



Universidade Federal do Rio Grande  
Centro de Ciências Computacionais



# Algoritmos e Estruturas de Dados I

## Funções

Profs. Drs. Cleo Billa, Rafael Penna e Thiago da Silveira

1º Semestre de 2020

# Roteiro

- Motivação
- Funções conhecidas
- Definição e chamada de funções
- Funções e argumentos
- Funções e retorno

# Motivação

- Clareza;
- Modularidade;
- Reutilização;
- *Debugging*;

```
lista = [1,2,4,3,5]
tamanho = 5

for i in range(tamanho):
    print(lista[i])

numero = int(input("Informe um número: "))

print("Multiplicando...")

for i in range(tamanho):
    lista[i] = numero * lista[i]

for i in range(tamanho):
    print(lista[i])
```

# Funções conhecidas

- `print(...)`
- `... = input(...)`
- `... = len(...)`

```
texto = input("Digite alguma coisa: ")
tamanho = len(texto)
print("O comprimento do texto é", tamanho)
```

# Definição e chamada de funções

- Definição

```
def nome():
```

 . . .

Indentação

- Chamada (uso)

. . .

```
nome()
```

. . .

```
def ...  
  
def ...  
  
def ...  
  
print(...)  
... = ...  
... = input(...)  
while ...:  
    print(...)  
    ... = ...  
    if ...:  
        print(...)  
        if ...:  
            ... = ...  
        else:  
            ...
```

# Definição e chamada de funções

- Definição

```
def nome():
```

```
    ...
```

Identação

- Chamada (uso)

```
...
```

```
nome()
```

```
...
```

```
def ...  
def ...  
def ...  
  
print(...)  
... = ...  
... = input(...)  
while ...:  
    print(...)  
    ... = ...  
    if ...:  
        print(...)  
        if ...:  
            ... = ...  
        else:  
            ...
```

# Definição e chamada de funções

- Definição
- Chamada (uso)

`def nome():`

`...`

Identação

`...`

`nome()`

`...`

Mesmas regras dos nomes de variáveis

```
def ...  
def ...  
def ...  
print(...)  
... = ...  
... = input(...)  
while ...:  
    print(...)  
    ... = ...  
    if ...:  
        print(...)  
        if ...:  
            ... = ...  
        else:  
            ...
```

# Definição e chamada de funções

- Definição
- Chamada (uso)

```
def ola_mundo():  
    print("Olá mundo!")
```

```
for indice in range(10):  
    ola_mundo()
```



# Funções e argumentos

- Argumentos

```
def nome( ):
```

{ . . .

Indentação

```
def nome(arg):
```

{ . . .

Indentação

# Funções e argumentos

- Argumentos

```
def nome(arg):
```

{ . . .

Indentação

Tantos quanto se queira

```
def nome(arg1, arg2):
```

{ . . .

Indentação

# Funções e argumentos

- Exemplos

```
def bem_vindo():  
    print("Bem-vindo ao Programa XYZ!")  
    print("Desenvolvedor: Fulano de Tal")  
    print("Outubro de 2020.")
```

```
def bem_vindo2(nome, dia, mes, ano):  
    print("Bem-vindo de novo,", nome)  
    print("Seu último acesso foi em",  
          dia, "/", mes, "/", ano)
```

```
def menu_info():  
    print("Para pular, aperte W")  
    print("Para ir para frente, aperte D")  
    print("Para ir para trás, aperte A")  
    print("Para ir para baixo, aperte S")
```

```
bem_vindo2("Fulano", 10, 11, 2019)  
menu_info()
```

# Funções e retorno

- Retorno

```
def nome(...):
```

 . . .

Identação

```
def nome(...):
```

 . . .

Identação

```
    return ret
```

# Funções e retorno

- Retorno

```
def nome(...):
```

 . . .

Identação

```
    return ret
```

```
def nome(...):
```

 . . .

Identação

```
    return ret1, ret2
```



Tantos quanto se queira

# Funções e retorno

- Exemplos

```
def bem_vindo2(nome, dia, mes, ano):  
    print("Bem-vindo de novo,", nome)  
    print("Seu último acesso foi em",  
          dia, "/", mes, "/", ano)
```

```
def pi(casas_decimais):  
    if casas_decimais == 2:  
        return 3.14  
    elif casas_decimais == 4:  
        return 3.1416  
    else:  
        return 3.14159265359
```

```
def quadrado(numero):  
    return numero*numero
```

```
def area_circulo(raio, casas_decimais):  
    return quadrado(raio)*pi(casas_decimais)
```

```
raio = float(input("Informe o raio: "))  
area = area_circulo(raio, 4)  
print("A area do circulo é", area)
```

# Funções, argumentos e retorno

	<b>Sem retorno</b>	<b>Com retorno</b>
<b>Sem argumento</b>	Sem argumento, sem retorno	Sem argumento, com retorno
<b>Com argumento</b>	Com argumento, sem retorno	Com argumento, com retorno

# Motivação... de novo

- Clareza;
- Modularidade;
- Reutilização;
- *Debugging*;

```
lista = [1,2,4,3,5]
tamanho = 5

for i in range(tamanho):
    print(lista[i])

numero = int(input("Informe um número: "))

print("Multiplicando...")

for i in range(tamanho):
    lista[i] = numero * lista[i]

for i in range(tamanho):
    print(lista[i])
```



# Motivação... de novo

```
def multiplica(valor1, valor2):  
    return valor1 * valor2  
  
def imprime(listaoustring):  
    for i in range(len(listaoustring)):  
        print(listaoustring[i])
```

```
lista = [1,2,4,3,5]  
  
imprime(lista)  
  
numero = int(input("Informe um número: "))  
  
print("Multiplicando...")  
  
for i in range(len(lista)):  
    lista[i] = multiplica(lista[i], numero)  
  
imprime(lista)
```



Universidade Federal do Rio Grande  
Centro de Ciências Computacionais



# Algoritmos e Estruturas de Dados I

## Funções

Profs. Drs. Cleo Billa, Rafael Penna e Thiago da Silveira

1º Semestre de 2020