

Rio de Janeiro, 7 de Outubro de 2013.

PROVA 1 DE PROJETO E ANÁLISE DE ALGORITMOS

PROFESSOR: EDUARDO SANY LABER

DURAÇÃO: 2:50h

OBSERVAÇÕES

1. Algumas questões pedem para explicar com palavras. pode-se utilizar algoritmos vistos em aula e a explicação deve ser suficientemente boa para permitir que alguém implemente, sem muitas dúvidas, a solução proposta.

2. Cabe ao aluno deixar bastante clara a solução proposta e não ao professor gastar muita energia para entendê-la.

1. (2.0pt) Seja A um vetor indexado de 1 a n de números inteiros. Dizemos que o par de índices (i, j) é uma inversão em A se $i < j$ e $A[i] > A[j]$.

a) Exiba o pseudo-código de um algoritmo para determinar o número de inversões do vetor A e analise a sua complexidade de pior caso em função de n .

b) Assuma que os números armazenados no vetor A pertencem ao conjunto $\{1, \dots, k\}$. Podemos aproveitar esta informação para melhorar a complexidade do algoritmo proposto? Como?

2. (2.0pt) Seja $T(n)$ a complexidade de pior caso do pseudo-código abaixo em função de n .

$i \leftarrow 0$

Enquanto $i^2 \leq n$

Se i é múltiplo de c (*)

Fim Para j variando 1 a n

$cont \leftarrow 0$

Fim Para

Fim Se

$i++$

Fim Enquanto

a) Assuma que $c = 2$ na linha (*). Encontre uma função $f(n)$ tal que $T(n) = \Theta(f(n))$.

b) Assuma agora que $c = \sqrt{n}/2$ na linha (*). Encontre uma função $f(n)$ tal que $T(n) = \Theta(f(n))$.

3. (2.0pt) Um projeto consiste de um conjunto de atividades $A = \{a_1, \dots, a_n\}$. Além disso temos uma lista L de pares ordenados de atividades que definem restrições de precedência entre elas. Se o par (a_i, a_j) pertence a L então a_j tem que começar depois de a_i terminar. O gerente do projeto afirmou que é possível realizar todas as atividades começando com a atividade a_1 e terminando com a atividade a_n . Explique com palavras como seria um algoritmo para determinar se o gerente do projeto está correto ou não. Qual seria a complexidade do algoritmo?

4. (2.0pt) Modifique o pseudo-código da DFS abaixo para ela calcular a altura da árvore DFS que ela produz para um grafo conexo não direcionado $G = (V, E)$ ao ser chamada a partir de um nó s . Lembre que cada vértice de V é um nó da árvore DFS e u é pai de v na árvore se e somente se v é visitado a primeira vez a partir de u . Qual a complexidade do procedimento? A altura representa o comprimento do maior caminho no grafo que tem início em s ?

5. (2.5pt) Dado um grafo conexo não direcionado $G = (V, E)$, explique com palavras como seria um algoritmo para encontrar um par de vértices u e v que satisfaz simultaneamente as seguintes propriedades: (i) $uv \notin E$ e (ii) o grafo $G + uv$ tem um ciclo que contém a aresta uv . Analise a complexidade do algoritmo proposto em função de m e n . Quanto mais eficiente o algoritmo proposto maior a pontuação.

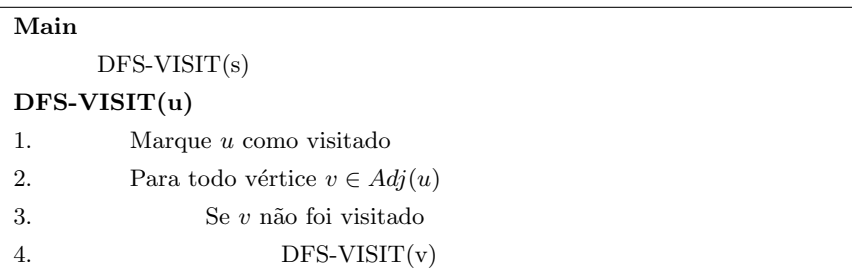


Figura 1: DFS