

Universidade Federal de Minas Gerais
Sistemas de Informação
2019/2

Daniel Pires Quirino

Trabalho Prático de Matemática Discreta

Belo Horizonte - MG

Sumário:

Introdução:	3
Descrição do formato de entrada dos dados:	3
Descrição do formato de saída dos dados:	3
Descrição geral do programa:	4
Funções utilizadas:	4
Bibliotecas utilizadas:	4
Análise Assintótica da solução realizada:	4
Testes realizados:	5

Introdução:

Esse trabalho prático consiste na análise de um vetor com n números inteiros, sendo n um número inteiro positivo, determinando a soma máxima encontrada em um sub-vetor contínuo desse vetor.

A partir dessa premissa, foi desenvolvido um software na linguagem C. Primeiramente, é obtido o número n (tamanho do vetor digitado pelo usuário), gerando um vetor de tamanho n , a partir de alocação dinâmica de memória. Esse vetor é percorrido, usando um algoritmo guloso, para que se possa determinar a maior soma e o maior índice. Posteriormente, esse algoritmo guloso é usado outra vez, com algumas modificações para que se possa encontrar o índice de menor valor que compõe a maior soma.

Descrição do formato de entrada dos dados:

O formato de entrada deve ser:

- Primeiramente o usuário deve digitar um número positivo e inteiro, que será o tamanho do vetor a ser analisado
- O segundo parâmetro deve ser os valores que o vetor assume da posição 0 até a posição $(n-1)$, sendo n o tamanho do vetor.
- Obs: Deve possuir uma quebra de linha entre os dois parâmetros:

```
10
2 9 6 4 -2 -9 3 9 0 3
```

Exemplo de parâmetro de entrada.

Descrição do formato de saída dos dados:

O formato de saída consiste no índice, que mostra o intervalo das posições em que se têm a maior soma possível, e soma, que determina a maior soma possível do vetor

```
Indice: 0 4
Soma: 10
```

Exemplo de saída do programa

Descrição geral do programa:

Funções utilizadas:

Com intenção de facilitar o entendimento do código e seguindo as boas práticas de programação, além da função principal `main()`, foram criadas algumas outras funções auxiliares que serão listadas abaixo:

- `max()`: retorna o maior valor entre dois números
- `getNumbersWithSpace()`: obtêm os índices do vetor digitado como parâmetro de entrada pelo usuário
- `getLowerIndex()`: Obtêm o menor índice da maior soma do vetor
- `maxSum()`: retorna o valor da maior soma do vetor
- Foram utilizados também algumas funções já prontas como o `'atoi()'`, `'malloc()'` e `'strtok()'`

Bibliotecas utilizadas:

Foram utilizadas as seguintes bibliotecas nesse software:

- `stdio.h`
- `stdlib.h`
- `string.h`
- `conio.h`

Análise Assintótica da solução realizada:

A complexidade desse algoritmo é dado por $O(n)$.

Isso se deve ao fato de que a maior interação que existe nesse algoritmo é um `for` que percorre as posições do vetor..

Testes realizados:

Geração de alguns testes realizados neste programa para ratificar a sua eficácia e correto funcionamento.

```
1 3 5 -8 9  
  
Indice: 0 4  
Soma: 10
```

Exemplo (1)

```
10  
1 4 -6 7 -3 8 9 -1 2 4  
  
Indice: 3 9  
Soma: 26
```

Exemplo(2)

```
6  
1 2 3 4 5 6  
  
Indice: 0 5  
Soma: 21
```

Exemplo(3)