

Curso Profissional: Programador/a de Informática
PSD – 10.º ano: UFCD 0810 - Programação em C/C++ - avançada
Ficha de Trabalho 2

Ano letivo 21/22

ESTRUTURAS

Definição de tipos – typedef: Podemos representar estruturas unicamente através de uma palavra (um sinónimo), tal como fazemos com os tipos base da linguagem. Isso é feito utilizando a palavra reservada **typedef** cuja sintaxe é:

typedef tipo_existente sinónimo

Exemplos:

1) **typedef struct Data**

```
{  
    int Dia, Ano;  
    char mes[12] ;  
} dt ;
```

agora podemos declarar as variáveis de dois modos diferentes:

struct Data dt1, dt2; OU **dt** dt1, dt2;

2) **typedef int inteiro;** /* o tipo int passa a ser representado pela palavra inteiro */
int a, b[10] ; **é equivalente a** inteiro a, b[10] ;

3) **typedef *char pointer;**
...
pointer string = "Olá"; /* STRING é do tipo (*char) */

Notas:

- A palavra **typedef** não cria um novo tipo. Permite apenas que um determinado tipo possa ser denominado de forma diferente, de acordo com as necessidades do programador;
- Na definição de um **typedef** não podem ser declaradas variáveis.

```
typedef struct Data  
{  
    int Dia, Ano;  
    char mes[12] ;  
} dt , dt1, dt2;       / * ERRO !!! */
```

A palavra reservada **typedef** nada mais é do que um atalho em C para nos referir a um determinado tipo existente com nomes sinónimos.

Onde definir Estruturas e typedef: No início de um programa ou num header file que se junte ao mesmo, de forma a serem visíveis por todo o programa (programa principal e subprogramas). No entanto podemos definir typedef de tipo de dados existentes dentro da função main.

```
# include < ... > ...
struct data { ... } ;
typedef ..... ;
/* protótipos das funções */
...
/*funções */
f () {...};
g() {...}
main() {...}
    printf ("\n Ano: %d   Turma: %c ", al[i].ano, al[i].turma);
    printf ("\n Data de Nascimento: %d - %d - %d", &al.dt.dia, &al.dt.mes, &al.dt.ano);
    return;
}
```

Exemplo 1 (uso de typedef para referir outros tipos de dados):

```
#include <stdio.h>
main( )
{
    typedef int meuInteiro;
    typedef char String[20];

    meuInteiro numero = 1;
    String nome;
    scanf(" %s", nome);

    printf("A variavel do tipo 'meuInteiro' eh um int e vale %d\n", numero);
    printf("Ja a variavel 'nome' eh uma String e armazena \"%s\"\n", nome);
}
```

EXERCÍCIOS

1-Define um tipo de dados denominado **PESSOA** que permita identificar a estrutura **Pessoa**, sendo os campos, da estrutura, o nome, a idade, o género, o estado civil e o salário.

2- Declara, de dois modos distintos, uma variável de nome individuo que se refira à estrutura declarada no exercício 1.

3- Tendo em conta a seguinte estrutura:

```
typedef struct Data
{
    int Dia, Ano;
    char mes[3+1] ;
} dt ;
```

- Declara, de 2 modos diferentes, duas variáveis do tipo struct Data: uma simples de nome **ds** e um vetor com 2 elementos com o nome **dv**;
- Escreve o código para inicializar **ds** com os dados 12,2012,"DEZ" e **dv** como os dados 10,2010,"JAN" e 03,2016,"FEV";
- Escreve o código para adicionar um dia à data de **ds**;
- Escreve o código para colocar o ano de 2000 no segundo registo de **dv**;
- Escreve o código para alterar o mês do 1º registo de **dv** para "MAR";
- Escreve o código para mostrar no ecrã a 1.ª letra de cada um dos meses dos registos em **dv**.

ESTRUTURAS DENTRO DE ESTRUTURAS

Como já foi referido, uma estrutura pode conter, na sua definição, variáveis simples, vetores, apontadores (a estudar no módulo 6) ou outras estruturas.

Para usarmos estruturas ou typedef dentro de uma estrutura, devemos ter o cuidado de as definir previamente e só depois definir a nova estrutura onde são utilizados.

Exemplo: Declarar um tipo denominado **pessoa** que contenha o nome, idade e data de nascimento referente a uma estrutura de nome individuo.

```
typedef struct Data
```

```
{
```

```
    int dia, ano;
```

```
    char mes[4] ;
```

```
} dt;
```

```
typedef struct individuo
```

```
{
```

```
    char nome[50];
```

```
    int idade;
```

```
    dt data_nasc ;
```

```
} pessoa;
```

/* declaração de uma variável simples, desse tipo, e de um vetor com 3 pessoas*/

```
pessoa p, pessoas[3] ;
```

Para colocar o ano de 1999 no segundo elemento do vetor faz-se:

```
    pessoas[1] . dt_nasc . ano = 1999 ;
```

Para alterar o mês do último elemento do vetor para “NOV” faz-se:

```
strcpy (pessoas[2] . data_nasc . mes , “NOV”);
```

Para mostrar no ecrã a 1ª letra de cada um dos meses de todos os elementos do vetor pessoas faz-se:

```
for (i=0;i<3;i++)
    putchar(pessoas[i].dt_nasc . mes[0]);
```

Exemplo de processamento de uma estrutura dentro de uma estrutura:

```
#include <stdio.h>
struct data                // definição da estrutura data
{
    int dia;                // ou int dia,mes,ano;
    int mes;
    int ano;
};
struct alunos              // definição da estrutura alunos
{
    char nome[80];
    int idade;
    int ano;
    char turma;
    struct data dt;
} al[18];                  //declaração de uma variável do tipo struct alunos
void exibesaída(int i);    // protótipo da função exibesaída
void leentrada(int i);    // protótipo da função leentrada

main()                    //função principal
{
    int i,n;
    printf("Sistema de registo de alunos\n");
    printf (" \nQuantos alunos? ");
    scanf("%d", &n);
    for (i=0; i<n; ++i)
        leentrada(i);
    for (i=0; i<n; ++i)
        exibesaída(i);
    getchar();
}
```

```

void leentrada(int i)                //função leentrada
{
    printf (" \n Aluno nº %d\n", i+1);
    printf (" Nome: "); scanf("%s", al[i].nome);
    printf ("\n Idade: "); scanf("%d", &al[i].idade);
    printf ("\n Ano: "); scanf("%d", &al[i].ano);
    printf ("\nTurma: "); scanf(" %c", &al[i].turma);
    printf ("\n Data de Nascimento: \n");
    scanf("%d%d%d",&al[i].dt.dia, &al[i].dt.mes, &al[i].dt.ano);
}

void exibsaida(int i)                // função exibsaida
{
    printf ("\n Aluno nº %d\n", i+1);
    printf (" Nome: %s", al[i].nome);
    printf ("\nIdade: %d\n\n", al[i].idade);
}

```

EXERCÍCIO:

4- Escreve um programa que permita inserir dados para 3 indivíduos e escrever no ecrã os valores inseridos, para a estrutura **indivíduo** declarada no exemplo da página 3.