

Projeto e Análise de Algoritmos - Programa 1.0

Complexidade do Algoritmo

Bruno Schubert Trindade (121151557)

Marcos Vinicius Treviso (121150107)

1 - Análise de Complexidade:

Será analisada a complexidade do método público *selecionaMusicas()*. Pois esse é o método que realiza os cálculos necessários para gerar a resposta para o programa, onde é ele que possui o laço principal.

<code>selecionaMusicas(musicas)</code>	Custo	#Execuções

<code>razao = sorted(razao, key = lambda x : x[3], reverse=True)</code>	c1	x
<code>somatam, somafav = 0, 0</code>	c2	1
<code>for i, tam, fav, more in razao:</code>	c3	n
<code>if(somatam+tam <= self.__tamMaximo):</code>	c4	n
<code>somatam += tam</code>	c5	n
<code>somafav += fav</code>	c6	n
<code>return somafav</code>	c7	1

O melhor e o pior caso dependem diretamente do tamanho do vetor (entrada do programa), pois a função *sorted* irá ordenar o vetor com algum algoritmo de ordenação de acordo com o número de entrada.

Sendo assim, ficaríamos com:

$$T(n) = (c3+c4+c5+c6).n + (c1).x$$

$$T(n) = n + x \rightarrow \text{sem analisar o } x \rightarrow n \rightarrow \mathbf{O(n)}$$

2 – Corretude:

Chamada:

`selecionaMusicas(musicas)`

Invariante de Laço:

No fim da iteração j a variável *somafav* terá o maior valor possível de acordo com o tamanho das músicas selecionadas que caibam em um disco com um tamanho determinado.

Inicialização: (iteração 0)

somatam e *somafav* podem receber um novo valor, caso a música selecionada satisfaça a condição.

Verdadeiro.

Manutenção: (iteração $j-1$)

somatam e *somafav* estão sendo somados de acordo com a condição, esperando a finalização para ter certeza que estão com os maiores valores possíveis, pois o último elemento pode também satisfazer a condição.

Verdadeiro.

Finalização: (iteração j)

somatam e *somafav* devem possuir os maiores valores, o que no caso depende da entrada do programa, ou seja, na etapa de finalização o resultado de *somafav* pode não ser o melhor segundo o invariante. Sendo assim, o programa não é correto, ele funciona para alguns casos, mas não para todos.

Falso.