

Introdução à Ciência da Computação

Registros em C

Prof. Ricardo J. G. B. Campello

Agradecimentos

◆ Parte dos slides a seguir são adaptações dos originais:

- de A. L. V. Forbellone e H. F. Eberspächer
- do Prof. Rudinei Goularte

Sumário

- Conceito de Registro
- Implementação em C (structs)
- Arranjos de Registros

Registros

■ Definição:

- Um **registro** é uma **variável composta heterogênea**
 - É um conjunto de dados estruturados, os quais podem ser de tipos diferentes
- Os dados em um registro são representados através de variáveis ou constantes, normalmente chamadas de **campos**

Registros

- ✦ Razão da denominação (variável composta heterogênea):
 - ✦ Podem ser compostos de variáveis (campos) de **tipos diferentes**
 - Caso mais trivial: apenas tipos simples (primitivos)
 - Porém, veremos que um campo pode também ser um arranjo ou até outro registro!
- ✦ Exemplo 1: Passagem de ônibus

Número: _____

Origem: _____ Destino: _____

Data: ____ / ____ / ____ Horário: _____ : _____

Poltrona: _____ Preço: _____

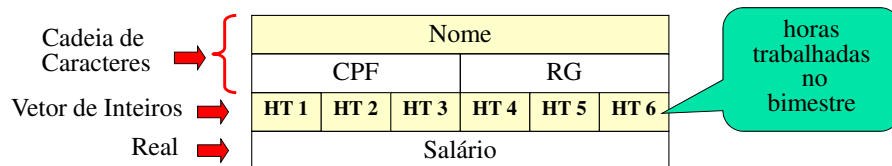
Registros

- Exemplo 2: Registro de Pagamento
 - Dados:
 - Nome, RG, CPF, salário, horas trabalhadas, etc...
 - Dados Estruturados (Estrutura de Dados):
 - Uma variável (campo) para cada dado, mas...
 - Um identificador comum para o conjunto (registro)

6

Registros

■ Exemplo 2: Registro de Pagamento



- **note que um dos campos é um vetor**
 - combinação de variáveis compostas heterogênea e homogênea

7

Declaração de Registros

Declaração de Registro Único:

```
struct {  
    tipo_1 campo(s);  
    tipo_2 campo(s);  
    ...  
    tipo_n campo(s);  
}; identificador;
```

8

Declaração de Registros

■ Exemplo:

```
struct {  
    char NOME [30];  
    char CPF[12];  
    char RG[10];  
    int HT[6];  
    float SALARIO;  
}; Reg_Pag1;
```

9

Declaração de Registros

Declaração de Múltiplos Registros (Forma 1):

```
struct tipo_registro {  
    tipo_1 campo(s);  
    tipo_2 campo(s);  
    ...  
    tipo_n campo(s);  
}; identificadores;
```

10

Declaração de Registros

■ Exemplo:

```
struct Registro_Pagamento {  
    char NOME [30];  
    char CPF[12];  
    char RG[10];  
    int HT[6];  
    float SALARIO;  
}; Reg_Pag1, Reg_Pag2, Reg_Pag3;
```

11

Declaração de Registros

Declaração de Múltiplos Registros (Forma 2):

```
struct tipo_registro {  
    tipo_1 campo(s);  
    tipo_2 campo(s);  
    ...  
    tipo_n campo(s);  
};  
  
struct tipo_registro identificadores;
```

12

Declaração de Registros

Exemplo:

```
struct Registro_Pagamento {  
    char NOME [30];  
    char CPF[12];  
    char RG[10];  
    int HT[6];  
    float SALARIO;  
};  
  
struct Registro_Pagamento Reg_Pag1, Reg_Pag2, Reg_Pag3;
```

13

Declaração de Registros

Declaração de Múltiplos Registros (Forma 3):

```
typedef struct {  
    tipo_1 campo(s);  
    tipo_2 campo(s);  
    ...  
    tipo_n campo(s);  
} tipo_registro;  
  
tipo_registro identificadores;
```

14

Declaração de Registros

Exemplo:

```
typedef struct {  
    char NOME [30];  
    char CPF[12];  
    char RG[10];  
    int HT[6];  
    float SALARIO;  
} Registro_Pagamento;  
  
Registro_Pagamento Reg_Pag1, Reg_Pag2, Reg_Pag3;
```

15

Registros como Campos

- Um registro pode conter campos que são registros
- Exemplo:
 - Use **typedef** para declarar um **tipo** de registro *address* com os campos Rua, No, Cidade, UF, CEP
 - Use o novo tipo declarado para incrementar o **tipo** de registro *Registro_Pagamento* declarado nos exemplos anteriores, de forma que este contenha um campo do tipo *address* para o endereço do empregado

16

Registros como Campos

➤ Solução:

```
typedef struct {
    char Rua[20], Cidade[15], UF[3];
    int No, CEP;
} address;

typedef struct {
    char NOME[30], CPF[12], RG[10];
    int HT[6];
    float SALARIO;
    address ENDERECO;
} Registro_Pagamento;
```

17

Exercícios

- Declare variáveis do tipo "Registro_Pagamento" definido no exemplo anterior
- Declare um tipo registro para o exemplo da passagem de ônibus visto anteriormente, definindo os campos data (dia, mês e ano) e horário (hora e minuto) como outros registros

18

Manipulação de Registros

- Como acessar os campos de um registro?
 - *nome_do_registro . nome_do_campo*

- Exemplos de Atribuição:

Reg_Pag1.HT[0] = 163;

Vetor de Inteiros

Reg_Pag1.SALARIO = 852.7;

Real

19

Manipulação de Registros

- Exemplo de Manipulação:

```
printf("Entre o nome do empregado: ");
scanf("%s", Reg_Pag1.NOME);
printf("Entre as horas trabalhadas no bimestre: ");
for (i=0; i<=5; i++) scanf("%d", &Reg_Pag1.HT[i]);
printf("Entre o salario: ");
scanf("%f", &Reg_Pag1.SALARIO);
```

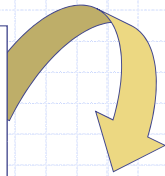
- **Nota:** C ANSI admite a atribuição direta entre estruturas de um mesmo tipo. Por exemplo, Reg_Pag2 = Reg_Pag1

20

Manipulação de Registros

- ◆ Quando um dos campos é outro registro:

```
struct {
    char NOME[30], CPF[12], RG[10];
    int HT[6];
    float SALARIO;
    struct {
        char Rua[20], Cidade[15], UF[3];
        int No, CEP;
    } ENDERECO;
} Reg_Pag1;
```



Atribuições

```
Reg_Pag1.SALARIO = 1200;
...
Reg_Pag1.ENDERECO.No = 230;
Reg_Pag1.ENDERECO.CEP = 15980;
...
```

21

Manipulação de Registros

- E se tenho 500 empregados?
 - Declaro 500 variáveis do tipo registro???

22

Arranjos de Registros

- ◆ Combinação de estruturas homogêneas com heterogêneas
 - ◆ vetores ou matrizes cujas células são registros
- ◆ Exemplo 1: ônibus composto por passageiros

1

2

3

4

44

Número: _____ Nome: _____ RG: _____

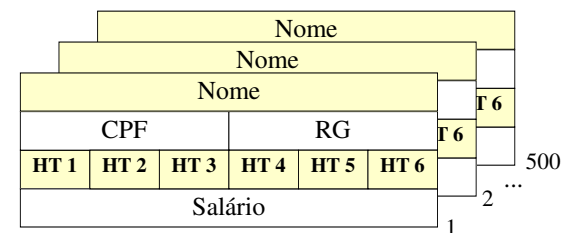
Origem: _____ Destino: _____

Data: ____ / ____ / ____ Horário: _____ : _____

Poltrona: _____ Preço: _____

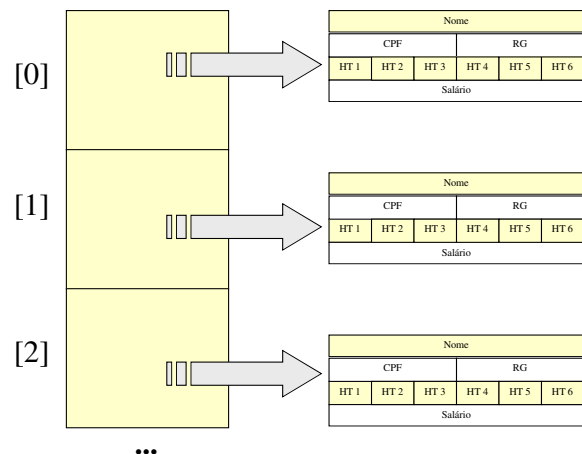
Arranjos de Registros

- Exemplo 2: Registro de Pagamento



24

Arranjos de Registros



25

Declaração

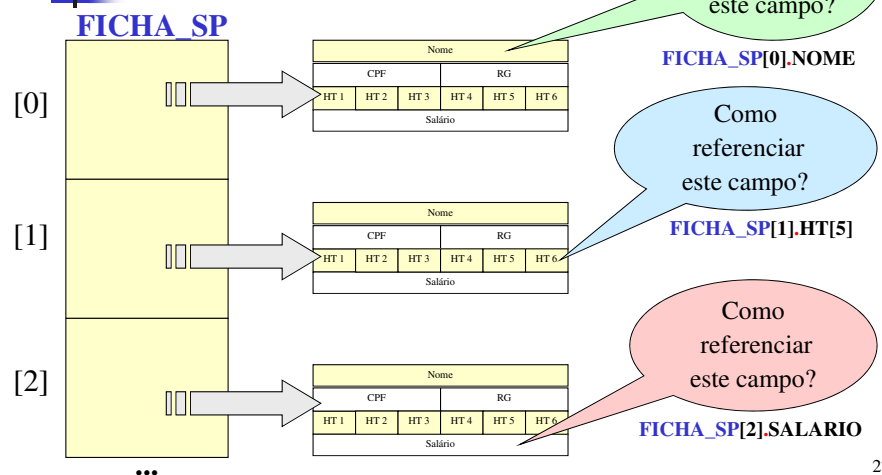
Exemplo:

```
typedef struct {
    char NOME [30];
    char CPF[12];
    char RG[10];
    int HT[6];
    float SALARIO;
} Registro_Pagamento;

Registro_Pagamento Ficha_SP[500], Ficha_RJ[300], Ficha_MG[100];
```

26

Arranjos de Registros



27

Arranjos de Registros

Exemplos de Atribuição:

- ♦ `FICHA_SP[2].SALARIO = 243.45;`
- ♦ `FICHA_SP[6].HT[3] = 228;`
- ♦ `scanf("%s", FICHA_SP[1].NOME);`

...

28

Exercício 1

- Fazer um programa, em C, que contenha a declaração de um registro PESSOA, contendo:
 - Nome (*nome*): até 30 caracteres;
 - Sexo (*sexo*): 'M' ou 'F';
 - Data de nascimento (*data_nas*): dd/mm/aa;
 - Número de dependentes (*n_dep*).
- O programa deve pedir ao usuário que digite as informações de modo a preencher o registro. As informações do registro devem ser escritas na tela

Exercício 2

- Generalize o programa do Exercício 1 de tal forma que ele permita registrar 300 funcionários
 - Não necessariamente o programa deve registrar todos os 300 funcionários. Logo, permita ao usuário uma opção de interromper o cadastro

Exercício 3

- ◆ Uma determinada biblioteca possui obras de ciências exatas, ciências humanas e ciências biomédicas, totalizando 1500 volumes, 500 de cada área. O proprietário resolveu informatizá-la e, para tal, agrupou as informações sobre cada livro do seguinte modo:

Código de Catalogação (4 dígitos): _____ Ano: _____
Dado: () Sim () Não No. de Páginas: _____
Nome da Obra: _____
Nome do Autor: _____ Editora: _____

1. Construa um programa que declare tal estrutura de forma que as informações das obras possam ser organizadas em três colunas de uma matriz, uma para cada área

Exercício 4

- ◆ Uma determinada biblioteca possui obras de ciências exatas, ciências humanas e ciências biomédicas, totalizando 1500 volumes, 500 de cada área. O proprietário resolveu informatizá-la e, para tal, agrupou as informações sobre cada livro do seguinte modo:

Código de Catalogação (4 dígitos): _____
Dado: () Sim () Não No. de Páginas: _____
Nome da Obra: _____
Nome do Autor: _____ Editora: _____

2. Complemente o programa do exercício anterior para realizar uma consulta: O usuário deve fornecer o código da obra e sua área. Escreve-se então os campos do registro correspondente, ou que a obra não existe. Código -1 encerra o algoritmo

Bibliografia

- ◆ Schildt, H. "C Completo e Total", 3a. Edição, Pearson, 1997.
- ◆ Damas, L. "Linguagem C", 10a. Edição, LTC, 2007