Rio de Janeiro, 29 de Novembro de 2007.
PROVA 2 DE ANÁLISE DE ALGORITMOS
PROFESSOR: EDUARDO SANY LABER
DURAÇÃO: 3 HORAS

Questão 1 (3.0pt) Considere um conjunto de n bandeiras $B = \{b_1, \ldots, b_n\}$, onde a bandeira b_i esta situada no ponto p_i da reta dos números reais. Além disso, assuma que a bandeira b_i fique visível somente durante o minuto i (instante i-1 a i) e que uma pessoa consiga se deslocar na reta a uma velocidade de 1 unidade / minuto. Dizemos que um subconjunto $S \subseteq B$ é viável se uma pessoa consegue enxergar todas as bandeiras de S partindo do ponto 0 no instante 0.

Por exemplo, se as bandeiras b_1, b_2, b_3, b_4 estão situadas nos pontos $-3, 2, 1, 3$, respectivamente, então $\{b_2, b_4\}$ é viável já que uma pessoa partindo do ponto 0, no instante 0, consegue enxergar as bandeiras b_2 e b_4 da seguinte forma: primeiro, ela se desloca do ponto 0 para o ponto 2 em dois minutos para enxergar a bandeira b_2 e depois se desloca do ponto 2 para o ponto 3 em um minuto para enxergar a bandeira b_4. Note também que o subconjunto $\{b_1\}$ não é viável já que uma pessoa partindo do ponto 0, no instante 0, não consegue enxergar a bandeira b_1 uma vez que ela levaria 3 minutos para chegar no ponto -3 e a bandeira b_1 só esta vísivel durante o primeiro minuto.

a) Seja $OPT(i)$ o tamanho do maior subconjunto viável de $\{b_1, \ldots, b_i\}$ que **inclui** b_i. Encontre uma equação de recorrência relacionando $OPT(i)$ com $OPT(1), \ldots, OPT(i-1)$. Note que nem todos $OPT(j)$, com $j < i$, precisam aparecer na equação.

b) Descreva o PSEUDO-CÓDIGO de um algoritmo polinomial para calcular $OPT(n)$. Análise sua complexidade

c) Assuma agora que você tem acesso a uma tabela M, já preenchida, aonde $M[i] = OPT(i)$ para $i = 1, \ldots, n$. Descreva o PSEUDO-CÓDIGO de um algoritmo polinomial para calcular o tamanho do maior subconjunto de B viável.

Questão 2 (2.0pt) Considere um vetor ordenado $A[1..n]$ com n números **inteiros distintos**. Explique com palavras como seria um algoritmo com complexidade $O(\log n)$ para determinar se existe um índice i tal que $A[i] = i$. Justifique porque o algoritmo funciona.

Questão 3 (2.0pt) Considere o número de Rebal definido da seguinte forma.

Se $i > 1$ ou $j > 1$ então

$$Reb(i,j) = 3 \times Reb(i-2,j) + \max_{k=1,\ldots,j-1}\{k + Reb(i,k)\}.$$

Se $i \leq 1$ ou $j \leq 1$, então

$$Reb(i,j) = 1$$

.

a)(1.0pt) Exiba o PSEUDO-CÓDIGO de um algoritmo **polinomial e recursivo** para calcular $Reb(m,n)$.

b)(1.0pt) Analise a complexidade do algoritmo proposto no item (a) em função de m e n .

Questão 4 (2.0pt) Seja um grafo $G = (V,E)$ não direcionado com pesos não negativos nas arestas e seja e uma aresta deste grafo. Explique com **palavras** como seria um algoritmo polinomial para encontrar a árvore geradora com menor peso dentre as árvores geradoras para G que contém a aresta e. Analise a complexidade deste algoritmo.

Questão 5 (2.0pt) Considere um conjunto de n itens onde o i-ésimo tem peso $p_i > 0$ e valor $v_i > 0$. O objetivo deste problema é maximizar o valor dos itens que podem ser transportados por um caminhão com capacidade máxima de peso W. Assuma que os itens **são divisíveis**, ou seja, que é possível transportar apenas uma **fração** de um item (se necessário). Proponha um algoritmo polinomial para maximizar o valor transportado e analise sua complexidade.