



Introdução a **Structs** em C

Prof. Fábio Takeda

Comando TypeDef

- o Este comando possibilita nomes alternativos para tipos existentes:
- o Neste caso, será utilizado para remover a necessidade de utilizar o comando **struct** (a seguir) na declaração de uma variável.
- o O exemplo será realizado juntamente com o estrutura que será apresentada a seguir.



Criando Estrutura no C

O que é uma estrutura?

Uma estrutura em linguagem de programação tem o objetivo de agrupar duas ou mais variáveis de mesmo ou tipos diferentes.

Como é a sintaxe de uma estrutura?

- o Basicamente pode ser realizada de duas formas básicas:
 - o Criando as variáveis logo abaixo da estrutura;
 - o Criando uma nome o qual será utilizado para definir o tipo de uma variável local ou global;

Criando as estruturas com as variáveis no fim

- o Neste caso, será definido o seguinte exemplo:
 - o Será utilizada a variável de estrutura coordenada. Está que será utilizado durante a programação para acessar o registro.
 - o Esta estrutura irá armazenar as informações de uma coordenada formada por duas variáveis do tipo inteiro: x e y

Criando as estruturas com as variáveis no fim

◦ Sendo assim tem-se:

```
struct
{
    int x;
    int y;
} coordenada1,
coordedana2;
```

Criando `struct` como tipo

- Outra forma de criar estruturas é definindo a estrutura como um tipo.
- A vantagem deste método é poder criar as variáveis conforme sua necessidade podendo ser declaradas como locais, globais ou variáveis de passagem e parâmetros.
- Estas duas últimas caso a estrutura tenha sido definida como global.

Criando `struct` como tipo

- A declaração da estrutura neste modelo pode ocorrer de duas formas básicas:
- Sem o comando `typedef`
- Com o comando `typedef`.

Criando `struct` como tipo (sem `typedef`)

- Neste caso, temos a definição da estrutura e das variáveis da seguinte forma:

```
struct ponto
{
    int x;
    int y;
};
```

```
struct ponto coordenada;
```

Criando struct como tipo (com typedef)

- Neste caso, temos a definição da estrutura e das variáveis da seguinte forma:

```
typedef struct ponto  
{  
    int x;  
    int y;  
};
```

```
ponto coordenada;
```


Importante...

- o Uma vez definida uma estrutura como um tipo é possível aplicar a mesma regra de qualquer tipo pré-definido na linguagem C como por exemplo, a definição de vetores e ponteiros
- o Utilizando o exemplo anterior, a definição de um vetor ficaria da seguinte forma:

```
ponto coordenada[10];
```

Acessando sub variáveis de uma estrutura

- o Para acessar uma sub variável é necessário utilizar o operador (.)
- o Ex:
 - o `coordenada.x = 10;`
 - o `scanf("%d",&coordenada.y);`
- o Assim, tem-se o nome da variável que é do tipo da estrutura, operador (.) e a variável declarada dentro da estrutura.

Acessando sub variáveis de uma estrutura

- o No caso se um vetor a regra é a mesma.
- o Ex:
 - o coordenada[1].x
 - o coordenada[2].y
- o Assim, é possível perceber que, para cada posição do vetor, a estrutura terá seu conjunto de variáveis.