



## PYTHON PARA CIÊNCIA DE DADOS

### TRABALHO 02



Criar um programa em Python que representa graficamente a trajetória de um projétil em movimento oblíquo (conhecido também como movimento balístico). Nesse tipo de movimento, o projétil se desloca para a frente até uma altura máxima, e depois começa a descer devido à ação da força da gravidade, formando uma trajetória parabólica (Figura 1).

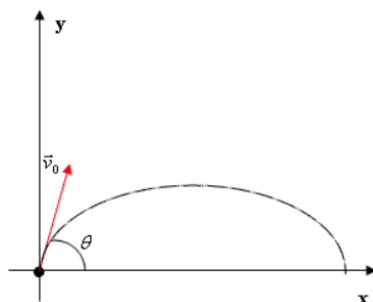


Figura 1: Ilustração da trajetória de um projétil em movimento oblíquo.

Como pode ser observado na figura acima, o projétil é lançado a um ângulo  $\theta$  com uma velocidade inicial  $v_0$ , estando sob a ação da força da gravidade  $g$ . As componentes horizontal e vertical da trajetória podem ser calculadas em função do tempo por meio das seguintes equações, respectivamente:

$$x(t) = |v_0| \cdot \cos(\theta) \cdot t$$

$$y(t) = |v_0| \cdot \sin(\theta) \cdot t - \frac{gt^2}{2}$$

#### Especificações:

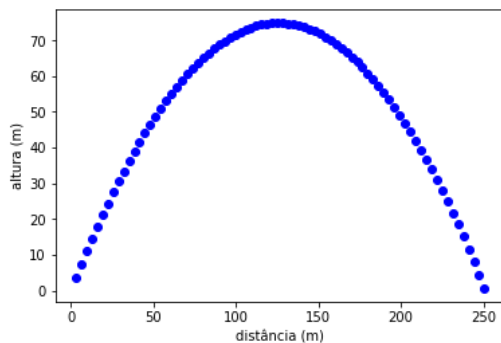
- O ângulo de lançamento ( $\theta$ ) deve ser lido do usuário (garanta que seja um valor válido).
- A velocidade inicial de lançamento ( $v_0$ ) deve ser lida do usuário (garanta que seja um valor válido).
- Deve-se variar o tempo ( $t$ ) em incrementos de 0,1 s.
- Deve-se assumir  $g = 9,8 \text{ m/s}^2$  (aceleração da gravidade no planeta Terra).
- Além de apresentar na tela o gráfico da trajetória do projétil, deve-se apresentar também: a distância total percorrida pelo mesmo (em metros), a altura máxima atingida (em metros) e a duração total do lançamento (em segundos).

#### Dicas:

- Para calcular as funções seno e cosseno em Python, é possível utilizar os métodos `np.sin()` e `np.cos()` pertencentes à biblioteca 'numpy' (importe a biblioteca fazendo: `import numpy as np`).
- Os métodos `np.sin()` e `np.cos()` esperam como parâmetro um valor em radianos. Para converter um ângulo de graus para radianos, utilize o método `np.deg2rad()`.
- Para calcular módulo em Python, utilize o método `abs()`.
- Para definir títulos nos eixos x e y de um gráfico em Python, utilize os comandos `plt.xlabel()` e `plt.ylabel()`, respectivamente.
- Para mostrar o gráfico, utilize o comando `plt.show()`.
- Para arredondar um valor 'x' em 'n' casas decimais, utilize o comando `round(x,n)`.

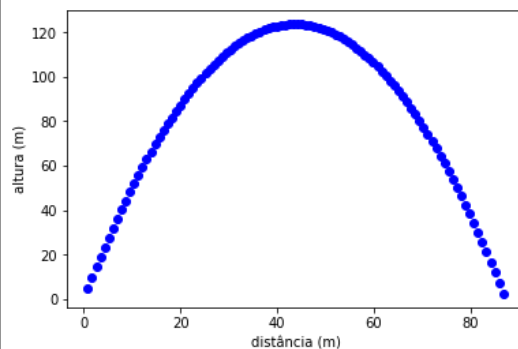
## Exemplo de execução:

Digite a velocidade (em m/s):50  
Digite o ângulo de lançamento (em graus):50



\*\*\*\*\*  
Distância percorrida..... 253.9 metros  
Altura máxima..... 74.8 metros  
Duração do lançamento.... 7.8 segundos  
\*\*\*\*\*

Digite a velocidade (em m/s):50  
Digite o ângulo de lançamento (em graus):80



\*\*\*\*\*  
Distância percorrida..... 87.7 metros  
Altura máxima..... 123.7 metros  
Duração do lançamento.... 10.0 segundos  
\*\*\*\*\*

### ⚠ ATENÇÃO:

- O trabalho pode ser feito individualmente, em dupla ou, no máximo, em trio.
- Colocar o nome de todos os integrantes na primeira linha do código, na forma de comentário.
- O programa em Python deverá ser postado no Blackboard. Para isso, acesse a sala “*Python para Ciência de Dados*” e clique em “*Conteúdo e Atividades*”. Para postar o programa, clique no item chamado “*Trabalho 2*”.

A interface do Blackboard mostra o menu lateral com opções como Avisos, Conteúdo e Atividades (destacado), Cronograma, Dúvidas e Comentários, Notas, Mensagens e Grupos. O Gerenciamento do curso está no rodapé. No painel principal, há uma barra de navegação com Criar conteúdo, Avaliações, Ferramentas e Conteúdo. Abaixo, há uma seção 'Materiais de Aula' com um link para acessar os materiais de aula da disciplina. Abaixo disso, há uma seção 'Trabalho 1 - PI (clique aqui para postar)' e, no final, uma seção 'Trabalho 2 - Projétil (clique aqui para postar)'.