

Unidade 4: Utilização da biblioteca TI PlotLib

Lição 2: Representar graficamente uma função

Nesta segunda lição da Unidade 4 vamos aprender como representar graficamente uma função utilizando a biblioteca **TI PlotLib** da aplicação TI-Python

Objetivos:

- Representar graficamente uma função.
- Revisitar o conceito de ciclo fechado.
- Configurar uma representação gráfica.

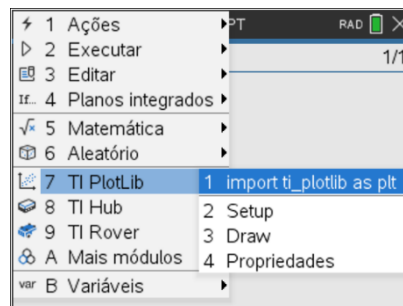
Propomos, nesta lição, a criação de um programa na aplicação TI-Python utilizando a biblioteca **TI PlotLib** que permite desenhar a representação gráfica de uma função para certos valores do objeto x pertencentes a um intervalo [a; b] com N segmentos.

O programa que irá criar será muito geral para que possa ser reutilizado em outros exemplos.

OBSERVAÇÃO:

Se os conceitos de ciclo e funções em Python não lhe são familiares, é aconselhável ir realizar as lições das unidades 1, 2 e 3 do TI-Código Python dos “10 Minutos de Código”.

- Começemos por abrir um novo documento e adicionar uma página com a aplicação TI-Python, criando um novo programa designando-o por U4L2.
- Importe, no início do programa, o módulo **TI PlotLib**, que se obtém clicando na tecla **menu** e depois escolhendo a opção **7: TI PlotLib**.
- Crie, de seguida, a função quadrática definida por $g(x)=x^2 - 3x + 4$.



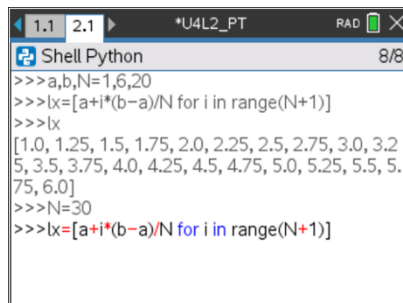
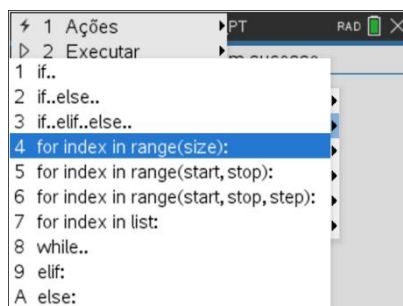
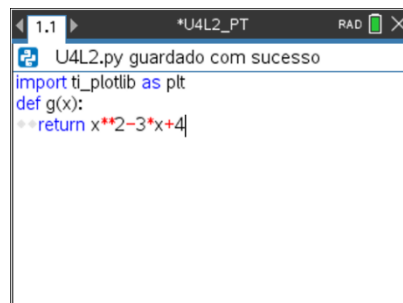
SUGESTÃO:

A lista das abcissas, que designaremos por **lx**, será construída utilizando um ciclo limitado cuja instrução **for i index in range(size)** poderá ser obtida clicando na tecla **menu**, depois seleccionar opção **4: Planos integrados** e por fim **2: Controlo**.

Já a lista das ordenadas, **ly**, obter-se-á a partir das imagens dos elementos da lista das abcissas, **lx**, portanto utilizaremos a opção **7: for index in list** do mesmo menu anterior.

DICA:

Insira um novo problema no seu documento e experimente esta forma de gerar uma lista de valores a partir de uma página do interpretador do TI-Python. Veja a imagem ao lado e faça a interpretação da estratégia usada.



Agora, estará pronto para obter uma representação gráfica da função g . Assim, insira sucessivamente as seguintes instruções no programa, tendo como objetivo efetuar o indicado em cada uma delas:

- Limpar o ecrã: `plt.cls()` .
- Ajustar os parâmetros da janela de visualização gráfica: `plt.window(xmin, xmax, ymin, ymax)`.
- Mostrar os eixos coordenados: `plt.axes("on")`.
- Colocar legenda nos eixos coordenados: `plt.labels("x", "y")`.
- Executar a representação gráfica: `plt.plot(lx, ly, "+")`.
- Mostrar a representação gráfica: `plt.show_plot()`.



```

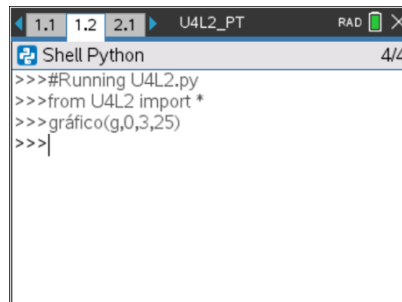
1.1 2.1 *U4L2_PT RAD 11/15
*U4L2.py
import ti_plotlib as plt
def g(x):
    return x**2-3*x+4
# Representação Gráfica em [a,b] sob N intervalos
def gráfico(f,a,b,N):
    lx=[a+i*(b-a)/N for i in range(N+1)]
    ly=[f(x) for x in lx]
    plt.cls()
    plt.window(-0.5,3.5,-1,5)
    plt.color(250,0,0)
    plt.plot(lx,ly,"+")
    plt.axes("on")
    plt.labels("x","y")
    plt.color(250,0,0)
    plt.plot(lx,ly,"+")
    plt.show_plot()
    
```

Todas essas instruções estão no módulo **TI PlotLib**, sendo que os parâmetros de configuração da representação gráfica se encontram no submenu **2: Configurar** e as instruções de representação gráfica no submenu **3: Desenhar**.

NOTE QUE:

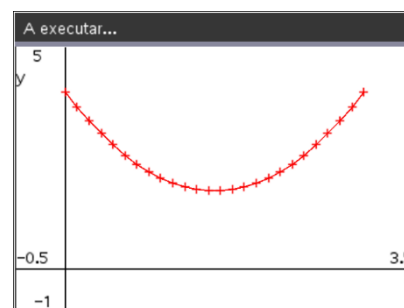
As instruções para a escolha da cor em RGB (vermelho, verde, azul) devem ser parametrizados entre 0 e 255 e inseridas no programa com localização criteriosa, de forma a evitar, por exemplo, que os eixos sejam vermelhos.

- Execute o programa, atalho `ctrl` + `R`, e na página do interpretador insira a função **gráfico()** clicando, por exemplo, na tecla `var`.
- Obtenha a representação gráfica da função g , no intervalo $[0, 3]$ e com a divisão em 25 subintervalos.
- Obterá a representação que se encontra na imagem ao lado.



```

1.1 1.2 2.1 U4L2_PT RAD 4/4
Shell Python
>>>#Running U4L2.py
>>>from U4L2 import *
>>>gráfico(g,0,3,25)
>>>|
    
```



SUGESTÃO:

Se optar por uma partição do intervalo num grande número de subintervalos, e consequentemente a representação ser obtida através de um grande número de pequenos segmentos, então deve optar pela marca da representação ser em pixel (\cdot) (em vez de $+$).

MAIS ALÉM:

Poderá alterar ou utilizar o programa para ir mais além na representação e estudo de funções, por exemplo:

- Mostrar a grelha de fundo no referencial.
- Alterar a função e/ou estudar várias funções
- Realizar um estudo matemático (cálculo diferencial), por exemplo, representando a função com 6 segmentos e depois 50.