

Linguagem C *Struct*

Prof. Edwar Saliba Júnior
Maio de 2011

Tipo Estruturado de Dados

- Tipos estruturados:
 - formam novos tipos de dados a partir dos tipos já existentes;
 - são entidades de programação que permitem o armazenamento de informação de diferentes tipos em um único local, denominado **estrutura**;
 - uma estrutura é formada por **campos**. Cada campo possui parte da informação;
 - um campo pode ser de um tipo primitivo (`int`, `float`, `char`, `double` e etc.) ou uma estrutura definida previamente;
 - ao final de sua declaração, a estrutura deve receber um nome, definindo assim um **novo tipo de dados**.

Estruturas

- Permitem modelar entidades compostas por mais de uma informação. Exemplo:

- Ponto no R^3 : $P = (x, y, z)$.

```
typedef struct {  
    float x;  
    float y;  
    float z;  
} ponto;
```

- Número complexo: $z = (Re, Im)$.

```
typedef struct {  
    float Re;  
    float Im;  
} complexo;
```

- Informação de aluno.

```
typedef struct {  
    char nome[80];  
    int matricula;  
    char curso[30];  
    int creditos;  
    float CR;  
} aluno;
```

Trabalhando com Estruturas

- Uma vez definida a estrutura, ela pode ser utilizada como um novo tipo de dados:

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  typedef struct{
5      char marca[30];
6      char modelo[30];
7      int ano;
8      float preco;
9  } automovel;
10
11 int main()
12 {
13     automovel carro;
14
15     printf("Digite a marca do carro: ");
16     gets(carro.marca);
17     printf("Digite o modelo do carro: ");
18     gets(carro.modelo);
19     printf("Digite o ano do carro: ");
20     scanf("%d",&carro.ano);
21     printf("Digite o preço do carro: ");
22     scanf("%f",&carro.preco);
23
24     printf("\n\nO carro digitado foi:");
25     printf("\nMarca : %s", carro.marca);
26     printf("\nModelo: %s", carro.modelo);
27     printf("\nAno   : %d", carro.ano);
28     printf("\nPreço  : %f", carro.preco);
29
30     return 0;
31 }
```

Exemplo

Calcular a distância entre dois pontos P e Q do R^2 , de acordo com a fórmula:

$$\text{dist}(P, Q) = \sqrt{(x_P - x_Q)^2 + (y_P - y_Q)^2}$$

```
1  #include <stdio.h>
2  #include <math.h>
3
4  typedef struct{
5      float x;
6      float y;
7  } ponto;
8
9  int main()
10 {
11     ponto P, Q;
12     float dist = 0;
13
14     printf("Digite a coordenada X do ponto P: ");
15     scanf("%f", &P.x);
16     printf("Digite a coordenada Y do ponto P: ");
17     scanf("%f", &P.y);
18     printf("Digite a coordenada X do ponto Q: ");
19     scanf("%f", &Q.x);
20     printf("Digite a coordenada Y do ponto Q: ");
21     scanf("%f", &Q.y);
22
23     dist = sqrt(pow(P.x - Q.x, 2) + pow(P.y - Q.y, 2));
24
25     printf("\n\nDistância(P,Q) = %f ", dist);
26
27     return 0;
28 }
29
```

Exemplo com Vetor

- Programa que armazena num vetor os dados dos 40 alunos de uma classe e os imprime em seguida.

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  typedef struct{
5      char nome[100];
6      char matricula[30];
7      char curso[50];
8  } aluno;
9
10 int main()
11 {
12     aluno classe[40];
13     int i;
14
15     for(i = 0; i < 40; i++){
16         printf("\n\nDigite o nome do aluno: ");
17         gets(classe[i].nome);
18         printf("Digite a matrícula do aluno: ");
19         gets(classe[i].matricula);
20         printf("Digite o curso do aluno: ");
21         gets(classe[i].curso);
22     }
23
24     printf("\n\nAlunos Cadastrados:");
25     for(i = 0; i < 40; i++){
26         printf("\nMatrícula: %s", classe[i].matricula);
27         printf("\nNome      : %s", classe[i].nome);
28         printf("\nCurso       : %s\n\n", classe[i].curso);
29     }
30
31     return 0;
32 }
```

Exercício

- Uma locadora de automóveis pretende informatizar seu cadastro para gerenciar sua frota composta por 200 veículos. Para isto pretende reunir as informações relevantes na seguinte estrutura:

```
1  typedef struct{
2      char marca[30];
3      char modelo[30];
4      int ano;
5      float km;
6      int portas;
7      char cor[20];
8  } automovel;
9
```

- Fundamentado na estrutura proposta, escreva um programa em linguagem C que informe:
 - O número de automóveis de determinada marca;
 - A quilometragem média dos automóveis de determinado ano;
 - A marca e modelo dos carros de uma determinada cor.

Bibliografia

- LAUREANO, Marcos. **Programação em C para ambiente Linux**. Disponível em: <<http://br-c.org/doku.php>>. Acesso em: 06 fev. 2011.
- MURTA, Cristina Duarte. *Slides da disciplina de Programação de Computadores I*. CEFET-MG, 2010.
- SENNE, Edson Luiz França. **Primeiro Curso de Programação em C**. 2. ed. Florianópolis: Visual Books, 2006.
- SOARES, Gabriela Eleutério. *Slides da disciplina de Programação de Computadores I*. CEFET-MG, 2011.