

Universidade Federal de Alfenas - UNIFAL-MG - *campus* Varginha
Bacharelado Interdisciplinar em Ciência e Economia
Disciplina: Fundamentos da programação - Profa. Patrícia de Siqueira Ramos
Lista 6 - Funções

Instruções:

- Implemente todas as questões computacionalmente
- Para todas as questões: defina a função no **python** e utilize um exemplo de uso para ela, veja no exemplo abaixo

Definição da função (no lugar de **return** use **print** se o valor retornado não for um número):

```
def funcao(parametros):  
    -----  
    -----  
    return -----
```

Uso da função:

```
var1 = int(input('Mensagem ao usuário:'))  
var2 = float(input('Outra mensagem ao usuário:'))  
funcao(var1, var2)      # chamada à função
```

1. Faça uma função que se chame **sinal** que receba um número e retorne 1 se esse número for positivo, 0 se for zero ou -1 se for negativo. Exemplos de saídas:

- **sinal(100)** deve retornar 1
- **sinal(-12)** deve retornar -1
- **sinal(0)** deve retornar 0
- e assim por diante.

2. Escreva a função **modulo** que recebe um número e retorna seu módulo. Dentro da função **modulo** utilize a função **sinal** feita na questão 1 para decidir como deve ser o retorno da função. Exemplos de saídas: **modulo(114)** deve retornar 114, **modulo(-4)** deve retornar 4 e assim por diante. Abaixo está a função, você só precisa completar seu retorno.

```
def modulo(x):  
    if sinal(x) < 0:  
        return -----  
    else:  
        return -----
```

3. A função **par** deve receber um número inteiro e retornar uma mensagem informando se ele é par ou ímpar.

4. Faça uma função chamada **verifica_ab** que recebe três números x , a e b como parâmetros e retorna 'True' se x estiver entre a e b e 'False' se ele não estiver dentro do intervalo. Exemplos:

- `verifica_ab(3, 2, 5)` deve retornar `True`
- `verifica_ab(9.5, 10, 100)` deve retornar `False`
- `verifica_ab(0, -10, 10)` deve retornar `True`.

5. Considere a função `fatorial(k)` criada em Python:

```
def fatorial(k):
    fat = 1
    if k < 0:
        return 'erro'
    i = 1
    while i < k:
        fat = fat * (i + 1)
        i += 1
    return fat
```

Desenvolva uma função chamada `combinacao` que recebe dois parâmetros (n e x) e retorna o valor da combinação $C_{n,x} = \frac{n!}{(n-x)!x!}$. Para resolver os fatoriais use a função `fatorial` definida nesta questão.

Exemplos:

- `combinacao(5, 2)` deve retornar 10
- `combinacao(9, 5)` deve retornar 126
- e assim por diante.