

Lógica de Programação

Vídeo Aula 1

Aula ministrada por Luiz Fernando Rodrigues Sanches, aborda o algoritmo para encontrar entre vários números o maior e a sua posição.

Nesta aula resolveremos o exercício número 1080 do URI Online Judge.

Problema

Leia 100 valores inteiros. Apresente então o maior valor lido e a posição dentre os 100 valores lidos.

O arquivo de entrada contém 100 números inteiros, positivos e distintos.

Algoritmo

Como não iremos realizar nenhum tipo de operação com os números que serão lidos, não será necessário armazená-los em um vetor.

O primeiro passo que iremos realizar é criar duas variáveis, uma para armazenar o maior valor e outra para armazenar sua posição:

```
int maior = sc.nextInt();
```

```
int posicao = 1;
```

Já criamos a variável e deixamos o primeiro valor lido como sendo o maior.

Algoritmo

Agora precisamos ler o restante dos números, como já sabemos quantos serão lidos, iremos utilizar o seguinte laço *for*:

```
for(int i = 2; i <= 100; i++) {  
    int number = sc.nextInt();  
}
```

Dessa forma armazenamos na variável *number* o valor lido, como iremos precisar desse valor apenas uma vez, não tem problema sobrepor seu valor a cada iteração.

Algoritmo

Precisamos agora implementar o código para comparar se acabamos de ler um valor maior que o armazenado em:

```
for(int i = 2; i <= 100; i++) {  
    int number = sc.nextInt();  
  
    if(number > maior){  
        maior = number;  
        posicao = i;  
    }  
}
```

Algoritmo

Assim que o maior valor até então for lido, a variável *maior* será substituída, e o valor de *i* será colocado na *posicao*, era exatamente isso que queríamos!

Após isso basta imprimirmos nossas variáveis na tela, como o exercício exige.

Conclusão

Trabalhar com laços *for* é essencial para garantirmos a automação do nosso algoritmo, porém temos que tomar cuidado com a lógica implementada.

Poderíamos ter solucionado este problema utilizando um vetor, porém considero essa forma mais prática.

Por esta aula é só,
Pratique bastante!