

Aula 03 Estrutura (*struct* na Linguagem C)

Profa. Carla G. Pelissoni



Estruturas

- **Definição:** São tipos de variáveis que agrupam dados, geralmente, de tipos diferentes.
- Os itens de dados de uma estrutura são chamados **membros** e são reconhecidos por um identificador próprio.
- Pode se fazer uma analogia com um registro e seus campos.



Estruturas

- Na Linguagem C, por meio da palavra-chave **struct** podemos definir um novo **tipo** de dado.
- Após ter sido definido, o novo **tipo** existe para o compilador e pode ser utilizado para criar variáveis de modo similar a qualquer tipo simples.

Estruturas

- **Definindo um novo tipo:**

```
struct TIPOPESSOA{  
    char nome[30];  
    char sobrenome[30];  
    int idade;  
    char rg[15];  
    float salario;  
};
```

Estruturas

- **Declarando variáveis deste tipo:**

```
main()
{
    // onde TIPOPESSOA é o tipo da variável Pessoa1
    TIPOPESSOA Pessoa1;

    // É possível declarar um vetor do TIPOPESSOA
    TIPOPESSOA VetPessoa[20];
    ....
}
```

Estruturas

- **Acesso aos membros (ou campos) de uma variável do tipo *struct*:**

```
...
cout << "\nDigite seu nome: ";
cin >> Pessoa1.nome;
cout << "\nDigite seu sobrenome: ";
cin >> Pessoa1.sobrenome;
cout << "\nQual sua idade: ";
cin >> Pessoa1.idade;
cout << "\nDigite seu RG: ";
cin >> Pessoa1.rg;
cout << "\nQual seu salario mensal: ";
cin >> Pessoa1.salario;
...
```

Estruturas

- **Acesso aos membros de um vetor do tipo *struct*:**

```
...
for (i=0; i < N; i++) {
    cout << "\nDigite seu nome: ";
    cin >> VetPessoa[i].nome;
    cout << "\nDigite seu sobrenome: ";
    cin >> VetPessoa[i].sobrenome;
    cout << "\nQual sua idade: ";
    cin >> VetPessoa[i].idade;
    cout << "\nDigite seu RG: ";
    cin >> VetPessoa[i].rg;
    cout << "\nQual seu salario mensal: ";
    cin >> VetPessoa[i].salario;
}
...
```

Estruturas

- **Um membro (ou campo) de uma estrutura pode ser outra estrutura:**

```
struct ENDERECO
{
    char rua[30];
    int numero;
    char bairro[30];
    char cidade[30];
    char estado[2];
    char cep[10];
};

struct TIPOPESSOA
{
    char nome[30];
    char sobrenome[30];
    ENDERECO end;
    int idade;
    char rg[15];
    float salario;
};
```

Estruturas

```
main()
{
    TIPOPESSOA Pessoa1;
    ....
    cout << "\nDigite o seu endereço:
    \nRua: ";
    cin >> Pessoa1.end.rua;
    cout << "\nNumero: ";
    cin >> Pessoa1.end.numero;
    cout << "\nCidade: ";
    cin >> Pessoa1.end.cidade;
    ...
}
```

Estruturas



- **Exercício 1:** Faça um programa em C que leia os dados (nome, sobrenome, idade, RG e salário) **de uma pessoa** e armazene numa estrutura. Em seguida, verifique se essa pessoa tem mais de 50 anos. Em caso positivo, seu salário deve ter um acréscimo de 20%. Como resultado escreva nome e sobrenome da pessoa, seu RG, seu salário e uma frase dizendo se teve aumento ou não.
- **Exercício 2:** Faça o mesmo programa anterior, agora para N pessoas (N deve ser lido antes e ser no máximo 50).

Estruturas

- **Exercício 3:** Faça um programa em C para gerenciar contas bancárias. Defina o tipo CONTA que contenha os seguintes campos: número da conta, CPF do cliente, nome do cliente e saldo. Seu programa deve ler os dados de N clientes (N deve ser lido antes e ser no máximo 50) e imprimir na tela nome, CPF e número da conta de todos os clientes do banco que estão com saldo negativo.