



```
print("Introduza os parâmetros de uma função do tipo
ax^3+bx^2+cx+d")
a=float(input("a = "))
b=float(input("b = "))
c=float(input("c = "))
d=float(input("d = "))
def f(x):
    y=a*x**3+b*x**2+c*x+d
    return y
h=10**-5
x=float(input("Qual é o valor de x?"))
d=(f(x+h)-f(x))/h
print(d)
```

Editar Python na TI-nspire CX II-T

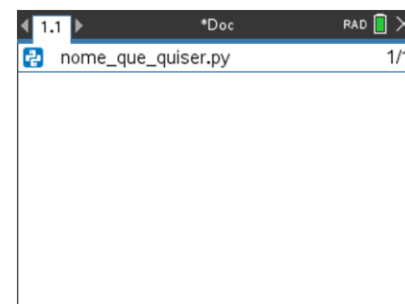
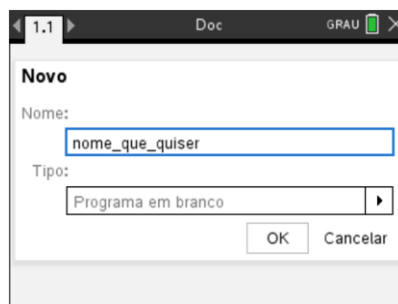
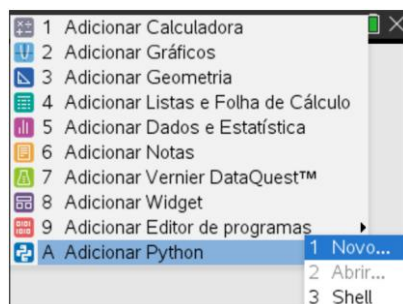
Ligue a sua calculadora e crie um novo documento.

Escolha uma página de *Python*:

A Adicionar Python → **1** Novo.

Coloque um nome à sua escolha, de seguida, prime em **OK**.

Abre-se uma página vazia, que é o editor de *Python* da calculadora/tecnologia TI-Nspire CX II-T, onde deve escrever o código.



1. Como determinar uma aproximação da variação instantânea de uma função do tipo $ax^3 + bx^2 + cx + d$ num dado ponto?

O que está por detrás deste programa em *Python* é a definição de taxa de variação instantânea como uma aproximação da taxa média de variação quando a amplitude do intervalo de variação é cada vez mais pequena, tanto quanto se queira.

Relembra-se que a taxa média de variação num intervalo $[x_0, x_1]$ do domínio de uma função f é:

$$t.m.v._{[f, x_0, x_1]} = \frac{f(x_1) - f(x_0)}{x_1 - x_0}$$

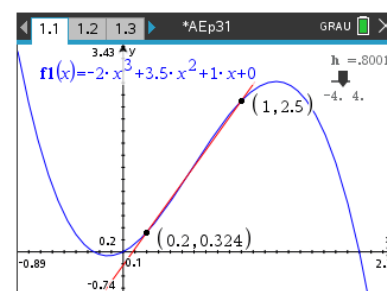
Designando por h a amplitude do intervalo, vem que $x_1 = x_0 + h$ e tem-se que:

$$t.m.v._{[f, x_0, x_1]} = \frac{f(x_0 + h) - f(x_0)}{x_0 + h - x_0} = \frac{f(x_0 + h) - f(x_0)}{h}$$

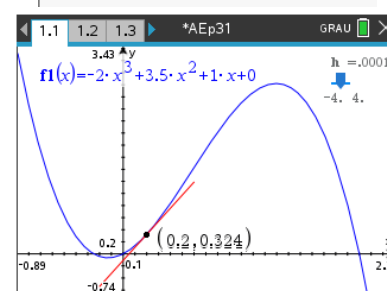
Quando $h \rightarrow 0$ estaremos a obter a taxa de variação instantânea em $x = x_0$.

Nas imagens ao lado, veja-se, graficamente, esta interpretação com a função $f(x) = -2x^3 + 3.5x^2 + x$, no ponto $x = 0.2$.

Há medida que diminuámos o intervalo de variação (aproximando h de 0), ficamos com uma melhor aproximação do valor da taxa de variação instantânea em 0.2.



$$\frac{f(0.2+h) - f(0.2)}{h} = 2.71991$$



$$\frac{f(0.2+h) - f(0.2)}{h} = 2.16023$$

- I. Embora se possa estender o programa para funções de grau superior a 3, aqui o foco será sobre as funções de grau 3 do tipo $f(x) = ax^3 + bx^2 + cx + d$. Assim, pretende-se fazer um programa que, depois de executado, comece por pedir ao utilizador para introduzir os valores dos parâmetros a , b , c e d . Este pedido é feito através de uma mensagem ao utilizador:

print("Introduza os parâmetros de uma função do tipo ax^3+bx^2+cx+d ")

Após a mensagem anterior, surge então o pedido ao utilizador para introduzir os valores dos coeficientes. Em *Python*, a função que permite receber um dado por parte do utilizador é **input()** e para que este seja atribuído numa variável, à qual deve ser dado um nome, tem de ser identificado que se trata de um valor numérico decimal, com **float()**.

a=float(input("a= "))

b=float(input("b= "))

c=float(input("c= "))

d=float(input("d= "))

- II. Para se calcular o valor da taxa média de variação é preciso a imagem de 0.2 pela função f , o valor da amplitude do intervalo, h , e a imagem de $0.2 + h$ pela mesma função. Para tal, será usada uma estrutura de função em *Python*.

Esta “função” é semelhante ao que noutras linguagens de programação se chama de sub-rotina, ou seja, uma secção de código que é ativada a partir das linhas de código do programa subsequentes. Esta estrutura é bastante útil, por exemplo, quando no programa se precisa de fazer algumas vezes as ações que nela estão previstas. A ideia desta estrutura é oportuna num contexto de funções matemáticas e, por isso, surge aqui como exemplo de um passo mais à frente em relação à simples utilização de uma estrutura condicional.

Para obter esta estrutura, caso conheça a sintaxe, poderá escrever com o teclado, mas pode também obter através do menu:

menu 4 Planos integrados → 1 funções → 1 def function():

Obtida a estrutura, é agora necessário preencher os campos em falta.

Assim, para a função f , que é definida como $f(x) = ax^3 + bx^2 + cx + d$ com os parâmetros já definidos anteriormente pelo utilizador, colocam-se as seguintes linhas de código:

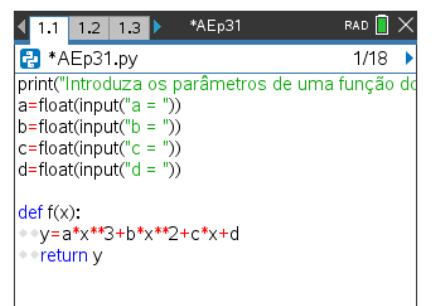
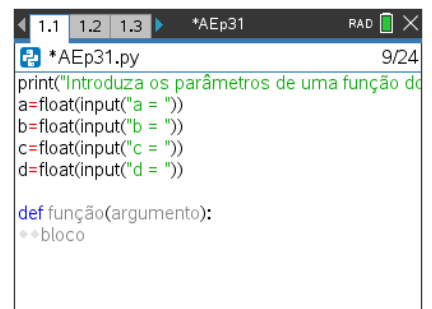
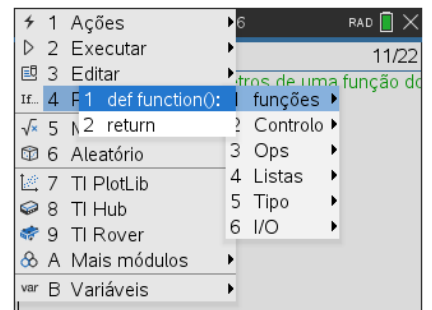
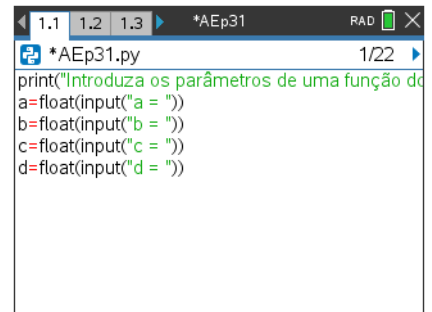
def f(x):

♦♦ y=a*x3+b*x**2+c*x+d**

♦♦ return y

Para que, depois da ação realizada por esta secção de código sobre o dado da entrada, o retorno ao programa principal se faça com o resultado, deve escrever-se **return** com o que se pretende que seja o resultado, a saída desta secção de código.

Este código pode ser facilmente escrito com o teclado, ou obtida no menu, na mesma lista pendente onde se obteve a função **def function()**. Da estrutura base, deve apagar-se o que não for necessário para esta secção funcionar conforme o pretendido.



III. Para se calcular a taxa média de variação é preciso definir o valor, que convém ser muito pequeno, da amplitude do intervalo, **h**, e o ponto, **x**, onde se pretende que seja calculado a taxa média de variação. Este último é um valor numérico decimal que será pedido ao utilizador.

Portanto, inserem-se, de seguida, as seguintes linhas de código:

```
h=10**-5                                # h=0.00005
a=float(input("Qual é o valor de x?"))
```

IV. Para se calcular a taxa média de variação basta dividir a diferença **fxh-fx** pelo valor da amplitude: $\frac{f(x_0+h)-f(x_0)}{h}$.

Para tal, insere-se a seguinte linha de código:

```
d=(fxh-fx)/h
```

V. De maneira a concluir o programa é necessário colocar as instruções que permitem a observação ou a impressão da informação pretendida, ou seja, do valor da aproximação, **d**, na página Shell Python, que se abre automaticamente logo após a execução do programa.

Deste modo, insere-se a seguinte linha de código:

```
print(d)
```

VI. Escrito o programa, falta executá-lo. Pode utilizar-se uma instrução do menu (menu [2] [1]), mas é claramente mais simples utilizar um atalho, uma combinação de teclas ([ctrl] + [R]).

O resultado aparece numa nova página destinada a mostrar o resultado da execução do programa, **Shell Python**, na qual também se podem fazer operações e pequenos programas, mas que não permanecerão gravados após o fecho da aplicação.

```
*AEp31.py 12/16
a=float(input("a = "))
b=float(input("b = "))
c=float(input("c = "))
d=float(input("d = "))

def f(x):
    y=a*x**3+b*x**2+c*x+d
    return y

h=10**-5
x=float(input("Qual é o valor de x?"))
```

```
*AEp31.py 14/16
d=float(input("d = "))

def f(x):
    y=a*x**3+b*x**2+c*x+d
    return y

h=10**-5
x=float(input("Qual é o valor de x?"))

d=(f(x+h)-f(x))/h
```

```
*AEp31.py 16/16
def f(x):
    y=a*x**3+b*x**2+c*x+d
    return y

h=10**-5
x=float(input("Qual é o valor de x?"))

d=(f(x+h)-f(x))/h

print(d)
```

```
Shell Python 11/11
>>>#Running AEp31.py
>>>from AEp31 import *
Introduza os parâmetros de uma função do tipo
ax^3+bx^2+cx+d
a = 1
b = 2
c = 3
d = 4
Qual é o valor de x?1
10.00005000015847
>>>
```

Para voltar ao editor de *Python*, onde poderá alterar os dados de entrada, por exemplo, há mais do que um procedimento à escolha, baseados no botão do touchpad. Pode fazer deslocar o cursor com o dedo até o sobrepor ao retângulo com a designação da página, [1.1], neste caso, e premir o touchpad na parte central (🖱️). Pode também utilizar os botões laterais do touchpad após premir a tecla [ctrl]. Neste caso, ao premir o botão lateral esquerdo, vai para a página anterior, a do editor. Pode voltar à página de *Shell Python* utilizando o mesmo tipo de procedimento, mas agora com o botão lateral direito.

Na parte superior do ecrã apenas se pode observar a designação e 3 páginas consecutivas, pelo que se o documento tiver mais páginas terá de conjugar os dois procedimentos referidos ou simplesmente o que recorre às teclas laterais do touchpad.



Algumas ideias sobre
programação,
relacionadas com o
contexto

