Universidade Federal Fluminense – Instituto de Computação IC/UFF
Prof. Carlos Alberto Martinhon

Algoritmos Randômicos e Aproximativos (Lista I)

01 - Seja x um número real e $S=\{a_1,a_2,...,a_n\}$ um conjunto de n números reais.

(a) Faça um algoritmo que determine se a soma de 2 elementos de S é igual a x. Construa um algoritmo $O(n^2)$ justificando sua resposta.
(b) Resolver o mesmo problema através de um algorimto de complexidade $O(nlogn)$. (**Sugestão:** Ordene S e depois busque o elemento x-z (onde $z \in S$).
(c) Resolva o mesmo problema supondo agora que S já esteja ordenada. Construa um algoritmo de complexidade $O(n)$.
(d) Novamente, seja S ordenada. Considere ainda que os elementos de S sejam distintos entre si. Construa um algoritmo de complexidade $O(logn)$ que retorne um índice i, caso exista, tal que $T[i]=i$.
(e) Considere dois subconjuntos S_1 *e* S_2, e um número real x. Construa um algoritmo de ordem *O(nlogn)* (onde n é o tamanho total de ambos os conjuntos) que verifique se a soma de 2 elementos de S_1 e S_2 respectivamente é igual a x.

02 – Resolva as seguinte recorrências (**Sugestão**: considere inicialmente n potência de 2 e depois n qualquer):

(a) $\begin{cases} T(1) = a, \\ T(n) = 8T(n/2) + bn^2, \quad p/\ n > 1 \end{cases}$

(b) $\begin{cases} T(2) = 1, \\ T(n) = 2T\left(n^{1/2}\right) + \log n, \quad p/\ n > 1 \end{cases}$

03 – Mostre que os problemas de decisão abaixo pertencem a NP:

(a) Seja S um vetor com n valores inteiros. No problema da *partição de conjuntos* desejamos saber se S pode ser particionado em *dois* subconjuntos A e $A'=S-A$ tal que, a soma dos elementos em A é igual a soma dos elementos em A'.
(b) Considere um grafo $G=(V,E)$ qualquer (não-orientado) e k, um inteiro positivo. Um conjunto *V' Í V* define um conjunto independente se, e somente se, " *i,j V'* tivermos $(i,j) \notin$ E. No *Problema do Conjunto Independente* (versão decisão) desejamos saber se existe ou não um subconjunto V' de V de tamanho maior ou igual a k.
(c) Considere novamente um grafo $G=(V,E)$ qualquer (não-orientado) e k, um inteiro positivo. Um conjunto *V' Í V* define um conjunto independente dominante se, e somente se, " $i \in V \backslash V'$, existir $j \in V'$ tal que $(i,j) \in E$. No problema do *Menor Conjunto Independente Dominante* (*Minimum Independent Dominating Set*) na versão decisão desejamos saber se existe V' de tamanho menor ou igual a k.

04 – Considere uma lista não ordenada L com n valores inteiros. Suponha que um *Quicksort Modificado* seja obtido escolhendo-se o elemento da metade da lista como *pivô* (lembre-se que escolhíamos sempre o primeiro elemento – *Quicksort padrão*). Suponha agora que o *Quicksort padrão* seja executado normalmente após a troca deste elemento com o elemento na primeira posição de L. Construa uma lista de 9 elementos para este *Quicksort modificado* (pior entrada), de forma que sejam necessários $O(n^2)$ passos para sua ordenação completa.

05 – Suponha que, a cada escolha do elemento *chave* no algoritmo *Quicksort* tenhamos sempre 2 sub-listas com $9n/10$ e $n/10$ elementos respectivamente. Mostre que a complexidade do *Quicksort* neste caso será de O*(nlogn)* iterações.

NOTA: A lista pode ser discutida em conjunto mas cada aluno deverá entregar sua própria resolução (s/ copiar do colega!).