando a multiplicação entre os vetores diretor e normal, é possível determinar se esse caso será de uma reta contida ou paralela ao plano, visto que, se o resultado for 0, é contida, se for diferente de 0, é paralela.

$$\vec{V}_r * \vec{N}_\pi = (1,1,1) * (1,1,-2) = (1 + 1 - 2) = 0$$

Além disso, para verificar se o resultado procede, basta substituir os valores do ponto na equação do plano.

$$\pi : x + y - 2z - 1 = 0$$

$$\pi : 1 + 2 - 2(1) - 1 = 0$$

$$\pi : 0 = 0$$

Ou seja, a reta está contida no plano.

Analisando outra situação, de mesma equação da reta $r : X = (1,2,1) + \alpha(1, 1, 1)$ e do plano $\pi : x + y - 2z + 3 = 0$.

O ponto da reta e o vetor diretor são, respectivamente, $(1,2,1)$ e $(1,1,1)$ e o vetor normal do plano é $(1,1,-2)$.

Calculando a multiplicação entre os vetores diretor e normal, é possível determinar se esse caso será de uma reta contida ou paralela ao plano, visto que, se o resultado for 0, é contida, se for diferente de 0, é paralela.

$$\vec{V}_r * \vec{N}_\pi = (1,1,1) * (1,1,-2) = (1 + 1 - 2) = 0$$

Além disso, para verificar se o resultado procede, basta substituir os valores do ponto na equação do plano.

$$\pi : x + y - 2z + 3 = 0$$

$$\pi : 1 + 2 - 2(1) + 3 = 0$$

$$\pi : 3 \neq 0$$

Dessa forma, a reta não está contida no plano.

Outro caso a se analisar é o dado pelo plano $\pi : x + 2y + z - 1 = 0$ e uma reta $r : X = (1, -1, 3) + \alpha(1, 1, 1)$, verifica-se a posição relativa entre eles.

O ponto da reta e o vetor diretor são, respectivamente, $(1, -1, 3)$ e $(1, 1, 1)$ e o vetor normal do plano é $(1, 2, 1)$.

Verificando os vetores diretor e normal, é possível determinar se esse caso será de uma reta concorrente não perpendicular, cujos vetores são linearmente dependentes, ou concorrente perpendicular ao plano, em que os vetores são linearmente independentes.

$$\frac{\vec{V}_r}{\vec{N}_\pi} = \frac{1}{1} \neq \frac{1}{2} \neq \frac{1}{1}$$

Ou seja, a reta é concorrente e não perpendicular.

Por fim, dado pelo plano $\pi : x + 2y - z = 0$ e uma reta $r : X = (2, 1, 3) + \alpha(2, 4, -2)$, verifica-se a posição relativa entre eles.

O ponto da reta e o vetor diretor são, respectivamente, $(2, 1, 3)$ e $(2, 4, -2)$ e o vetor normal do plano é $(1, 2, -1)$.

Verificando os vetores diretor e normal, é possível determinar se esse caso será de uma reta concorrente não perpendicular, cujos vetores são linearmente dependentes, ou concorrente perpendicular ao plano, em que os vetores são linearmente independentes.

$$\frac{\vec{V}_r}{\vec{N}_\pi} = \frac{2}{1} = \frac{4}{2} = \frac{-2}{-1} = 2$$

Ou seja, a reta é concorrente e perpendicular.

Demonstrando os casos, é possível ver, por meio da aplicação, a precisão da calculadora e a conferência em relação aos resultados.

```

1 posi = int(input("É posição relativa entre plano e reta? "))
2
3 if posi == 1:
4     print("Digite os dados referentes à equação do plano:")
5     pont1 = int(input("Qual a coordenada a do vetor normal? "))
6     pont2 = int(input("Qual a coordenada b do vetor normal? "))
7     pont3 = int(input("Qual a coordenada c do vetor normal? "))
8     pont4 = int(input("Qual a coordenada d do plano? "))
9     print("A equação geral do plano é p = ({}x + {}y + {}z + {}
= 0".format(pont1,pont2,pont3,pont4))
10    print("Digite os dados referentes à equação da reta:")
11    pont1r = int(input("Qual a coordenada p1 do ponto? "))
12    pont2r = int(input("Qual a coordenada p2 do ponto? "))
13    pont3r = int(input("Qual a coordenada p3 do ponto? "))
14    vetor1r = int(input("Qual a coordenada u1 do vetor diretor u? "))
15    vetor2r = int(input("Qual a coordenada u2 do vetor diretor u? "))
16    vetor3r = int(input("Qual a coordenada u3 do vetor diretor u? "))
17    print("A equação vetorial da reta é t: X = {} +
a({} ".format((pont1r,pont2r,pont3r),(vetor1r,vetor2r,vetor3r)))
18
19 if (vetor1r*pont1) + (vetor2r*pont2) + (vetor3r*pont3) == 0:
20
21     if (pont1*pont1r)+(pont2*pont2r)+(pont3*pont3r)+(pont4) == 0:
22         print("A reta está contida no plano")
23
24     if (pont1*pont1r)+(pont2*pont2r)+(pont3*pont3r)+(pont4) != 0:

```

```

É posição relativa entre plano e reta? 1
Digite os dados referentes à equação do plano:
Qual a coordenada a do vetor normal? 1
Qual a coordenada b do vetor normal? 1
Qual a coordenada c do vetor normal? -2
Qual a coordenada d do plano? -1
A equação geral do plano é p = (1)x + (1)y + (-2)z + (-1) = 0
Digite os dados referentes à equação da reta:
Qual a coordenada p1 do ponto? 1
Qual a coordenada p2 do ponto? 2
Qual a coordenada p3 do ponto? 1
Qual a coordenada u1 do vetor diretor u? 1
Qual a coordenada u2 do vetor diretor u? 1
Qual a coordenada u3 do vetor diretor u? 1
A equação vetorial da reta é t: X = (1, 2, 1) + a((1, 1, 1))
A reta está contida no plano

```

Figura 24 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para posição relativa entre reta e plano (reta contida).

Fonte: Autor, 2022.

```

1 posi = int(input("É posição relativa entre plano e reta? "))
2
3 if posi == 1:
4     print("Digite os dados referentes à equação do plano:")
5     pont1 = int(input("Qual a coordenada a do vetor normal? "))
6     pont2 = int(input("Qual a coordenada b do vetor normal? "))
7     pont3 = int(input("Qual a coordenada c do vetor normal? "))
8     pont4 = int(input("Qual a coordenada d do plano? "))
9     print("A equação geral do plano é p = ({}x + {}y + {}z + {}
= 0".format(pont1,pont2,pont3,pont4))
10    print("Digite os dados referentes à equação da reta:")
11    pont1r = int(input("Qual a coordenada p1 do ponto? "))
12    pont2r = int(input("Qual a coordenada p2 do ponto? "))
13    pont3r = int(input("Qual a coordenada p3 do ponto? "))
14    vetor1r = int(input("Qual a coordenada u1 do vetor diretor u? "))
15    vetor2r = int(input("Qual a coordenada u2 do vetor diretor u? "))
16    vetor3r = int(input("Qual a coordenada u3 do vetor diretor u? "))
17    print("A equação vetorial da reta é t: X = {} +
a({} ".format((pont1r,pont2r,pont3r),(vetor1r,vetor2r,vetor3r)))
18
19 if (vetor1r*pont1) + (vetor2r*pont2) + (vetor3r*pont3) == 0:
20
21     if (pont1*pont1r)+(pont2*pont2r)+(pont3*pont3r)+(pont4) == 0:
22         print("A reta está contida no plano")
23
24     if (pont1*pont1r)+(pont2*pont2r)+(pont3*pont3r)+(pont4) != 0:

```

```

É posição relativa entre plano e reta? 1
Digite os dados referentes à equação do plano:
Qual a coordenada a do vetor normal? 1
Qual a coordenada b do vetor normal? 1
Qual a coordenada c do vetor normal? -2
Qual a coordenada d do plano? 3
A equação geral do plano é p = (1)x + (1)y + (-2)z + (3) = 0
Digite os dados referentes à equação da reta:
Qual a coordenada p1 do ponto? 1
Qual a coordenada p2 do ponto? 2
Qual a coordenada p3 do ponto? 1
Qual a coordenada u1 do vetor diretor u? 1
Qual a coordenada u2 do vetor diretor u? 1
Qual a coordenada u3 do vetor diretor u? 1
A equação vetorial da reta é t: X = (1, 2, 1) + a((1, 1, 1))
A reta é paralela ao plano, logo, não está contida

```

Figura 25 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para posição relativa entre reta e plano (reta paralela).

Fonte: Autor, 2022.

```

1 posi = int(input("É posição relativa entre plano e reta? "))
2
3 v if posi == 1:
4     print("Digite os dados referentes à equação do plano:")
5     pont1 = int(input("Qual a coordenada a do vetor normal? "))
6     pont2 = int(input("Qual a coordenada b do vetor normal? "))
7     pont3 = int(input("Qual a coordenada c do vetor normal? "))
8     pont4 = int(input("Qual a coordenada d do plano? "))
9     print("A equação geral do plano é p = ({}x + ({}y + ({}z + ({}
= 0".format(pont1,pont2,pont3,pont4))
10    print("Digite os dados referentes à equação da reta:")
11    pont1r = int(input("Qual a coordenada p1 do ponto? "))
12    pont2r = int(input("Qual a coordenada p2 do ponto? "))
13    pont3r = int(input("Qual a coordenada p3 do ponto? "))
14    vetoru1r = int(input("Qual a coordenada u1 do vetor diretor u? "))
15    vetoru2r = int(input("Qual a coordenada u2 do vetor diretor u? "))
16    vetoru3r = int(input("Qual a coordenada u3 do vetor diretor u? "))
17    print("A equação vetorial da reta é t: X = {} +
a({})".format((pont1r,pont2r,pont3r),(vetoru1r,vetoru2r,vetoru3r)))
18
19 v if (vetoru1r*pont1) + (vetoru2r*pont2) + (vetoru3r*pont3) == 0:
20
21 v     if (pont1*pont1r)+(pont2*pont2r)+(pont3*pont3r)+(pont4) == 0:
22         print("A reta está contida no plano")
23
24 v     if (pont1*pont1r)+(pont2*pont2r)+(pont3*pont3r)+(pont4) != 0:

```

```

É posição relativa entre plano e reta? 1
Digite os dados referentes à equação do plano:
Qual a coordenada a do vetor normal? 1
Qual a coordenada b do vetor normal? 2
Qual a coordenada c do vetor normal? 1
Qual a coordenada d do plano? -1
A equação geral do plano é p = (1)x + (2)y + (1)z + (-1) = 0
Digite os dados referentes à equação da reta:
Qual a coordenada p1 do ponto? 1
Qual a coordenada p2 do ponto? -1
Qual a coordenada p3 do ponto? 3
Qual a coordenada u1 do vetor diretor u? 1
Qual a coordenada u2 do vetor diretor u? 1
Qual a coordenada u3 do vetor diretor u? 1
A equação vetorial da reta é t: X = (1, -1, 3) + a((1, 1, 1))
A reta é concorrente e não perpendicular em relação ao plano

```

Figura 26 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para posição relativa entre reta e plano (reta concorrente e não perpendicular).

Fonte: Autor, 2022.

```

1 posi = int(input("É posição relativa entre plano e reta? "))
2
3 v if posi == 1:
4     print("Digite os dados referentes à equação do plano:")
5     pont1 = int(input("Qual a coordenada a do vetor normal? "))
6     pont2 = int(input("Qual a coordenada b do vetor normal? "))
7     pont3 = int(input("Qual a coordenada c do vetor normal? "))
8     pont4 = int(input("Qual a coordenada d do plano? "))
9     print("A equação geral do plano é p = ({}x + ({}y + ({}z + ({}
= 0".format(pont1,pont2,pont3,pont4))
10    print("Digite os dados referentes à equação da reta:")
11    pont1r = int(input("Qual a coordenada p1 do ponto? "))
12    pont2r = int(input("Qual a coordenada p2 do ponto? "))
13    pont3r = int(input("Qual a coordenada p3 do ponto? "))
14    vetoru1r = int(input("Qual a coordenada u1 do vetor diretor u? "))
15    vetoru2r = int(input("Qual a coordenada u2 do vetor diretor u? "))
16    vetoru3r = int(input("Qual a coordenada u3 do vetor diretor u? "))
17    print("A equação vetorial da reta é t: X = {} +
a({})".format((pont1r,pont2r,pont3r),(vetoru1r,vetoru2r,vetoru3r)))
18
19 v if (vetoru1r*pont1) + (vetoru2r*pont2) + (vetoru3r*pont3) == 0:
20
21 v     if (pont1*pont1r)+(pont2*pont2r)+(pont3*pont3r)+(pont4) == 0:
22         print("A reta está contida no plano")
23
24 v     if (pont1*pont1r)+(pont2*pont2r)+(pont3*pont3r)+(pont4) != 0:

```

```

É posição relativa entre plano e reta? 1
Digite os dados referentes à equação do plano:
Qual a coordenada a do vetor normal? 1
Qual a coordenada b do vetor normal? 2
Qual a coordenada c do vetor normal? -1
Qual a coordenada d do plano? 0
A equação geral do plano é p = (1)x + (2)y + (-1)z + (0) = 0
Digite os dados referentes à equação da reta:
Qual a coordenada p1 do ponto? 2
Qual a coordenada p2 do ponto? 2
Qual a coordenada p3 do ponto? 3
Qual a coordenada u1 do vetor diretor u? 2
Qual a coordenada u2 do vetor diretor u? 4
Qual a coordenada u3 do vetor diretor u? -2
A equação vetorial da reta é t: X = (2, 1, 3) + a((2, 4, -2))
A reta é concorrente e perpendicular em relação ao plano

```

Figura 27 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para posição relativa entre reta e plano (reta concorrente e perpendicular).

Fonte: Autor, 2022.

Partindo da análise das posições relativas entre dois planos, inicia-se a verificação por meio de dois planos dados, são eles:

$$\alpha : x + y + 2z + 4 = 0$$

$$\beta : -2x - 2y - 4z - 8 = 0$$

O procedimento para determinar se esses planos são coincidentes ou paralelos distintos, basta que os vetores normais do plano α e do plano β sejam linearmente dependentes, além das coordenadas d promoverem o mesmo valor.

$$-\frac{1}{2} = -\frac{1}{2} = -\frac{1}{2} = -\frac{1}{2}$$

Ou seja, esses planos são coincidentes.

Outro caso pode ser visto verificando outros dois planos dados:

$$\alpha : 2x + y - z + 3 = 0$$

$$\beta : -4x - 2y + 2z + 7 = 0$$

O procedimento para determinar se esses planos são coincidentes ou paralelos distintos, basta que os vetores normais do plano α e do plano β sejam linearmente dependentes, além das coordenadas d promoverem o mesmo valor.

$$-\frac{2}{4} = -\frac{1}{2} = -\frac{1}{2} \neq \frac{3}{7}$$

Como as coordenadas de d não promovem o mesmo valor, pode-se inferir que estes planos são paralelos distintos.

Já realizando o cálculo através de outros dois planos, busca-se determinar se estes são secantes não perpendiculares, em que o produto entre os vetores normais é diferente de 0 e linearmente independentes ou secantes perpendiculares, cujo produto entre os vetores normais é igual a 0 e linearmente dependentes.

Supondo duas situações, no caso 1, duas equações de plano, sendo elas:

$$\alpha : x + y + z + 1 = 0$$

$$\beta : 2x - y + 3z + 3 = 0$$

$$\vec{N}_\alpha * \vec{N}_\beta = (1,1,1) * (2,-1,3) = (2 - 1 + 3) = 4 \neq 0$$

Portanto, os planos são secantes não perpendiculares.

Já no caso 2, são elas:

$$\alpha : x + y + z + 1 = 0$$

$$\beta : x + y - 2z = 0$$

$$\vec{N}_\alpha * \vec{N}_\beta = (1,1,1) * (1,1,-2) = (1 + 1 - 2) = 0$$

Dessa forma, os planos são secantes perpendiculares.

Demonstrando os casos vistos acima na calculadora desenvolvida, é possível verificar a igualdade entre os valores obtidos.

```

1 posi = int(input("É posição relativa entre plano e reta? "))
2
3 if posi == 1:
4     print("Digite os dados referentes à equação do plano:")
5     pont1 = int(input("Qual a coordenada a do vetor normal? "))
6     pont2 = int(input("Qual a coordenada b do vetor normal? "))
7     pont3 = int(input("Qual a coordenada c do vetor normal? "))
8     pont4 = int(input("Qual a coordenada d do plano? "))
9     print("A equação geral do plano é p = ({}x + ({}y + ({}z + ({} =
0".format(pont1,pont2,pont3,pont4))
10    print("Digite os dados referentes à equação da reta:")
11    pont1r = int(input("Qual a coordenada p1 do ponto? "))
12    pont2r = int(input("Qual a coordenada p2 do ponto? "))
13    pont3r = int(input("Qual a coordenada p3 do ponto? "))
14    vetoru1r = int(input("Qual a coordenada u1 do vetor diretor u? "))
15    vetoru2r = int(input("Qual a coordenada u2 do vetor diretor u? "))
16    vetoru3r = int(input("Qual a coordenada u3 do vetor diretor u? "))
17    print("A equação vetorial da reta é t: X = {} +
a({})).format((pont1r,pont2r,pont3r),(vetoru1r,vetoru2r,vetoru3r)))
18
19     if (vetoru1r*pont1) + (vetoru2r*pont2) + (vetoru3r*pont3) == 0:
20

```

É posição relativa entre plano e reta? 2
 Digite os dados referentes à equação do plano A:
 Qual a coordenada a do vetor normal? 1
 Qual a coordenada b do vetor normal? 2
 Qual a coordenada c do vetor normal? 2
 Qual a coordenada d do plano? 4
 A equação geral do plano é p = (1)x + (1)y + (2)z + 4 = 0
 Digite os dados referentes à equação do plano B:
 Qual a coordenada a do vetor normal? -2
 Qual a coordenada b do vetor normal? -2
 Qual a coordenada c do vetor normal? -4
 Qual a coordenada d do plano? -8
 A equação geral do plano é p = (-2)x + (-2)y + (-4)z + -8 = 0
 Os planos são coincidentes

Figura 28 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para posição relativa entre planos (coincidentes).

Fonte: Autor, 2022.

```

1 posi = int(input("É posição relativa entre plano e reta? "))
2
3 if posi == 1:
4     print("Digite os dados referentes à equação do plano:")
5     pont1 = int(input("Qual a coordenada a do vetor normal? "))
6     pont2 = int(input("Qual a coordenada b do vetor normal? "))
7     pont3 = int(input("Qual a coordenada c do vetor normal? "))
8     pont4 = int(input("Qual a coordenada d do plano? "))
9     print("A equação geral do plano é p = ({}x + ({}y + ({}z + ({} =
0".format(pont1,pont2,pont3,pont4))
10    print("Digite os dados referentes à equação da reta:")
11    pont1r = int(input("Qual a coordenada p1 do ponto? "))
12    pont2r = int(input("Qual a coordenada p2 do ponto? "))
13    pont3r = int(input("Qual a coordenada p3 do ponto? "))
14    vetoru1r = int(input("Qual a coordenada u1 do vetor diretor u? "))
15    vetoru2r = int(input("Qual a coordenada u2 do vetor diretor u? "))
16    vetoru3r = int(input("Qual a coordenada u3 do vetor diretor u? "))
17    print("A equação vetorial da reta é t: X = {} +
a({})).format((pont1r,pont2r,pont3r),(vetoru1r,vetoru2r,vetoru3r)))
18
19     if (vetoru1r*pont1) + (vetoru2r*pont2) + (vetoru3r*pont3) == 0:
20
21     if (pont1*pont1r)+(pont2*pont2r)+(pont3*pont3r)+(pont4) == 0:

```

É posição relativa entre plano e reta? 2
 Digite os dados referentes à equação do plano A:
 Qual a coordenada a do vetor normal? 2
 Qual a coordenada b do vetor normal? 1
 Qual a coordenada c do vetor normal? -1
 Qual a coordenada d do plano? 3
 A equação geral do plano é p = (2)x + (1)y + (-1)z + 3 = 0
 Digite os dados referentes à equação do plano B:
 Qual a coordenada a do vetor normal? -4
 Qual a coordenada b do vetor normal? -2
 Qual a coordenada c do vetor normal? 2
 Qual a coordenada d do plano? 7
 A equação geral do plano é p = (-4)x + (-2)y + (2)z + 7 = 0
 Os planos são paralelos distintos

Figura 29 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para posição relativa entre planos (paralelos distintos).

Fonte: Autor, 2022.

```

1 posi = int(input("É posição relativa entre plano e reta? "))
2
3 if posi == 1:
4     print("Digite os dados referentes à equação do plano:")
5     pont1 = int(input("Qual a coordenada a do vetor normal? "))
6     pont2 = int(input("Qual a coordenada b do vetor normal? "))
7     pont3 = int(input("Qual a coordenada c do vetor normal? "))
8     pont4 = int(input("Qual a coordenada d do plano? "))
9     print("A equação geral do plano é p = ({}x + ({}y + ({}z + ({})) =
0".format(pont1,pont2,pont3,pont4))
10    print("Digite os dados referentes à equação da reta:")
11    pont1r = int(input("Qual a coordenada p1 do ponto? "))
12    pont2r = int(input("Qual a coordenada p2 do ponto? "))
13    pont3r = int(input("Qual a coordenada p3 do ponto? "))
14    vetoru1r = int(input("Qual a coordenada u1 do vetor diretor u? "))
15    vetoru2r = int(input("Qual a coordenada u2 do vetor diretor u? "))
16    vetoru3r = int(input("Qual a coordenada u3 do vetor diretor u? "))
17    print("A equação vetorial da reta é t: X = {} +
a({})".format(pont1r,pont2r,pont3r),(vetoru1r,vetoru2r,vetoru3r)))
18
19 if (vetoru1r*pont1) + (vetoru2r*pont2) + (vetoru3r*pont3) == 0:
20
21     if (pont1*pont1r)+(pont2*pont2r)+(pont3*pont3r)+(pont4) == 0:

```

```

É posição relativa entre plano e reta? 2
Digite os dados referentes à equação do plano A:
Qual a coordenada a do vetor normal? 1
Qual a coordenada b do vetor normal? 1
Qual a coordenada c do vetor normal? 1
Qual a coordenada d do plano? 1
A equação geral do plano é p = (1)x + (1)y + (1)z + 1 = 0
Digite os dados referentes à equação do plano B:
Qual a coordenada a do vetor normal? 2
Qual a coordenada b do vetor normal? -1
Qual a coordenada c do vetor normal? 3
Qual a coordenada d do plano? 3
A equação geral do plano é p = (2)x + (-1)y + (3)z + 3 = 0
Os planos são secantes não perpendiculares

```

Figura 30 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para posição relativa entre planos (secantes não perpendiculares).

Fonte: Autor, 2022.

```

1 posi = int(input("É posição relativa entre plano e reta? "))
2
3 if posi == 1:
4     print("Digite os dados referentes à equação do plano:")
5     pont1 = int(input("Qual a coordenada a do vetor normal? "))
6     pont2 = int(input("Qual a coordenada b do vetor normal? "))
7     pont3 = int(input("Qual a coordenada c do vetor normal? "))
8     pont4 = int(input("Qual a coordenada d do plano? "))
9     print("A equação geral do plano é p = ({}x + ({}y + ({}z + ({})) =
0".format(pont1,pont2,pont3,pont4))
10    print("Digite os dados referentes à equação da reta:")
11    pont1r = int(input("Qual a coordenada p1 do ponto? "))
12    pont2r = int(input("Qual a coordenada p2 do ponto? "))
13    pont3r = int(input("Qual a coordenada p3 do ponto? "))
14    vetoru1r = int(input("Qual a coordenada u1 do vetor diretor u? "))
15    vetoru2r = int(input("Qual a coordenada u2 do vetor diretor u? "))
16    vetoru3r = int(input("Qual a coordenada u3 do vetor diretor u? "))
17    print("A equação vetorial da reta é t: X = {} +
a({})".format(pont1r,pont2r,pont3r),(vetoru1r,vetoru2r,vetoru3r)))
18
19 if (vetoru1r*pont1) + (vetoru2r*pont2) + (vetoru3r*pont3) == 0:
20
21     if (pont1*pont1r)+(pont2*pont2r)+(pont3*pont3r)+(pont4) == 0:

```

```

É posição relativa entre plano e reta? 2
Digite os dados referentes à equação do plano A:
Qual a coordenada a do vetor normal? 1
Qual a coordenada b do vetor normal? 1
Qual a coordenada c do vetor normal? 1
Qual a coordenada d do plano? 1
A equação geral do plano é p = (1)x + (1)y + (1)z + 1 = 0
Digite os dados referentes à equação do plano B:
Qual a coordenada a do vetor normal? 1
Qual a coordenada b do vetor normal? 1
Qual a coordenada c do vetor normal? -2
Qual a coordenada d do plano? 0
A equação geral do plano é p = (1)x + (1)y + (-2)z + 0 = 0
Os planos são secantes perpendiculares

```

Figura 31 – Demonstração do resultado dos dados inseridos na aplicação, bem como o código programado para posição relativa entre planos (secantes perpendiculares).

Fonte: Autor, 2022.

4.2 INTERFACE GRÁFICA

A segunda etapa de desenvolvimento da programação foi a elaboração de uma interface gráfica para a calculadora. O recurso utilizado para a criação desta forma de interação está contido em uma das bibliotecas intrínsecas ao Python, conhecida no meio tecnológico como GUI, sendo a “tela de um programa” à qual o usuário tem acesso.

Dessa forma, ela diz respeito a forma de interação entre o usuário do computador e um programa através de uma representação gráfica, como ícones, botões e outros indicadores visuais. A interação se dá através do uso de mouses, teclados ou toque. Com eles, o usuário é capaz de selecionar símbolos e manipulá-los de forma a obter algum resultado prático para acessar aquela funcionalidade (MUNDO MAKER, 2021).

A elaboração da interface gráfica foi realizada em conjunto com a programação, por meio da plataforma Visual Studio Code, que oferece o recurso de execução dos dois códigos em apenas uma tela de exibição.

Para tornar possível a realização desta função, foi necessário utilizar os recursos contidos na biblioteca PYSIMPLEGUI, em que se fez imprescindível tornar as variáveis globais, de forma que fosse possível executá-las em qualquer linha de código, assim como definir todas as características das telas e com quais comandos deveriam ser acionadas.

A criação das telas foi executada definindo em quantos pixels cada uma delas seria gerada, quais informações iria disponibilizar e os dados referentes às instruções e tarefas da calculadora.

Para o desenvolvimento da tela inicial, estipulou-se uma tela de dimensões (800x500 pixels), contendo informações como: nome da instituição, nome da calculadora e instruções de uso, conforme vê-se nas linhas de código e na imagem da tela, disponibilizadas abaixo:

```
nb = ttk.Notebook(app)
nb.place(x=0, y=0, width=800, height=500)
global vdep; vdep = IntVar()
global vmudbase; vmudbase = IntVar()
global vprodescalar; vprodescalar = IntVar()
global vprodvetorial; vprodvetorial = IntVar()
global vprodmisto; vprodmisto = IntVar()
global vretas; vretas = IntVar()
global vretpos; vretpos = IntVar()
global vplanos; vplanos = IntVar()
global vplanospos; vplanospos = IntVar()
```

Figura 32 – Linhas de código utilizadas para elaboração da tela inicial.

Fonte: Autor, 2022.

```
txt1 = Label(app, text="Calculadora de Geometria Analítica", background="#0571B0", foreground="#000")
txt1.place(x=292.5, y=50, width=200, height=30)

txt2 = Label(app, text="Centro Universitário FEI", background="#0571B0", foreground="#000")
txt2.place(x=292.5, y=10, width=200, height=30)

imglogo = PhotoImage(file=pastaApp+"\\fei.png")
l_logo = Label(app, image=imglogo)
l_logo.place(x=312, y=90)

txt3 = Label(app, text="Pressione o botão abaixo para inserir qual operação deseja realizar", background="#0571B0", foreground="#000")
txt3.place(x=146.25, y=270, width=500, height=30)

btn1 = tk.Button(app, text="Prosseguir", command=createNewWindow, background="#0571B0")
btn1.pack()
btn1.place(x=292.5, y=310, width=200, height=30)

txt4 = Label(app, text="Por Julia Vitória de Lima", background="#0571B0", foreground="#000")
txt4.place(x=267, y=350, width=250, height=30)

app.mainloop()
```

Figura 33 – Linhas de código utilizadas para elaboração da tela inicial.

Fonte: Autor, 2022.

O resultado obtido está apresentado abaixo:



Figura 34 – Demonstração da interface gráfica da tela inicial da calculadora.

Fonte: Autor, 2022.

Na etapa subsequente, programou-se a tela de escolha das operações disponíveis, em que, executando uma ligação entre as variáveis e a tela, ao escolher uma das opções, é acionado um terminal de comando, no qual é possível escrever as informações referentes as coordenadas dos vetores e obter os resultados.

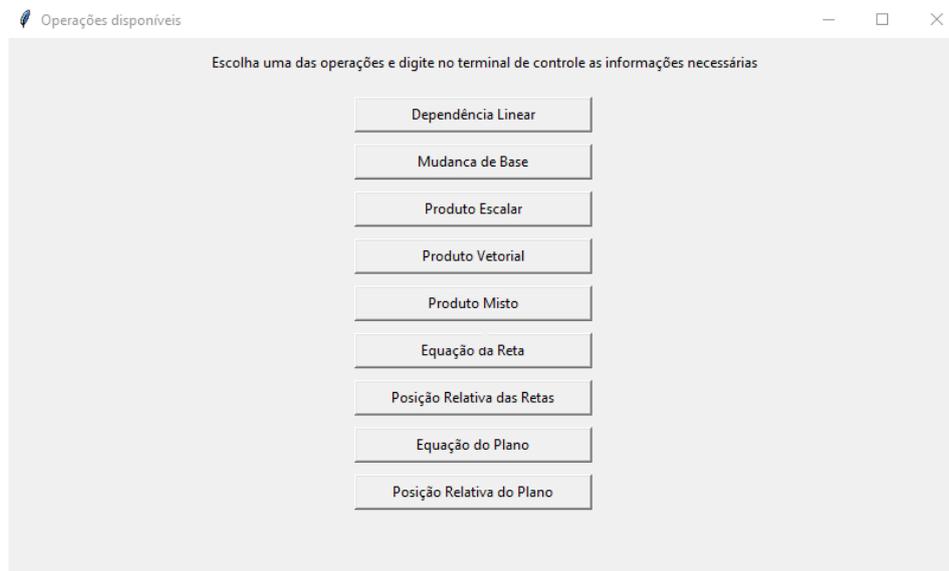


Figura 35 – Demonstração da interface gráfica da tela inicial da calculadora.

Fonte: Autor, 2022.

Exemplificando uma das operações, ao acionar o botão de produto escalar, é aberto o terminal de comando, permitindo que informações no que tange as coordenadas vetoriais sejam digitadas.

```
Quantos vetores serão analisados? 2
Qual a coordenada u1 do vetor u? 1
Qual a coordenada u2 do vetor u? 9
Qual a coordenada u3 do vetor u? 7
Qual a coordenada v1 do vetor v? 5
Qual a coordenada v2 do vetor v? 6
Qual a coordenada v3 do vetor v? 4
O valor do produto escalar entre estes vetores é 87
```

Figura 36 – Demonstração do terminal de comando para produto escalar.

Fonte: Autor, 2022.

A interface gráfica elaborada torna possível uma ligação entre o código e a apresentação dos dados e funções para os usuários, permitindo uma interação simples e prática.

5 CONCLUSÃO

Ao fim deste trabalho de pesquisa, é possível inferir que os objetivos iniciais foram atingidos, tendo em vista que uma calculadora utilizando a linguagem de programação Python com interface gráfica voltada para cálculos analíticos e vetoriais foi desenvolvida.

O trabalho obteve resultado satisfatório, analisando os dados obtidos com a execução do código e comparando-os com as plataformas referenciadas como recursos e apoios bibliográficos, sendo elas WolframAlpha, Geogebra e Symbolab.

O mecanismo desenvolvido visa tornar prática a resolução de exercícios e conferência dos alunos das disciplinas. Este recurso é um avanço no âmbito educacional no que tange as megatendências voltadas para a educação e a inteligência artificial, tendo em vista que é uma proposta desenvolvida no meio tecnológico e que possibilita a ampliação dos conteúdos, alterações de layout e melhorias no código da programação.

O projeto elaborado tem possibilidade de ser convertido para um aplicativo, de forma a ser uma ferramenta ainda mais acessível e compatível com os sistemas operacionais mais utilizados pelos estudantes, sendo eles Android e iOS.

Dessa forma, deixa-se como sugestão de continuidade para este trabalho de pesquisa a ampliação dos meios de acesso, tornando este recurso cada vez mais acessível e prático.

REFERÊNCIAS

WOLFRAM ALPHA. **Mathematics**. Disponível em: <https://www.wolframalpha.com/>. Acesso em: 13 jun. 2021.

SYMBOLAB. **Solucionador Matemático**. Disponível em: <https://pt.symbolab.com/>. Acesso em: 13 jun. 2021.

GEOGEBRA. **Aplicativos Matemáticos**. Disponível em: <https://www.geogebra.org/?lang=pt>. Acesso em: 3 jun. 2021.

LORETO, A. C. D. C; JUNIOR, A. P. L. **Vetores e Geometria Analítica: Teoria e Exercícios**. 4. ed. São Paulo: LCTE Editora, 2014. p. 1-202.

HALLIDAY, David; RESNICK, Robert; WALKER, Jearl. **Fundamentos da Física: Eletromagnetismo**. 10. ed. Rio de Janeiro: LTC, 2016. p. 198-395.

MENEZES, N. N. C. **Introdução à Programação com Python: Algoritmos e Lógica de Programação Para Iniciantes**. 3. ed. São Paulo: Novatec, 2010. p. 19-322.

CURSO EM VÍDEO. **Linguagem de Programação Python**. Disponível em: <https://www.cursoemvideo.com/>. Acesso em: 13 jun. 2021.

BORGES, Luiz Eduardo. **Python para Desenvolvedores: Aborda Python 3.3**. 1. ed. São Paulo: Novatec, 2014. p. 13-356.

TECNOBLOG. **O que é Python?** Disponível em: <https://tecnoblog.net/405640/o-que-e-python-guia-para-iniciantes/>. Acesso em: 14 jun. 2021.

BOULOS, Paulo; CAMARGO, Ivan. **Geometria Analítica: Um Tratamento Vetorial**. 3. ed. São Paulo: Pearson Universidades, 2004. p. 3-387.

LEE, Kent D. **Python Programming Fundamentals: Undergraduate Topics in Computer Science**. 2. ed. Iowa, USA: Springer, 2015. p. 6-241.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL. **Recursos Educacionais Abertos de Matemática**. Disponível em: <https://www.ufrgs.br/reamat/>. Acesso em: 12 jan. 2022.

WINTERLE, Paulo. **Vetores e Geometria Analítica: Cálculo Vetorial**. 2. ed. São Paulo: Pearson Universidades, 2014. p. 4-256.

MUNDOMAKER. **Interface gráfica: O que é, quais os tipos e como funciona**. Disponível em: <https://mundomaker.com.br/graphical-user-interface/>. Acesso em: 18 mai. 2022.