

# Lista 01

## Programação Funcional

Prof. Maycon Amaro

### Orientações

- O objetivo dessa lista é prepará-lo para a forma “funcional” de pensar.
- Implemente os exercícios dessa lista utilizando a linguagem C.
  - Crie todas as suas variáveis e parâmetros com `const` na frente. Exemplo: `const int x = 0`. Assim, o compilador rejeitará se você tentar modificá-las. Exceção: `scanf`.
  - Não utilize `fors`, `whiles`, `do..whiles`, `switch..cases`. Use recursão para repetir computações.
  - O `if` está permitido, mas prefira o operador de condições ternárias.
  - Não utilize funções `void`.
  - O único `include` permitido é o de `<stdio.h>`. No entanto, apenas a função `main` está permitida a utilizar as funções provenientes dessa biblioteca.
  - Não utilize vetores, ponteiros ou passagem por referência. Exceção: `scanf`.
- Se quiser feedback sobre as suas soluções, envie-a para o e-mail `maycon.amaro@aufop.edu.br`, iniciando o assunto com [BCC222].

### Exemplo

Criar um programa que calcula o fatorial de um número.

```
#include <stdio.h>

int factorial(const int x) {
    return x == 0 ? 1 : x * factorial(x - 1);
}

int main() {
```

```
int x;  
scanf("%d", &x);  
printf("%d\n", factorial(x));  
}
```

## Exercícios

1. Criar um programa que recebe um número inteiro e o multiplica por 3 se ele é par, e o multiplica por 5 se ele é ímpar.
2. Criar um programa que recebe dois números inteiros e retorna aquele com maior valor absoluto.
3. Criar um programa que recebe dois números inteiros  $x$  e  $y$ , e retorne o resultado de  $x^y$ . Considere que  $0^0 = 1$  e ignore o caso em que  $y$  é negativo.
4. Analise as funções que você escreveu: quais são parciais e quais são totais? Elas são puras (referencialmente transparentes)?

## Observação

A linguagem C não foi feita para se programar dessa forma. Não adote essas restrições para seus projetos em C, a menos que faça sentido. Essa lista de exercícios é apenas para utilizarem uma linguagem que já sabem para ver na prática alguma das restrições do paradigma funcional que estudaremos a fundo.