

Primeira avaliação escrita – parte A: valor 2,0 pts.

O acadêmico deve utilizar o SageMath, construir as rotinas pedidas a seguir e enviar o arquivo PDF com os códigos no ambiente Google-Classroom oficial da turma até dia 11 de maio às 11h59min. Após esse horário não será possível postar essa parte da avaliação.

Obs: Se o aluno tiver problemas com a configuração do SageMath na hora de exportar as linhas de programação para PDF, poderá copiar as linhas de código em um arquivo DOCX e exportá-lo para PDF.

01) Crie uma rotina que receba do usuário as coordenadas dos vértices de um triângulo $ABC \subset \mathbb{R}^2$. Seu programa deve imprimir na tela as coordenadas dos vértices do triângulo e imprimir as coordenadas dos pontos médios dos lados desse triângulo. Ele deve esclarecer que o ponto médio impresso é do lado AB , ou AC ou BC . Ao final, construa ilustração mostrando os vértices, o triângulo, e os pontos médios. O comando “line(P, Q)” cria o segmento de reta.

02) Crie um programa no qual o usuário defina em que dimensão quer trabalhar. Após tal escolha, seu programa deve mostrar um menu no qual o usuário escolha se quer calcular $\mathbf{u} + \mathbf{v}$, $\mathbf{u} - \mathbf{v}$ ou $\alpha \mathbf{v}$. Feita a escolha, o usuário deve inserir, separadas por vírgulas, as n coordenadas dos vetores necessários e/ou o escalar $\alpha \in \mathbb{R}$ (trabalhe com “float”). O programa deve retornar os vetores e/ou o escalar inserido e o vetor resultante da operação escolhida. Ao final, seu programa deve indagar se o usuário quer realizar outra operação ou encerrar o programa.

03) Crie um programa que receba do usuário um ponto $A = (x, y, z)$. A rotina deve indagar que projeção o usuário deseja obter: se quer a projeção do vetor \overrightarrow{OA} no eixo Ox , no eixo Oy ou no eixo Oz . O retorno em tela deve exibir as coordenadas do ponto inserido e as coordenadas da projeção com sua respectiva descrição. Ao final, deve mostrar uma ilustração com o vetor \overrightarrow{OA} e o vetor projeção escolhido pelo usuário. Seu programa deve indagar se o usuário quer obter projeções de outro vetor ou quer encerrar o programa.

§§§

Exemplo de captura de coordenadas

```
v=[]  
print("Insira as coordenadas do vetor separadas por vírgulas e aperte ENTER")  
v = input().split(',')  
v = list(map(float,v))  
print("O vetor inserido é v =",v,"de dimensão",len(v))  
§§§
```

Nesse procedimento, o usuário deve tomar cuidado para que a quantidade de coordenadas esteja correta. Vejamos qual o significado desses comandos. Numa outra célula digite os seguintes comandos e os execute.

