

**Professor:** Claudio Fabiano Motta Toledo ([claudio@icmc.usp.br](mailto:claudio@icmc.usp.br))

**Estagiário PAE:** Jesimar da Silva Arantes ([jesimar.arantes@usp.br](mailto:jesimar.arantes@usp.br))

---

Essa lista de exercícios tem como objetivo principal desenvolver algoritmos a partir dos conteúdos abordados em sala de aula. Todos os exercícios também devem ser implementados em linguagem C. Nos programas que pedem para implementar apenas funções desenvolva também o programa principal (main) para testá-los. Não utilize variáveis globais.

1. Criar uma estrutura para carros com as seguintes informações: fabricante, modelo, ano, cor, preço. Em seguida, crie as seguintes funções.
  - (a) Ler os dados de um carro.
  - (b) Imprimir os dados de um carro.
  - (c) Imprimir os dados de todos os carros.
  - (d) Modificar o fabricante para “GM” caso o registro seja Chevrolet.
2. Crie uma estrutura para armazenar dados sobre filmes com as seguintes informações: nome do filme, diretor, gênero e ano. Crie um vetor com informações para 10 filmes. Em seguida, faça as seguintes funções.
  - (a) Cadastrar um filme.
  - (b) Imprimir os filmes cadastrados.
  - (c) Buscar por filmes do gênero aventura produzidos entre 2001 e 2005.
3. Crie um registro para os funcionários de uma empresa com as seguintes informações: número do funcionário, nome, idade, telefone, cargo e salário. O programa deve manter o cadastro de até 100 funcionários. Crie funções que realizem as seguintes tarefas:
  - (a) Inserir funcionário.
  - (b) Listar todos os funcionários cadastrados.
  - (c) Procurar funcionário pelo nome ou pelo número, e imprimir seus dados.
  - (d) Eliminar o cadastro de um funcionário
  - (e) Editar as informações de um funcionário, dado o seu número de registro.
4. Utilizando a struct abaixo, escreva um programa que lê um inteiro  $n$  e crie e leia um vetor de alunos. Em seguida, faça funções para:

```
struct aluno {  
    long nusp;  
    char nome[50];  
    double nota1;  
    double nota2;  
    double nota3;  
};
```

- (a) Cadastrar um aluno. As notas dos alunos são inicializadas com zero.
  - (b) Alteração de dados de um aluno.
  - (c) Lançar as notas de um aluno.
  - (d) Exclusão de um aluno.
  - (e) Encontrar o aluno com maior média geral.
  - (f) Encontrar o aluno com menor média geral.
  - (g) Imprimir para cada aluno se ele foi aprovado ( $media \geq 5$ ) ou reprovado ( $media < 5$ ).
5. Desenvolva um programa que leia um inteiro  $n$  e crie e leia um vetor de structs com título (máximo 30 letras), autor (máximo 15 letras) e ano.
- (a) Buscar e mostrar informações através título.
  - (b) Buscar e mostrar informações através autor.
  - (c) Buscar e mostrar informações através período.
6. Qual comando você usaria **#define** ou **typedef** nas questões a seguir:
- (a) Para definir valores para **true** e **false**. Escreva como você codificaria isso.
  - (b) Para definir valores para um número complexo. Escreva como você codificaria isso.
  - (c) Para definir um número natural de 64 bits. Escreva como você codificaria isso.
  - (d) Para definir um número inteiro de 64 bits. Escreva como você codificaria isso.
7. Desenvolva um programa para manipulação de números complexos. Para isso crie um registro chamado `complexo` onde tem-se a parte real e imaginária. As seguintes funções devem ser criadas.
- (a) Adicionar números complexos.
  - (b) Subtrair números complexos.
  - (c) Adicionar número complexo com número real.
  - (d) Subtrair número complexo de número real.
  - (e) Multiplicar números complexos
  - (f) Dividir números complexos
  - (g) Determinar o módulo de um número complexo.

8. Faça as adaptações necessárias para que o código abaixo funcione, ou seja, para que ele compile e execute como um programa C. Para isso você deve usar **#define** e/ou **typedef** no local onde está indicado para se incluir o código.

```
1 #include <stdio.h>
3 //incluir codigo
5 inteiro fatorial(inteiro n);
7 inteiro principal()
  inicio
9   inteiro n = 5;
   inteiro fat = fatorial(n);
11  imprima("Fatorial(%d) = %d", n, fat);
   retorno 0;
13 fim
15 inteiro fatorial(inteiro n)
  inicio
17  se (n igual 1) inicio
    retorno 1;
19  senao
    retorno n vezes fatorial(n menos 1);
21  fim
   fim
```

Listing 1: Código C com palavras chave em português.