Universidade Federal de Minas Gerais – UFMG
Instituto de Ciências Exatas
Departamento de Ciência da Computação
DCC206 – Algoritmos 1
Professor: Renato Vimieiro
Lista de exercícios 2 – Paradigmas e Teoria da Complexidade

---

1. (Goodrich e Tamassia R-10.4) Mostre a execução do algoritmo guloso de agendamento de tarefas para o seguinte conjunto $T = \{(1,2), (1,3), (1,4), (2,5), (3,7), (4,9), (5,6), (6,8), (7,9)\}$.

2. (Goodrich e Tamassia R-10.1) Considere a seguinte lista de itens com seus respectivos pesos e valores: $\{(12,4), (10,6), (8,5), (11,7), (14,3), (7,1), (9,6)\}$. Mostre a execução do algoritmo guloso para o problema da mochila fracionário com esses itens e uma capacidade de 18.

3. (Adaptado de Goodrich e Tamassia C-10.4) Um peregrino quer andar de uma cidade A até uma outra B. Para minimizar o peso que ele levará em sua mochila, ele pretende fazer sua jornada com apenas uma garrafa de água. No percurso a ser percorrido, contudo, existem vários pontos de abastecimento de água. Dado que ele deseja realizar o percurso no menor tempo possível, ele quer minimizar o número de paradas para encher sua garrafa. Ele sabe que consegue andar até k quilômetros com uma garrafa cheia. Como ele sente muita sede, ele não consegue andar nenhuma distância adicional após sua garrafa esvaziar. Sabendo que ele marcou em seu mapa todos os pontos de abastecimento no caminho de A para B, descreva um algoritmo guloso que encontre o menor número de paradas necessárias para ele realizar seu percurso. Prove a otimalidade do seu algoritmo.

4. Seja T uma árvore (um grafo conexo com n vértices e $n-1$ arestas). Desenhe um algoritmo guloso que encontre um conjunto independente máximo em T em tempo polinomial ou prove que isso não é possível. Caso desenhe um algoritmo, mostre que ele é ótimo.

5. (Goodrich e Tamassia R-11.1) Supondo que diferentes algoritmos de dividir-e-conquistar tenham gerado as recorrências abaixo, use o teorema mestre para caracterizar suas complexidades.

   (a) $T(n) = 2T(n/2) + logn$
   (b) $T(n) = 8T(n/2) + n^2$
   (c) $T(n) = 16T(n/2) + (nlogn)^4$
   (d) $T(n) = 7T(n/3) + n$
   (e) $T(n) = 9T(n/3) + n^3 logn$

6. (Kleinberg e Tardos 5.2) Considere o problema de encontrar o número de inversões. Nesse problema, dada uma sequência de n números $a_1, \ldots, a_n$, os quais assumimos ser todos distintos, definimos uma inversão como um par $i < j$ tal que $a_i > a_j$. Motivamos o problema de contagem de inversões como uma boa medida de quão diferentes são duas ordenações. No entanto, pode-se notar que essa medida é muito sensível (algumas inversões são geradas por pequenas flutuações nos valores). Vamos chamar uma inversão de significativa se $i < j$ e $a_i > 2a_j$. Apresente um algoritmo com complexidade $O(n \log n)$ para contar o número de inversões significativas entre duas ordenações.

7. (Adaptado de Goodrich e Tamassia A-11.6) Suponha que você tenha uma descrição geométrica dos edifícios de Manhattan e gostaria de construir uma representação do horizonte (skyline) de Nova Iorque. Ou seja, suponha que lhe seja dada uma descrição de um conjunto de retângulos, todos têm um de seus lados no eixo x, e você gostaria de construir uma representação da união de todos esses retângulos. Formalmente, uma vez que cada retângulo tem um lado no eixo x, você pode assumir que lhe é dado um conjunto, $S = \{(l_1, r_1, h_1), (l_2, r_2, h_2), \ldots, (l_n, r_n, h_n)\}$, sendo $l_i < r_i$, para $i = 1, 2, \ldots, n$, as coordenadas no eixo x das laterais do prédio e h_i sua altura. O horizonte de S é definido como uma lista de pares $[(x_0, h_0), (x_1, h_1), (x_2, h_2), \ldots, (x_m, h_m), (x_{m+1}, 0)]$, ordenada pelos valores de x_i, tal que, cada subintervalo, $[x_i, x_{i+1}]$, é o subintervalo maximal que possui um único intervalo mais alto, que está na altura h_i, em S, contendo $[x_i, x_{i+1}]$, para $i = 0, 1, \ldots, m$. As figuras abaixo ilustram essa situação:

Responda:

(a) Descreva um algoritmo *merge* que receba dois skylines A e B, de tamanhos $|A| = n_a$ e $|B| = n_b$, e retorne o skyline que reuna as duas representações em tempo $O(n_a + n_b)$.

(b) Apresente um algoritmo que, dado um conjunto de representações de edifícios, encontre seu skyline em tempo $O(n \log n)$. Demonstre a complexidade do algoritmo.

8. (Levitin 8.2-1) Mostre a execução do algoritmo de programação dinâmica para encontrar a solução ótima para o problema da mochila binário na instância a seguir. A capacidade de mochila é $W = 6$. O conjunto de itens é $\{(1, 3, 25), (2, 2, 20), (3, 1, 15), (4, 4, 40), (5, 5, 50)\}$. Cada item no conjunto anterior é representado por um ID, peso e valor, respectivamente.

9. (Goodrich e Tamassia A-12.8) Sistemas de reconhecimento de fala precisam correlacionar fluxos de áudio que representam as mesmas palavras ditas em velocidades diferentes. Suponha, portanto, que lhe sejam dadas duas sequências de números, $X = (x_1, x_2, \ldots, x_n)$ e $Y = (y_1, y_2, \ldots, y_m)$, representando dois fluxos de áudio distintos que precisam ser correlacionados.

   Um *mapeamento* entre X e Y é uma lista, M, de pares distintos, (i, j), ordenada lexicograficamente, tal que, para cada $i \in [1, n]$, há pelo menos um par (i, j) em M, e para cada $j \in [1, m]$, há pelo menos um par (i, j) em M.

   Um mapeamento é *monotônico* se, para quaisquer (i, j) e (k, l) em M, com (i, j) vindo antes de (k, l) em M, tivermos $i \leq k$ e $j \leq l$. Por exemplo, dadas $X = (3, 9, 9, 5)$ e $Y = (3, 3, 9, 5, 5)$ um possível mapeamento monotônico entre X e Y seria $M = [(1, 1), (1, 2), (2, 3), (3, 3), (4, 4), (4, 5)]$.

   O problema de *dynamic time warping* (DTW) é encontrar um mapeamento monotônico, M, entre X e Y, que minimize a distância, $D(X, Y)$, definida como: $D(X, Y) = \sum_{(i,j) \in M} |x_i - y_j|$. Essa minimização é realizada sobre todos os possíveis mapeamentos monotônicos entre X e Y. No exemplo anterior, temos $D(X, Y) = 0$.

   Descreva um algoritmo eficiente para resolver o problema de *dynamic time warping*. Qual é o tempo de execução do seu algoritmo?

10. (Adaptado de Goodrich e Tamassia A-12.11) Suponha que n computadores em uma rede local cabeada estejam dispostos em uma estrutura de árvore, T. Considere que T esteja enraizada em um desses computadores (digamos, aquele que está conectado à Internet). Ou seja, os computadores nesta rede formam os nós de T, e as arestas de T são determinadas pelos pares de computadores que possuem cabos conectando-os uns aos outros. O administrador desta rede está preocupado com a confiabilidade e deseja comprar alguns dispositivos de monitoramento de rede para verificar continuamente a qualidade das conexões em T. Se tal dispositivo de monitoramento for instalado em um computador, x, então x é capaz de monitorar continuamente todas as conexões diretas que ele tem com outros computadores. Para economizar dinheiro, ele está pedindo que você determine o menor número de dispositivos de monitoramento necessários para que haja pelo menos um dispositivo monitorando cada conexão em T. Por exemplo, se todo computador em T, exceto a raiz, fosse filho da raiz, o administrador precisaria de apenas um dispositivo de monitoramento (colocado na raiz). Responda:

    (a) Esse problema parece ser um caso especial de um problema NP-completo. Qual problema é esse?

    (b) Descreva um algoritmo de programação dinâmica em tempo $O(n)$ para mostrar que, para esses casos, existe solução polinomial.

11. (Goodrich e Tamassia C-18.10) Um hipergrafo $H = (V, \mathcal{E})$ é a generalização de um grafo, em que as hiperarestas são subconjuntos de vértices, $\mathcal{E} = \{S_1, S_2, \ldots S_{|\mathcal{E}|}\}$ e $S_i \subseteq V$. O problema de otimização de Hitting-Set consiste em encontrar o menor conjunto $T \subseteq V$ tal que $T \cap S_i \neq \emptyset$ para todo i. Prove que esse problema é NP-Completo.

12. (Goodrich e Tamassia C-17.7) Considere a versão 2-SAT do problema CNF-SAT, na qual cada cláusula na fórmula S fornecida tem exatamente dois literais. Note que qualquer cláusula da forma $(a \vee b)$ pode ser vista como duas implicações: $(\neg a \rightarrow b)$ e $(\neg b \rightarrow a)$. Considere um grafo G derivado de S, tal que cada vértice em G está associado a uma variável, x, em S, ou à sua negação, $\neg x$. Será adicionada uma aresta direcionada em G de a para b para cada cláusula equivalente à implicação $(\neg a \rightarrow b)$. Mostre que S é insatisfatível se e somente se houver uma variável x tal que exista um caminho em G de x para $\neg x$ e um caminho de $\neg x$ para x.

    (a) Descreva um algoritmo de tempo polinomial para resolver este caso especial do problema CNF-SAT.

    (b) Qual é o tempo de execução do seu algoritmo?