

```

/*****
Programa: e0705.cpp
Proposito: Mostra a modificacao de elementos de um vetor passado
como
argumento de uma

```

funcao.

Última Revisao: 20/10/97

```

*****/

```

```

#include <dos.h>
#include <conio.h>
#include <stdlib.h>
#include <stdio.h>
#define MAX 150

```

```

// *****
// 1 Programa principal
// *****

```

```

void main(){
    int i;                // contador de iteracao
    int n;                // numero de elementos do vetor
    int num[MAX];        // declaracao do vetor 'num' de MAX elementos
    void ordem(int[],int); // prototipo da funcao de ordenamento
    randomize();         // inicializa gerador aleatorio

```

```

// *****
// 1.1 Entrada de dados
// *****
clrscr();
do{

```

```

    puts("\nDigite a quantidade de numeros a ordenar: ");
    scanf("%d",&n);
}while(n < 1 || n > MAX);

```

```

// *****
// 1.2 Geracao de numeros aleatorios
// *****
puts("Gerando lista de numeros aleatorios...");
for(i = 0; i <= n-1; i++){

```

```

    num[i] = random(100); // gera numero aleatorio
    printf("%2d ",num[i]); // imprime vetor gerado
    delay(50);
}

```

```

// *****
// 1.3 Ordenacao e impressao
// *****

```

```

ordem(num,n); // passagem do vetor idade para a funcao
puts("\nValores ordenados:");
for(i=0 ; i<=(n-1) ; i++){
    printf("%2d ",num[i]); // imprime vetor ordenado
    delay(50);
}
getch();
}

```

```

// *****
// 2 rotina ordem()
// *****

```

// Metodo: Ordenacao Bolha.

// Referencia: MANZANO,J.A.N.G.,OLIVEIRA,J.F., Algoritmos: Logica para desenvolvimento de programacao,Sao Paulo? Erica, 1996 [p.121]

// "O mais simples e mais ineficiente algoritmo desenvolvido pela humanidade"

```

// *****
void ordem(int valor[],int num){ // declaracao da funcao

```

```

int t;    // contador de trocas
int temp; // armazenador temporario para troca
int i;    // contador de iteracao

do{
    // ordena...
    t = 0; // nenhuma troca ainda
    for(i = 0; i <= num-2; i++){ // para todos os elementos do vetor...
        if(valor[i] > valor[i+1]){ // se fora de ordem...
            temp = valor[i]; // troca...
            valor[i] = valor[i+1];
            valor[i+1] = temp;
            t++; //conta troca
        }
    }
}while(t); // ...enquanto houver trocas
}

```