

Linguagem C

Estruturas de Dados
Heterogêneas (struct)

Estruturas (struct)

- “structs” são **Estruturas de Dados Heterogêneas**
- Uma estrutura agrupa várias variáveis numa só.
- Funciona como uma espécie de “ficha”. A ficha armazena diversos dados relacionados, porém de tipos diferente.
- A estrutura, então, serve para agrupar um conjunto de dados não similares, formando um novo tipo de dados.

Estruturas (struct)

- Declaração:

```
struct nome_do_tipo_da_estrutura
{
    tipo_campo1 nome_campo1;
    tipo_campo2 nome_campo2;
    ...
}    variáveis_que_armazenam_a_estrutura;
```

Onde:

- tipo_campo1, tipo_campo2: tipos de dados em C (char, int, float, double, char[])

Estruturas (struct)

- Exemplo:

```
struct tipo_ficha_cadastral
{
    char nome[50];
    long int RA;
    int idade;
}    aluno1, aluno2;
strcpy(aluno1.nome , "Joao da Silva");
aluno1.RA = 992042;
aluno1.idade = 21;
strcpy(aluno2.nome , "Jose dos Santos");
aluno2.RA = 899932;
aluno2.idade = 19;
```

Estruturas (struct)

- Exercício 1:
 - Criar um programa que permita ao usuário digitar os dados de 3 alunos, (seus nomes, idades e registros acadêmicos). Em seguida, pedir ao usuário para digitar uma idade, e o programa deve imprimir os nomes e idades dos alunos com idade menor que a digitada;

Estruturas (struct)

- Exercício 1(cont):
 - Criar um programa que permita ao usuário digitar os dados de 3 alunos, (seus nomes, idades e registros acadêmicos). Em seguida, pedir ao usuário para digitar uma idade, e o programa deve imprimir os nomes e idades dos alunos com idade menor que a digitada;
 - Altere o programa para que ele use uma função para ler os dados dos acadêmicos do teclado;
para isto, crie uma função com o seguinte protótipo:

```
                                struct  
tipo_ficha_cadastral le_dados_aluno();
```

Estruturas (struct)

- Exercício 1 (cont):
 - Criar um programa que permita ao usuário digitar os dados de 3 alunos, (seus nomes, idades e registros acadêmicos). Em seguida, pedir ao usuário para digitar uma idade, e o programa deve imprimir os nomes e idades dos alunos com idade menor que a digitada;
 - Altere o programa para que ele use uma função para ler os dados dos acadêmicos do teclado; para isto, crie uma função com o seguinte protótipo:

struct

tipo_ficha_cadastral le_dados_aluno();
 - Altere o programa para que ele use uma função para escrever os dados dos acadêmicos na tela; para isto, crie uma função com o seguinte protótipo:

void

imprime_dados_aluno(struct tipo_ficha_cadastral aluno);

Estruturas (struct)

- Exercício 2:
 - Criar um programa que permita ao usuário digitar os dados de 30 alunos, (seus nomes, idades, registros acadêmicos e). Em seguida, pedir ao usuário para digitar uma idade, e o programa deve imprimir os nomes e idades dos alunos com idade menor que a digitada;
 - (Para isto, será necessário criar um vetor de alunos.....)

Estruturas Compostas(struct)

- Declaração Estruturas Compostas:

```
struct nome_do_tipo_da_estrutura
{
    struct nome_de_outra_estrutura nome_campo1;
    tipo_campo2 nome_campo2;
    ...
} variáveis_que_armazenam_a_estrutura;
```

- Desta forma, uma estrutura pode ser parte de outra estrutura!

Estruturas Compostas(struct)

- Exemplo:

```
struct tipo_data
{
    int dia, mes, ano;
}
struct tipo_ficha_cadastral
{
    char nome[50];
    long int RA;
    struct tipo_data nascimento;
}    aluno1;

strcpy(aluno1.nome , "Joao da Silva");
aluno1.RA = 992042;
aluno1.nascimento.ano=1990;
```

Estruturas (struct)

- Exercício 3:
 - Criar um programa em C que leia 4 pontos num plano cartesiano, e o armazene na estrutura “retângulo” definida abaixo, verificando se os 4 pontos formam realmente um retângulo.

```
struct ponto {  
    int x;  
    int y;  
};  
struct retangulo {  
    struct ponto p1;  
    struct ponto p2;  
    struct ponto p3;  
    struct ponto p4;  
} ret1;
```

Estruturas (struct)

- Exercício 4:
 - Criar um programa em C que leia 4 pontos num plano cartesiano, e o armazene na estrutura “quadrilatero” definida abaixo, verificando se os 4 pontos formam um retângulo, um quadrado ou um losango (**Losango é um quadrilátero cujos lados são de igual comprimento. Trançando-se suas diagonais é possível dividi-lo em quatro triângulos retângulos simétricos.**).

```
struct ponto {  
    int x;  
    int y;  
};  
struct quadrilatero {  
    struct pontos p[4];  
} quadrilatero1;
```

Estruturas (struct)

- Exercício 5:
 - Criar um programa em C que armazene os dados de todos os alunos da turma, e permita os cálculos de:
 - Idade de todos os alunos(sendo dada uma data atual);
 - Médias de todos os alunos;
 - Listagem dos alunos que ficaram para final(média entre 4 e 7);

```
struct tipo_nota
{
    int dia, mes, ano;
}
struct tipo_ficha_cadastral
{
    char nome[50];
    struct tipo_data nascimento;
    float notas[2];
}alunos[22];
```