

Programação / Programação I

LEI/1, LTSI/1, LMAT/1

Hugo Pedro Proença

Universidade da Beira Interior
Departamento de Informática

Resumo

- Tipos de Dados Compostos: “typedef”
 - Sintaxe
 - Acesso
 - Exemplos
- Exercícios

Tipos de Dados

- Até ao momento, os programas implementados utilizavam exclusivamente os tipos de dados básicos:
 - int
 - char
 - float
 - double
 - void
- Como fazer para guardar informação sobre uma pessoa, por exemplo?

typedef

- A palavra chave “typedef” serve para definir novos tipos de dados:
- Sintaxe:
 - `typedef <descrição> <identificador>;`

typedef: Exemplo 1

- Pode ser utilizada para renomear tipos de dados existentes:
 - `typedef float num_real`
- A partir deste momento, pode-se utilizar “`num_real`” para definir uma variável que guarda um valor do tipo “`float`”.
 - Declarar variáveis, parâmetros, ...
 - `num_real x;`
- Aumentar a legibilidade de programas.

typedef: Exemplo 2, tipos compostos

- O mais comum é utilizar “typedef” para criar tipos de dados compostos:

- typedef struct{

- int bi;

- float altura;

- char sexo;

- } Pessoa;

- O novo tipo de dados (Pessoa) é composto por 3 atributos (1 inteiro, 1 real e 1 carácter).

typedef: Exemplo 2, tipos compostos

- Pode-se utilizar o novo tipo exatamente da mesma forma e em todas as circunstâncias, como se de um tipo básico se tratasse:
 - Declarar variáveis:
 - Pessoa p1;
 - Passar parâmetros:
 - void mostra(Pessoa p);
 - Retornar valores:
 - Pessoa recebeDados();

typedef: Exemplo 2, tipos compostos

- Instruções de Atribuição:

```
Pessoa p1,p2;
```

```
...
```

```
p1=p2;           //Copia todos os atributos de p2 para p1.
```

- Passagem de parâmetros:

```
mostraPessoa(p1);           //é criada uma cópia  
de todos os atributos de p1 para o parâmetro
```

typedef: Exemplo 2, tipos compostos

- Encadeamento de tipos compostos:

```
typedef struct{  
    int hora, minuto;  
}Hora;
```

```
typedef struct{  
    int codigoEmpregado;  
    Hora registo;  
}Assiduidade;
```

- Declaração de conjuntos (vectores, matrizes,...):

```
Assiduidade registo[1000]; //”registo” é um vector com  
espaço para 1000 variáveis do tipo “Assiduidade”
```

Tipos compostos: acesso

- Para aceder (leitura / escrita) a cada atributo de uma variável de tipo composto, deve-se utilizar “.”

- Exemplo simples:

```
Pessoa p1;  
scanf(“%d”, &p1.bi);  
...  
printf(“Altura=%f\n”, p1.altura);
```

- Exemplo vectores:

```
Pessoa turma[20];  
...  
turma[0].bi=123456;
```

Tipos compostos: definição

- Usualmente, a definição de tipos faz-se imediatamente após a inclusão de bibliotecas standard.
 - Qualquer função / biblioteca pessoal poderá utilizar o novo tipo de dados:

```
#include <stdio.h>
#include <math.h>
typedef struct{
    int hora;
    int minuto;
} Hora;
...
void mostraHora(Hora x);
```

Tipos compostos: Exercício

- Defina um tipo de dados (Data) adequado para registrar uma data (composta por um dia, mês e ano).
- De seguida, implemente funções para:
 - Mostrar uma data no écran (formato: dd/mm/aaaa).
 - **Protótipo:** void mostraData(Data d);
 - Verificar se uma determinada data é ou não válida.
 - **Protótipo:** int dataValida(Data d);
 - Receber os valores de uma data válida.
 - **Protótipo:** Data recebeData();
 - Comparar duas datas.
 - **Protótipo:** int comparaDatas (Data d1, Data d2);
 - Somar dias a uma data.
 - **Protótipo:** Data somaDias(Data d, int dias);
 - Contar os dias entre duas datas válidas.
 - **Protótipo:** int contaDias(Data d1, Data d2);