



```
idades=[14,16,14,15,17]
idades.sort()
soma=0
for i in range(5):
    soma=soma+idades[i]
print('Idades:',idades)
print('Media:',soma/5)
print('Mediana:',idades[2])
print('Máximo:',idades[4])
print('Mínimo:',idades[0])
```

Editar Python na TI-nspire CX II-T

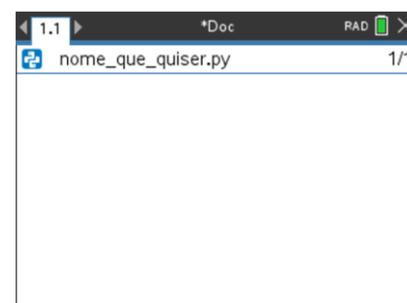
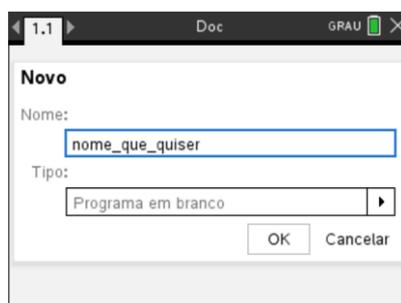
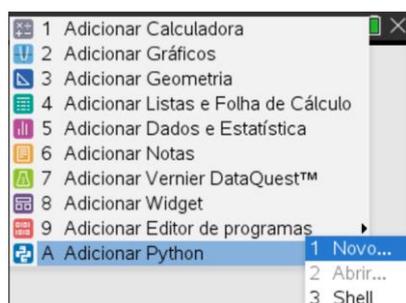
Ligue a sua calculadora e crie um novo documento.

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.



I. Dadas as idades de 5 alunos de uma turma, como obter a média, a mediana, o valor máximo e o valor mínimo?

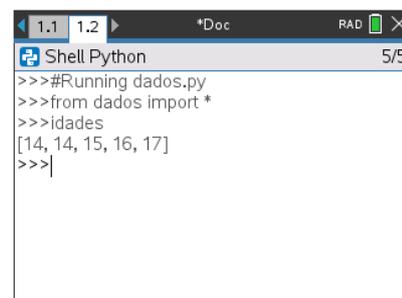
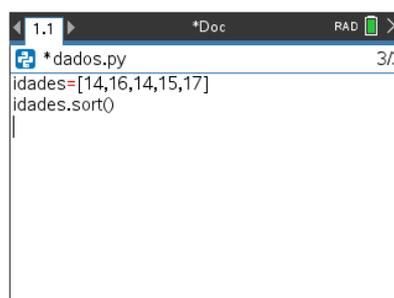
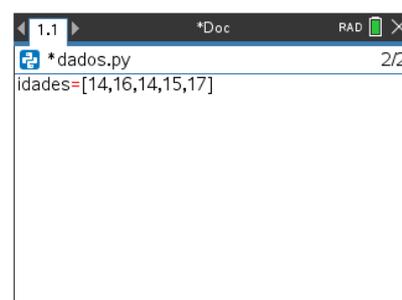
- I. Pretende-se um programa que, depois de executado, apresente em relação às idades dos 5 alunos, a média, a mediana, o maior valor da idade e também o menor. Estes dados (idades) estão organizados numa **lista** (em Python, é representada como uma sequência de objetos separados por vírgula e dentro de parênteses retos []). Em *Python*, a lista é um dos tipos de variável.

[] escreve-se facilmente com o teclado, mas pode também obter através do menu:

4 Planos integrados → **4** Listas → **1** []

Neste submenu **4** Listas, poderá encontrar diversas expressões a utilizar nas linhas de código

- II. Para obter a **mediana** é importante ter os dados ordenados, o que se pode fazer utilizando o código `nome_da_lista.sort()`. Tal como está o código, a ordenação ocorre de forma crescente. Depois de executar o programa com a sequência de teclas (**ctrl** + **R**), escreva o nome da lista na página *Shell Python* e prima **enter**. É uma forma rápida de fazer surgir a lista, imprimida com a ordem prevista.



Para obter a impressão da lista também poderia ter-se preparado o código nesse sentido, com a instrução **print(idades)**.

```
* dados.py 3/4
idades=[14,16,14,15,17]
idades.sort()
print(idades)
```

```
Shell Python 7/7
>>>#Running dados.py
>>>from dados import *
[14, 14, 15, 16, 17]
>>>#Running dados.py
>>>from dados import *
[14, 14, 15, 16, 17]
>>>
```

Note-se ainda que cada elemento de uma lista pode ser obtido pela sua posição.

Por exemplo, na lista **idades=[14,16,14,15,17]**, o primeiro elemento é **idades[0]**, o segundo **idades[1]**, até ao último, que é **idades[4]**. Por exemplo, **idades[3]=15**.

Assim, a mediana é **idades[2]** na lista ordenada.

```
* dados.py 4/5
idades=[14,16,14,15,17]
idades.sort()
print(idades)
print("Mediana: ",idades[2])
```

```
Shell Python 5/5
>>>#Running dados.py
>>>from dados import *
[14, 14, 15, 16, 17]
Mediana: 15
>>>
```

III. Para obter a média, um procedimento possível passa por determinar a soma das idades, dividida pelo número de elementos da lista “idades”.

Para obter a soma referida, pode percorrer-se a lista, elemento a elemento, e acumular-se os valores numa variável, iniciada em 0, o que pressupõe um algoritmo aplicado tantas vezes quantas o número de elementos da lista. Designe-se essa variável por **soma**, por exemplo.

soma=0 # no início vale 0

soma=soma+idades[0] # soma passa a valer o que valia a soma anterior (0), adicionada de **idades[0]**, que é 14.
soma=0+14

soma=soma+idades[1] # soma passa a valer o que valia a soma anterior (14), adicionada de **idades[1]**, que é também 14.
soma=14+14

Este procedimento repete-se até:

soma=soma+idades[4]

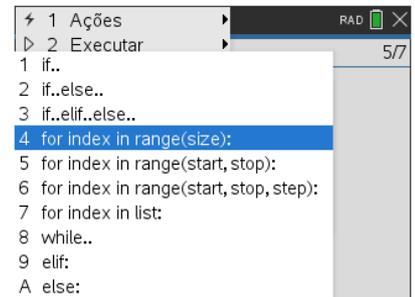
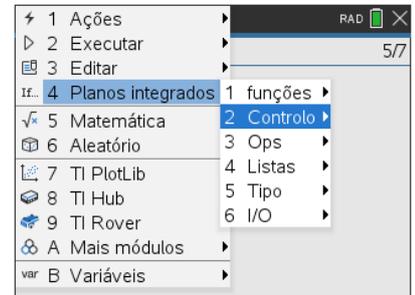
Na verdade, o algoritmo pode ser traduzido pela utilização de **soma=soma+idades[i]**, desde que **i=0** até que **i=4.**, que em código *Python* pode ser traduzido por:

```
for i in range(5):
    soma=soma+idades[i]
```

(Note que a variável *i* inicia em 0 e incrementa até ao valor anterior ao que está entre parênteses)

Para obter estas linhas de código pode utilizar o teclado, não esquecendo dos “:” e também da indentaçã “♦♦”, para o que o ciclo de repetição **for** se aplique apenas ao que estiver indentado. Pode também recorrer ao menu para obter as linhas de código, a completar com os elementos específicos.

menu **4** Planos integrados → **2** Controlo → **4** for index in range(size):



```
* dados.py 4/11
idades=[14,16,14,15,17]
idades.sort()
soma=0
for indice in range(tamanho):
    ♦♦bloco
```

```
* dados.py 8/10
idades=[14,16,14,15,17]
idades.sort()
soma=0
for i in range(5):
    ♦♦soma=soma+idades[i]
```

IV. Para se concluir o programa é necessário colocar as instruções que permitem a observação ou a impressão da informação pretendida, na página *Shell Python*, que se abre automaticamente logo após a execução do programa.

As linhas de código, a colocar agora fora da indentação do ciclo de repetição, sem necessidade de introdução dos comentários, são as que se seguem:

```
print( "Idades:" , idades) # imprime a lista inicial
```

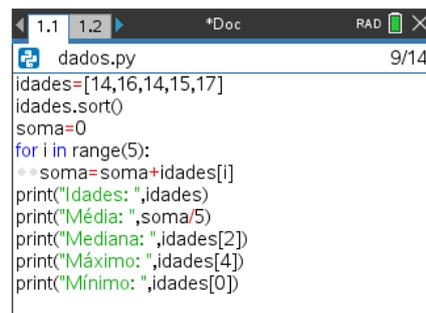
```
print( "Média:" , soma/5) # imprime a média das idades
```

```
print( "Mediana:" , idades[2]) # imprime a mediana das idades (o número de dados é ímpar)
```

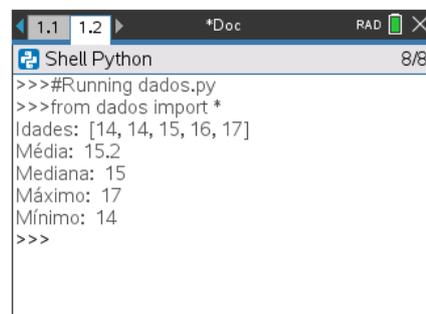
```
print( "Máximo:" , idades[4]) # imprime o máximo (como a lista está ordenada por ordem crescente, então é o último elemento da lista)
```

```
print( "Mínimo:" , idades[0]) # imprime o mínimo (o primeiro elemento na lista ordenada)
```

Nota: Para executar o programa pode utilizar-se uma instrução do menu (**menu** **2** **1**), mas é claramente mais simples utilizar um atalho, uma combinação de teclas (**ctrl** + **R**).



```
1.1 1.2 *Doc RAD 9/14
dados.py
idades=[14,16,14,15,17]
idades.sort()
soma=0
for i in range(5):
    soma=soma+idades[i]
print("Idades:" ,idades)
print("Média:" ,soma/5)
print("Mediana:" ,idades[2])
print("Máximo:" ,idades[4])
print("Mínimo:" ,idades[0])
```



```
1.1 1.2 *Doc RAD 8/8
Shell Python
>>>#Running dados.py
>>>from dados import *
Idades: [14, 14, 15, 16, 17]
Média: 15.2
Mediana: 15
Máximo: 17
Mínimo: 14
>>>
```

Para voltar ao editor de *Python*, deve utilizar o touchpad, seja com o toque do dedo e o botão central () depois de sobrepor o cursor ao número da página, ou com as teclas laterais, premindo previamente **ctrl**., para abrir a página anterior ou posterior.



Algumas ideias sobre programação, relacionadas com o contexto

