Ficha de Trabalho N.º 6

Versão 2024/25

Objetivos: Utilização de Ponteiros e Alocação Dinâmica de Memória.

Conceitos Necessários à Resolução da Ficha

Alocação de memória

A alocação dinâmica de memória pode realizar-se usando duas funções relativamente semelhantes: a função malloc e a função calloc.

Função malloc (*Memory allocation*)

SINTAXE
void * malloc(size_t n_Bytes)

size_t está normalmente definido no #include <stdlib.h> como sendo
typedef unsigned int size_t;

A função malloc permite alocar o conjunto de Bytes indicados pelo programador, devolvendo um apontador para o bloco de Bytes criados, ou NULL caso a alocação falhe.

Repare que a função malloc devolve um tipo "curioso": void . Como permite criar qualquer tipo de dados, o resultado terá que ser colocado num apontador. Devolve um endereço de memória.*

Alocação de memória (cont.)

NOTAS RELATIVAS À FUNÇÃO REALLOC

- *Se o bloco actualmente alocado puder ser aumentado para suportar a o novo, a memória adicional é reservada também e é retornado ptr.*
- *Se não existir espaço suficiente para prolongar o bloco, é criado um novo bloco com a totalidade dos Bytes necessários. Os dados são copiados para a nova localização e é retornado o novo endereço.*
- *Se o parâmetro ptr for igual a NULL, a função comporta-se como malloc.*
- *Se por algum motivo não for possível a alocação de memória ou se o número de Bytes for igual a zero, é devolvido NULL.*

Libertação de memória

A libertação de memória alocada através das funções malloc e a função calloc pode realizar-se através da função free.

SINTAXE
void free (void *ptr)

Alocação de memória (cont.)

Função calloc

SINTAXE
void * calloc(size_t num, size_t size)

Esta função permite criar, dinamicamente, *num* elementos, cada um com *size Bytes*.

As funções malloc e calloc alocam memória no *heap* de acordo com o tamanho passado e retornam um ponteiro para o local onde houve a alocação. A diferença é que a função calloc coloca todos os bytes alocados com o valor 0 (zero), algo que a função malloc não faz.
Ambas devolvem o endereço da zona criada ou NULL.

Função realloc

SINTAXE
void * realloc(void *ptr, size_t new_size)

A função realloc permite alterar o número de Bytes que estão presentemente associados a um bloco previamente criado, utilizando as funções *malloc* ou *calloc*.

Exercício
Implementar a função *strdup* que cria uma nova *string* exactamente igual à que lhe foi passada por parâmetro.

```
char *strdup(char *s)
// cria uma nova string idêntica à que lhe foi passada como parâmetro
{
     char *tmp = (char *) malloc(strlen(s)+1);
     //+1 para o caracter terminador ('\0')
     //tmp ficou com o endereço do bloco de Bytes criado
     if (tmp!=NULL)
       strcpy(tmp, s);
     return tmp;
}
```

- Declara-se um apontador que recebe a nova zona de memória criada recorrendo à função malloc.
- Se a criação de memória for bem sucedida, copia-se o conteúdo de s para tmp e devolve-se o endereço em que a nova *string* foi criada.

Nota: embora esta função não faça parte da norma ANSI, a generalidade dos compiladores incorporou-a na sua biblioteca de funções (string.h)

Problemas Propostos

1 - Elabore um programa que realize as operações a seguir indicadas, com matrizes de dimensão variável e que onde seja permitido registar dados do tipo inteiro ou real, ambos especificados pelo utilizador:

a) Alocação dinâmica de memória para armazenar uma matriz com um número de linhas e colunas indicado pelo utilizador;

b) Leitura, por linhas, das componentes da matriz;

c) Escrita, por linhas, das componentes da matriz;

d) Cálculo do TRAÇO (soma das componentes da diagonal principal);

e) Verificação da existência de simetria relativamente à diagonal principal.

2 - Acrescente ao programa anterior uma função que permita efetuar o produto de duas matrizes.

3 - Elabore um programa que determine os resultados de um acto eleitoral em que concorrem nListas, a nLugares. A atribuição dos lugares a cada lista candidata deve ser feita usando o MÉTODO DE HONDT.
Infra, disponibiliza-se alguma informação relativa ao método de Hondt.

MÉTODO de HONDT

O método de HONDT aplica-se mediante a divisão sucessiva do número total de votos obtidos por cada candidatura pelos divisores (1, 2, 3, 4, 5 etc.) e pela atribuição dos mandatos em disputa por ordem decrescente aos quocientes mais altos que resultarem das divisões operadas. O processo de divisão prossegue até se esgotarem todos os mandatos e todas as possibilidades de aparecerem quocientes iguais aos quais ainda caiba um mandato.

Em Portugal encontra-se legalmente prevista uma correção ao método Hondt puro, na medida em que, caso falte atribuir o último mandato e se verifique igualdade do quociente em duas listas diferentes, tal mandato será atribuído à lista que em termos de resultados totais tenha obtido menor número de votos.

Exemplo prático (conversão dos votos em mandatos):

O círculo eleitoral "X" tem direito a eleger 7 deputados e concorrem 4 partidos: A, B, C e D. Apurados os votos, a distribuição foi a seguinte: A - 12.000 votos; B - 7.500 votos; C - 4.500 votos; e D - 3.000 votos. Da aplicação do método de Hondt resulta a seguinte série de quocientes:

Divisor	Partido			
	A	B	C	D
1	**12000**	**7500**	**4500**	**3000**
2	**6000**	**3750**	2250	1500
3	**4000**	2500	1500	1000
4	3000	1875	1125	750

No exemplo constante da tabela, os quocientes correspondentes a mandatos, assinalados a cinzento, levam à seguinte distribuição:

Partido A - 3 deputados, correspondentes aos quocientes 12000 (1.º eleito), 6000 (3.º eleito) e 4000 (5.º eleito). Note-se que apesar do quociente resultante da divisão por 4 ser 3000, igual aos votos obtidos pelo partido D, o mandato é atribuído ao menos votado, isto é, ao Partido D, que assim elege o seu deputado.
Partido B - 2 deputados, correspondentes aos quocientes 7500 (2.º eleito) e 3750 (6.º eleito).
Partido C - 1 deputado, correspondente ao quociente 4500 (4.º eleito).
Partido D - 1 deputado, correspondente ao quociente 3000 (7.º e último eleito), beneficiando da regra que em igualdade atribui o lugar à lista menos votada, arrebatando o lugar ao partido A.

Na prática, a ideia é muito simples:

havendo vários mandatos para atribuir, se há n listas concorrentes, cada uma com v_n votos, e a cada uma já foram atribuídos a_n mandatos (no início a_n é obviamente 0) então o próximo mandato é atribuído à lista com o máximo valor de

$$v_n / (1+a_n)$$

O texto acima é um excerto que foi retirado do site da Comissão Nacional de Eleições.
O documento completo está disponível em https://www.cne.pt/content/metodo-de-hondt .