

MO417 – Complexidade de Algoritmos  
Segundo Semestre de 2011  
Sexta Lista de Exercícios

## 1 Programação Dinâmica

1. Em programação dinâmica primeiro resolvemos os subproblemas e então decidimos qual deles usar para obter uma solução ótima do problema. Professora Capuleto afirma que não é sempre necessário resolver todos os subproblemas de modo a obter uma solução ótima. Ela sugere que uma solução ótima do problema de multiplicar uma cadeia de matrizes pode ser sempre encontrada escolhendo a matriz  $A_k$  para dividir o subproblema de multiplicar  $A_i A_{i+1} \cdots A_j$  escolhendo  $k$  que minimiza  $p[i-1]p[k]p[j]$  mesmo sem antes resolver todos os subproblemas. Mostre uma instância (exemplo) do problema de multiplicar uma cadeia de matrizes para a qual esta estratégia gulosa não resulta em uma solução ótima.
2. Determine uma SCM (Subseqüência Comum Máxima) para as seguintes seqüências:  
 $A = \langle 1, 0, 0, 1, 0, 1, 0 \rangle$  e  $B = \langle 0, 1, 0, 1, 1, 0, 1, 1, 0 \rangle$ .
3. Descreva um algoritmo de memorização para o problema SCM com complexidade de tempo  $O(mn)$ .
4. Descreva um algoritmo de complexidade  $O(n^2)$  de tempo e  $O(n)$  de espaço para encontrar uma subseqüência crescente mais longa de uma seqüência de  $n$  números. Por exemplo, uma subseqüência crescente mais longa de 10, 5, 15, 7, 9, 21, 12 é 5, 7, 9, 12.
5. Um ex-aluno da UNICAMP gostaria de desenvolver um novo e sofisticado vídeo-gravador digital chamado ViTo. No software ViTo, um programa de televisão  $i$  é representado por uma tripla  $(c_i, s_i, t_i)$  onde  $c_i$  é o **canal do programa**,  $s_i$  é o **horário de início do programa** e  $t_i$  é o **horário de término do programa**.

O proprietário do ViTo fornece uma lista de  $n$  programas de televisão que ele tem interesse em assistir e a cada programa  $i$ , ele atribui um **grau de interesse**  $r_i$ . ViTo pode gravar apenas um programa de cada vez e os horários de alguns programas podem ser **conflitantes**: dois programas  $i$  e  $j$  são conflitantes se  $s_i \leq s_j < t_i$  ou  $s_j \leq s_i < t_j$ . Note que  $i$  e  $j$  **não** são conflitantes se  $t_i = s_j$  (significando que  $j$  começa imediatamente depois de  $i$ ).

ViTo deve escolher um subconjunto dos programas a ser gravados de modo a **maximizar o grau de interesse agregado** dos programas escolhidos (ou seja, a soma dos graus de interesse dos programas escolhidos).

- (a) Mostre que o problema tem subestrutura ótima. Denote por  $T(n)$  o grau de interesse agregado de uma solução ótima para um conjunto de  $n$  programas. Escreva uma recorrência que define  $T(n)$ .
- (b) Escreva um algoritmo baseado em programação dinâmica que devolve um **conjunto de programas** com grau de interesse agregado máximo em tempo  $O(n^2)$ . Justifique a correção e a complexidade do seu algoritmo.

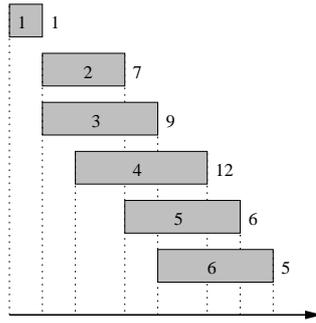


Figura 1: Exemplo de instância para o ViTo. Cada retângulo representa um programa (o canal foi omitido) e os números à direita deles representam o grau de interesse do usuário. Uma solução é gravar os programas 1, 2 e 5 com graus de interesse 1, 7, 6 totalizando 14. A melhor solução é gravar os programas 1, 3 e 6 com graus de interesse 1, 9, 5 totalizando 15.

## 2 Algoritmos Gulosos

1. Descreva um algoritmo guloso para o problema do exercício anterior.
2. Professor Midas pretende ir de carro de Newark até Reno usando a InterEstadual 80. O tanque de gasolina de carro do Prof. Midas, quando cheio, permite que o carro ande por  $n$  Km, e seu mapa fornece as distâncias entre postos de gasolina no seu trajeto. O professor gostaria de fazer o menor número possível de paradas ao longo da viagem. Descreva um método eficiente pelo qual o Prof. Midas pode determinar em quais postos de gasolina ele deve parar e prove que sua estratégia obtém uma solução ótima.
3. Descreva um algoritmo eficiente que dado um conjunto  $\{x_1, \dots, x_n\}$  de pontos na reta real, determina a menor coleção de intervalos fechados unitários (tamanho 1) que contém todos os dados pontos (por menor quero dizer, menor número de intervalos). Prove que seu algoritmo está correto. Justifique a complexidade do seu algoritmo.