

# Núcleo Básico das Engenharias

C202E/H

Algoritmos e Estruturas de Dados I

## 09 – Registros (*structs*) – (part1)

Prof. Edson J. C. Gimenez

2019/Sem1

*Material adaptado de: Algoritmos e Estruturas de Dados*  
- Profa. Rosanna Mara Rocha Silveira  
- Prof. Evandro Luís Brandão Gomes

### **VARIÁVEIS COMPOSTAS HETEROGÊNEAS – REGISTROS**

→ São estruturas que podem agrupar diferentes informações gerando novos tipos de dados.

→ Pode ser visto como um conjunto de campos onde cada campo pode ser de um mesmo tipo de dado ou de tipos de dado diferente.

Exemplo: uma ficha de cadastro de um cliente

Código do Cliente:
Nome do Cliente:
Endereço:
Saldo:

## Declaração de uma struct em C/C++

```
struct nome_struct  
{  
    tipo1 campo1;  
    tipo2 campo2;  
    :  
    :  
    tipo campoN;  
};
```

sendo:

- nome\_da\_struct: nome válido associado ao novo tipo criado;
- campo1, campo2, ..., campoN são os nomes dos campos que irão compor a struct;
- **tipo1, tipo2, ..., tipoN**: são os tipos de dados associados a cada campo da struct.

Ex1 (ap): para a ficha mostrada no exemplo anterior:

```
struct fichacad  
{  
    int codigo;  
    char nome[30];  
    char endereco[40];  
    float saldo;  
};
```

## Declaração de uma struct em C/C++

A declaração de uma variável do tipo struct pode ser feita de duas formas:

- Após criada a struct, esta é usada como o **tipo de variável**, na declaração da variável;
- A variável já é declarada juntamente com a criação da struct.

**Exemplo 2:** declarando uma variável ficha como sendo do tipo fichacad, a struct do exemplo anterior:

```
struct fichacad  
{  
    int codigo;  
    char nome[30];  
    char endereco[40];  
    float saldo;  
};
```

**fichacad** ficha;

```
struct fichacad  
{  
    int codigo;  
    char nome[30];  
    char endereco[40];  
    float saldo;  
} ficha;
```

### **Uso de uma variável do tipo struct em C/C++**

- A variável criada (no exemplo, **ficha**) pode ser acessada normalmente como qualquer outra variável (pelo nome dela, mas é preciso informar o nome da variável e, separado por ponto ( . ), o nome do campo desejado.

Exemplos:

#### **1) Como atribuir um valor à uma posição (campo) do REGISTRO:**

```
cin >> ficha.codigo;  
cin.getline(ficha.nome, 30);  
cin.getline(ficha.endereco, 30);  
...  
ficha.saldo = 100.0;
```

#### **2) Como atribuir valores a um registro:**

```
fichacad ficha = {1, "Jose da Silva", "Rua 2, no. 55 – Abaeté", 1200.00};
```

#### **3) Como se mostra o elemento contido em uma posição (campo) do REGISTRO:**

```
cout << ficha.saldo;
```

### **Exemplo 3**

```
int main( )  
{  
    setlocale(LC_ALL, "Portuguese");  
    struct fichacad  
    {  
        int codigo;  
        char nome[30];  
        char endereco[40];  
        float saldo;  
    };  
    fichacad ficha;          //declaração da variável ficha junto da struct  
    //entrando com os dados da pessoa  
    cout<<"Código :";  
    cin>>ficha.codigo;  
    cin.ignore();  
    cout<<"Nome :";  
    cin.getline(ficha.nome,30);  
    cout<<"Endereço :"; cin.getline(ficha.endereco,40);  
    cout<<"Saldo :"; cin>>ficha.saldo;          // continua.....
```

```
:  
:  
//mostrando os dados da pessoa  
cout<<"\nDados da pessoa:" << endl;  
cout<<"Código: " <<ficha.codigo << endl;  
cout<<"Nome: " <<ficha.nome << endl;  
cout<<"Endereço: " <<ficha.endereco << endl;  
cout<<"Saldo: " <<ficha.saldo << endl;  
  
return 0;  
}
```

**Exemplo 4:** Deseja-se, para um dado aluno, ler sua **matrícula (int)** e suas **duas notas (float)** e, a seguir, calcular e mostrar sua **média** e sua condição: “APROVADO”, se média  $\geq 60$ , ou “REPROVADO”, caso contrário.

- declaração do registro ficha:

```
struct ficha  
{  
    int nmat,  
    float np1, np2, md;  
};
```

- declaração da variável aluno (tipo ficha): **ficha** aluno;

```
#include <iostream>
#include <locale>
using namespace std;
//exemplo sem uso de struct
int main( )
{
    setlocale(LC_ALL, "Portuguese");
    struct ficha //declaração da struct
    {
        int nmat;
        float np1, np2, md;
    } aluno; //declaração da variável aluno junto com a struct

    //entrada de dados
    cout << "Matricula: ";
    cin >> aluno.nmat;
    cout << "Nota 1: ";
    cin >> aluno.np1;
    cout << "Nota 2: ";
    cin >> aluno.np2;

    // continua.....
```

```

    :
    :
    aluno.md = (aluno.np1 + aluno.np2)/2; // cálculo da média

    // saída de dados
    if (aluno.md >= 60)
        cout << "Media: "<<aluno.md<<" - APROVADO!";
    else cout << "Media: "<<aluno.md<<" - REPROVADO!";

    return 0;
}
```