Ordenação por Seleção

Zé Augusto

São Paulo, 13 de fevereiro de 2003

Sumário

1. Um exemplo bem simples de CWEB usando LaTeX

Aqui está implementado o algoritmo de ordenação por seleção que recebe como entrada um inteiro positivo n seguido por n inteiros. Como saída, o programa imprime os inteiros lidos em ordem crescente.

2. Estrutura geral do programa

Vamos descrever a estrutura geral do programa.

```
⟨ Arquivos a serem incluídos 3 ⟩
⟨ Programa Principal 4 ⟩
```

3. Precisamos incluir as definições padrões de I/O.

```
⟨ Arquivos a serem incluídos 3 ⟩ ≡
#include <stdio.h>
```

Este código é usado no bloco 2.

4. Definição do programa principal

```
⟨ Programa Principal 4 ⟩ ≡
  int main()
  { ⟨ Variáveis locais de main 6 ⟩
    ⟨ leitura dos dados de entrada 5 ⟩
    ⟨ Ordenação propriamente dita 7 ⟩
    ⟨ Impressão do vetor ordenado 10 ⟩return 0;
  }
```

Este código é usado no bloco 2.

5. Vamos fazer a leitura dos dados de entrada. Estes serão digitados pelo usuário.

```
⟨ leitura dos dados de entrada 5 ⟩ ≡
  printf("Entre␣com␣a␣quantidade␣de␣inteiros␣a␣serem␣ordenados:␣");
  scanf("%d", &n);
  printf("Entre␣com␣os␣%d␣inteiros␣a␣serem␣ordenados:␣", n);
  for (i = 0; i < n; i++) scanf("%d", &v[i]);
```

Este código é usado no bloco 4.

6. Acho melhor definir as variáveis locais que usamos, antes que eu me esqueça. Vamos definir também uma constante `MAX` que limitará o tamanho máximo do vetor a ser ordenado.

```
#define MAX 100
⟨ Variáveis locais de main 6 ⟩ ≡
  int n, i, v[MAX];
```

Veja também bloco 9.

Este código é usado no bloco 4.

7. A ordenação consiste em se encontrar o índice do maior elemento do vetor, dentre os elementos com índices de 0 até m. Inicialmente, $m = n - 1$. Esse elemento é colocado na posição m e, então, m é decrementado de 1.

```
⟨ Ordenação propriamente dita 7 ⟩ ≡
  for (m = n - 1; m > 0; m--) {
    ⟨ Encontre o índice maior do maior elemento em v[0..m] 8 ⟩
    aux = v[maior];
    v[maior] = v[m];
    v[m] = aux;
  }
```

Este código é usado no bloco 4.

8.

```
⟨ Encontre o índice maior do maior elemento em v[0..m] 8 ⟩ ≡
  maior = 0;
  for (i = 1; i ≤ m; i++)
    if (v[i] > v[maior]) maior = i;
```

Este código é usado no bloco 7.

9. Opa. Acabamos usando mais variáveis locais. Precisamos defini-las.

```
⟨ Variáveis locais de main 6 ⟩ +≡
  int maior, aux, m;
```

10. Por último, a impressão do vetor ordenado.

```
⟨ Impressão do vetor ordenado 10 ⟩ ≡
  printf("Impressão␣do␣vetor␣ordenado:␣");
  for (i = 0; i < n; i++) printf("%d␣", v[i]);
  printf("\n");
```

Este código é usado no bloco 4.

Índice Remissivo

Aqui vai uma lista de identificadores usados e onde eles aparecem. Entradas grifadas indicam os lugares de definições. Mensagens de erros também são mostradas.

Lista de Refinamentos