

Algoritmos e Programação I

Módulo 3: entrada e saída de dados e modularização

Prof. Dr. Said Sadique Adi



Entrada e saída de dados

- Muitos programas precisam interagir com o usuário, lendo dados fornecidos por ele pelo teclado e imprimindo mensagens ao usuário na tela (entrada e saída padrão).
- Em Python, essas interações são providas por meio de duas funções:
 - `print()`: imprime na tela o que recebe como argumento;
 - `input()`: realiza a leitura do que é digitado pelo usuário no teclado.

Função print()

- Recebe uma lista de valores separados por vírgula e imprime cada valor adicionando um espaço em branco entre eles e pulando de linha no final.

print(lista_de_valores_separados_por_vírgulas)

>>> print(42)		>>> x = 2		>>> q = 10
42		>>>> print(x)		>>> print("A carne pesa ", q, "kg")
>>> print("Olá")		2		A carne pesa 10 kg
Olá				

Outros argumento da função print()

- parâmetro **sep**: pode ser usado para alterar a string inserida automaticamente entre dois valores consecutivos da lista.

```
>>> print(2, 3, 4, sep="-*-")  
2-*-3-*-4
```

- parâmetro **end**: pode ser usado para alterar a string inserida automaticamente ao final da lista de valores.

```
>>> print("primeira linha", end = " || ")  
>>> print("segunda linha")  
primeira linha || segunda linha
```

Função input()

- Recebe uma mensagem (string) que é exibida na tela e que informa ao usuário o que precisa ser digitado.
- Ao terminar de digitar o que foi solicitado, o usuário deve teclar ENTER, e a função envia o que foi digitado (string) a uma variável.

var = input(mensagem)

- ***Para transformar o valor dado como entrada em um número, podemos fazer um casting;***

Exemplos de uso da função input()

```
>>> x = input("Digite o valor de x: ")
```

```
Digite o valor de x: 3
```

```
>>> x
```

```
'3'
```

```
>>> x + 2
```

```
Traceback (most recent call last):
```

```
  File "<stdin>", line 1, in <module>
```

```
TypeError: can only concatenate str (not "int") to str
```

```
>>> y = int(x)
```

```
>>> y + 2
```

```
5
```

Funções

- Uma função é um pedaço de código que executa alguma tarefa e que pode ser reutilizado quantas vezes for necessário.
- Úteis quando necessita-se executar uma sequência de instruções várias vezes.
- Permitem reutilizar partes do código e modularizá-lo, facilitando a sua manutenção.
- Permitem aos programadores resolver um problema decompondo-o em problemas menores.

Funções em Python

- Sintaxe de uma definição de função:

```
def NOME ( lista de parâmetros ):  
    comandos (com indentação)  
    return <valor> → opcional
```
- O nome que pode ser atribuído a uma função segue as mesmas regras de nomes dados a variáveis.
- A lista de parâmetros especificam o que a função precisa para cumprir seu objetivo, podendo ser vazia ou conter qualquer número de parâmetros separados por vírgula.

Exemplo de função em Python

```
>>> def tabuada7():  
...     n = 1  
...     while n < 11:  
...         print(n * 7, end = ' ')  
...         n = n + 1
```

```
>>> tabuada7()  
7 14 21 28 35 42 49 56 63 70
```