# Laço fechado

## NUMWORKS

O laço `for` é normalmente usado quando se quer repetir a mesma instrução várias vezes. Sendo fechado, é preciso saber o número de vezes que se deseja repetir a instrução.

Precisará de uma variável para contar o número de vezes que se vai repetir a instrução.

Em Python, indicamos os valores que devem ser tidos em conta por esta variável de contagem numa lista.

```
gra                    PYTHON
1 for i in [1,2,3,4,5]:
2   print("Hello!")
3
4
5
6
7
8
9
10
11
12
```

Neste exemplo, a variável de contagem é `i`. À primeira iteração, `i` toma o valor do primeiro elemento da lista, ou seja, 1, e a instrução é executada: exibimos `"Hello!"`. Uma vez executada a instrução, `i` toma o valor do segundo elemento da lista de instrução e executa novamente: `"Hello!"` é exibido uma segunda. Uma vez terminado o laço, teremos exibido `"Hello!"` cinco vezes.

## 1 for i in range(n)

```
gra                    PYTHON
1 for i in range(5):
2   print("Hello!")
3
4
5
6
7
8
9
10
11
12
```

A maior parte das vezes, não vamos escrever a lista que valores que `i` vai tomar. É preferível

usar `range(n)` que gera a lista dos primeiros n números inteiros. E para exibir 5 vezes `"Hello!"`, escrevemos:

ATENÇÃO

`range(n)` gera a lista dos primeiros n números inteiros a começar no 0 (portanto a terminar em n-1). Assim:

```
>>> range(5)
[0,1,2,3,4]
```

Pode também utilizar `range(n,m)` que lista os inteiros entre n e m-1:

```
>>> range(1,5)
[1,2,3,4]
```

Finalmente, o comando `range(n,m,p)` lista os números inteiros de de n a m-1 de p a p:

```
>>> range(1,10,2)
[1,3,5,7,9]
```

## 2 Exercício

Escreva a função `soma(n)` que tome um número inteiro como argumento e devolva a soma dos n primeiros inteiros (1 + 2 + 3 + ... + n) e que contenha um laço `for`.

## 3 Outro exercício

Escreva uma função `potencia(x,n)` que tome um valor real x e um inteiro natural n como argumento e devolva a x elevado n-ésima potência, $x^a$.