Universidade Federal de Rondônia

Estrutura de Dados I

**Lista 7 – Registros (Structs)**

Até agora, as variáveis vistas eram

- simples: definidas por tipo **int**, **float**, **double**, **char**;

- compostas homogêneas: vetores e matrizes.

Agora veremos estrutura de dados heterogêneos: struct.

Essas estruturas podem possuir variáveis de tipos diferentes e são declaradas da seguinte forma:

typedef struct{

tipo1 id1;

tipo2id2;

...

tipon idn;

} novo\_tipo;

Vamos ver um exemplo de uma estrutura para cadastro de um produto:

typedef struct{

char cod[8];

char nome[100];

int quant;

float custo;

}produto; // nome da nova estrutura

O nome deste novo tipo de dado é **produto**. Para declarar uma variável do tipo produto, basta fazer:

produto prod;

O acesso às variáveis se dá da seguinte maneira:

prod.cod = ‘12345678’;

prod.nome = ‘arroz’;

prod.quant = 1000;

prod.custo = 0.82;

Observe que para acessar os campos, usamos o operador ponto (.)

registro.campo

E, se ao invés de um único produto, quisermos cadastrar n produtos?

A solução é criar um vetor de estruturas...

A declaração de um vetor do tipo produto será:

produto P[5];

Na declaração anterior, **produto** é o nome da estrutura, **P** é a variável e **5** é a quantidade de produtos máxima que podem ser cadastrados.

Resumindo...

Estruturas (struct): define um “conjunto” de variáveis que podem ser de tipos diferentes;

Vetor : é uma “lista” de elementos de mesmo tipo. Por exemplo, a seguinte declaração:

produto lista[5];

declara uma variável **lista** contendo elementos do tipo **produto**.

**Exercício 1**:

O programa abaixo lê 10 fichas de empregados de uma empresa. Cada ficha possui o seguinte formato



Observações:

- O campo **DATAEMP** deve ser dividido em 3 variáveis: dia, mês, ano;

- O campo **NUMEMP** é do tipo **int**;

- O campo **NOMEMP**, **ENDEREÇO** é do tipo **char**;

- O campo **SALÁRIO** é do tipo **float**.

**typedef struct**{

**int** numemp;

**char** nomemp[20], endereco[20];

**int** dia, mes, ano;

float salario;

}ficha;

**ficha** vfichas[10];

**int** i;

int main(){

int i;

for (i=0;i<=10;i++){

printf("\nentre com o numero do empregado");

scanf("%d",&fichas[i].numemp);

printf("\nentre com o nome do empregado");

scanf("%s", fichas[i].nomemp);

printf("\nentre com o endereco do empregado");

scanf("%s",fichas[i].endereco);

getchar(); //de string para inteiro eh necessario

printf("\nentre com o dia da admissao");

scanf("%d",&fichas[i].dia);

printf("\nentre com o mes da admissao");

scanf("%d",&fichas[i].mes);

printf("\nentre com o ano da admissao");

scanf("%d",&fichas[i].ano);

getchar(); //de string para inteiro e real eh necessario

printf("\nentre com o salario do empregado");

scanf("%f",&fichas[i].salario);

}\\end for

for (i=0;i<=10;i++){

printf("%d\n",fichas[i].numemp);

printf("%s\n", fichas[i].nomemp);

\_ printf("%s\n",fichas[i].endereco);

\_ printf("%d\n",fichas[i].dia);

\_ printf("%d\n",fichas[i].mes);

printf("%d\n",fichas[i].ano);

\_ printf("%f\n",fichas[i].salario);

} // end for

\_ getch();

\_ } // end main

Execute o programa acima.

**Exercício 2 (Entregar)**: Faça um programa que faz uma ficha de dados de 7 pessoas. A estrutura deve conter os seguintes campos: RG, Nome da Pessoa, Rua, número da casa.

**Exercício 3 (Entregar)**: 