



Lista 14 - Structs

1. **(Exemplo com solução)** Crie uma struct chamada **Pessoa** com os campos **nome** (string de até 50 caracteres) e **idade** (inteiro). Crie uma função chamada **imprimirPessoa** que receba uma pessoa como parâmetro e imprima seus dados. No programa principal, leia os dados de uma pessoa, armazene na struct e chame a função para imprimir.

Solução sugerida:

```
#include <stdio.h>
#include <string.h>

typedef struct {
    char nome[50];
    int idade;
} Pessoa;

void imprimirPessoa(Pessoa p) {
    printf("Nome: %s\n", p.nome);
    printf("Idade: %d\n", p.idade);
}

int main() {
    Pessoa p;

    printf("Digite o nome: ");
    scanf(" %[\n]", p.nome);
    printf("Digite a idade: ");
    scanf("%d", &p.idade);

    imprimirPessoa(p);

    return 0;
}
```

2. Crie uma struct chamada **Livro** com os seguintes campos: **titulo** (string até 100 caracteres), **autor** (string até 50 caracteres) e **anoPublicacao** (inteiro). No programa principal, crie uma variável do tipo **Livro**, leia os dados de um livro, armazene na variável e depois exiba essas informações.
3. Crie um programa que permita armazenar o **nome**, a **altura** e a **data de nascimento** de até 10 pessoas. Cada pessoa deve ser representada por uma **struct** chamada **Pessoa**, e a data de nascimento deve ser outra struct chamada **Data**. Armazene essas structs em um vetor de 10 posições.

- O `nome` e a `altura` de cada pessoa devem ser informados pelo teclado.
 - A `data de nascimento` deve ser gerada aleatoriamente por uma função.
 - Para simplificar, considere que todos os meses possuem 30 dias.
 - As datas geradas devem estar no intervalo de 01/01/2000 a 30/12/2024.
4. Crie uma struct chamada `Acao` para controlar ações de uma bolsa de valores com as seguintes informações:
- `nomeCompanhia` — Nome da companhia (string de até 50 caracteres)
 - `areaAtuacao` — Área de atuação da companhia (string de até 30 caracteres)
 - `valorAtual` — Valor atual da ação (em reais, do tipo `float`)
 - `valorAnterior` — Valor da ação na abertura da bolsa (em reais, do tipo `float`)
 - `variacaoPorcentagem` — Variação da ação em porcentagem (do tipo `double`)

No programa principal, leia os dados da ação (exceto a variação), calcule a variação percentual usando a fórmula:

$$\text{variação} = \left(\frac{\text{valorAtual} - \text{valorAnterior}}{\text{valorAnterior}} \right) \times 100$$

Armazene o valor da variação na struct e exiba todos os dados da ação.

5. Crie uma struct chamada `Banda`, contendo os seguintes campos: `nome` (string de até 50 caracteres), `estiloMusical` (string de até 30 caracteres), `numeroIntegrantes` (inteiro) e `posicaoRanking` (inteiro). No programa principal, crie um vetor com 5 bandas e utilize um laço de repetição para preencher os dados de cada uma delas. Após o preenchimento, exiba todas as informações das bandas cadastradas.
6. Na questão anterior crie uma função que peça o nome de uma banda ao usuário e diga se ela está entre suas bandas favoritas ou não.
7. Crie uma struct chamada `ponto2d` que tenha como atributos `x` e `y`, ambos do tipo `float`. Crie duas variáveis do tipo `ponto2d`, chamadas `ponto_inicial` e `ponto_final`.

Implemente um menu interativo com as seguintes opções:

1. Digitar os valores do ponto inicial
2. Digitar os valores do ponto final
3. Calcular e exibir a distância entre os dois pontos
4. Sair

O menu deve ser repetido até que o usuário escolha a opção de sair.

Dica: A distância entre dois pontos (x_1, y_1) e (x_2, y_2) pode ser calculada pela fórmula:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

8. Crie uma struct chamada `Aluno` contendo os seguintes campos: `matricula` (inteiro), `nome` (string de até 50 caracteres), `nota1`, `nota2` e `nota3` (todas do tipo `float`).

Utilize um vetor para armazenar os dados de 5 alunos e implemente as seguintes funcionalidades:

- (a) Ler os dados dos 5 alunos.
 - (b) Encontrar e exibir o aluno com a maior nota na primeira prova.
 - (c) Calcular a média geral de cada aluno e exibir o aluno com a maior média.
 - (d) Exibir o aluno com a menor média.
 - (e) Para cada aluno, informar se está aprovado ou reprovado, considerando média maior ou igual a 6 como critério de aprovação.
9. Faça um programa que utilize uma struct para armazenar os dados de um funcionário de uma empresa. A estrutura deve conter os seguintes campos: nome (string de até 50 caracteres), idade (inteiro), sexo (caractere, podendo ser 'M' ou 'F'), CPF (string), data de nascimento (outra struct com dia, mês e ano), código do setor onde trabalha (inteiro entre 0 e 99), cargo (string de até 30 caracteres) e salário (float). Os dados devem ser informados pelo usuário, armazenados na struct e, em seguida, exibidos na tela.