

PC1 - Aula 29

Registros

Hebert Coelho

Instituto de Informática
Universidade Federal de Goiás

Roteiro

- Registros
- Exercícios

Registro

Definição

Um registro é uma estrutura que contém diversas variáveis (chamadas de campos), usualmente de tipos diferentes, mas que dentro de um determinado contexto, fazem sentido se agrupadas.

- Podemos comparar um registro com uma ficha que possui todos os dados sobre uma determinada entidade, por exemplo: Registro de alunos (nome, RA, médias de provas, médias de labs, etc...)
- Registro de produtos (Nome, código, descrição, etc...)

Declarando o formato do registro

Para criar um registro é preciso declarar seu formato/estrutura. Isso é feito utilizando a palavra chave struct, da seguinte forma:

```
struct nome_do_tipo_do_registro {  
    tipo_1 nome_1;  
    tipo_2 nome_2;  
    tipo_3 nome_3;  
    ...  
    tipo_n nome_n;  
};
```

Declarando o formato do registro

- O registro pode ser declarado dentro ou fora da função **main**. Normalmente é feita fora da **main**, conforme exemplo.
- A próxima etapa é declarar uma variável do tipo **struct nome do tipo da estrutura**.
- Podemos acessar individualmente os campos de um determinado registro como se fossem variáveis normais.

```
#include <stdio.h>
struct aluno {
    int mat; //matricula
    float media;
};
int main () {
    struct aluno j;
    j.mat = 10;
    j.media = 8.5;
    printf("Matricula %d, media %f.", j.mat, j.media);
}
```

Utilizando os campos de um Registro

- Para o registro declarado anteriormente, utilizamos

`j.mat`

para acessar o campo `mat` do registro `j` (note que usamos o nome da variável e não o nome dado ao formato do registro).

- Podemos colocar o campo de um registro em qualquer lugar onde colocaríamos uma variável.

Lendo os campos de um Registro

A leitura dos campos de um registro a partir do teclado deve ser feita campo a campo, como se fossem variáveis independentes.

```
printf ("Digite a matricula do aluno: ");  
scanf ("%d", &j.mat);  
printf ("Digite a média do aluno: ");  
scanf ("%f", &j.media);
```

Copiando registros

A cópia de um registro pode ser feita como se fosse a cópia de uma variável normal, ou seja

```
registro_1 = registro_2;
```


Vetor de registros

Registros podem ser declarados como um vetor.

```
#include <stdio.h>
struct aluno {
    int mat; //matricula
    float media;
};
int main () {
    struct aluno turma[40];
    turma[0].mat = 10;
    turma[0].media = 8.5;
    ...
}
```

Registros Aninhados

Pode-se também declarar um registro como uma das variáveis de outro registro, quantas vezes isso for necessário.

```
#include <stdio.h>
struct medias {
    float p1;
    float p2;
    float p3;
};
struct ficha{
    int ra;
    struct medias provas;
}
```

Registros

Faça um programa que faça o cadastro dos alunos da turma de PC1, o número máximo de alunos na turma são 50. Para cadastrar um aluno, o usuário deverá digitar o número de matrícula, o nome do aluno, as notas de 3 provas e a quantidade de presenças. O seu programa deverá realizar o cadastro dos alunos, fazer a consulta de um aluno por número de matrícula ou nome, imprimir a lista dos alunos com conceito aprovado ou reprovado.

Obs.:O aluno será considerado aprovado se a Média for maior que seis e frequência maior que 75%, sendo que o número de aulas é de 64 horas aulas.