Universidade Federal de Pernambuco
Centro de Informática
Bach. em Eng^{a}. da Computação

IF672ec Algoritmos e Estr. de Dados 2015.2
Prof. Paulo Fonseca

Segunda Prova — 03 de Dezembro de 2015

- Esta prova tem 05 questões.
- A duração da prova é de 02h00min.

QUESTÃO 1 (2,5pt)

Considere o problema a seguir. Temos três baldes de 8, 5 e 3 litros de capacidade, respectivamente, estando o primeiro completamente cheio e os demais completamente vazios. Desejamos dividir os 8ℓ do primeiro balde, de forma exata, em duas metades de 4ℓ, contidas, respectivamente, no primeiro e segundo baldes. Para tal, é permitido apenas despejar o conteúdo dum balde num outro até que o balde doador fique vazio ou o balde receptor fique cheio (ou ambos simultaneamente).

a) Represente o espaço de busca do problema na forma de um grafo dirigido.

b) É possível encontrar a solução *ótima* do problema, i.e. aquela que corresponde ao menor número de despejamentos, através de um percurso nesse grafo? Se sim, indique o tipo do percurso, ilustre-o através da árvore correspondente sobre o grafo do item (a), e exiba a sequência ótima na forma

$$g_0 m_0 p_0 = 800 \to g_1 m_1 p_1 \to g_2 m_2 p_2 \to \cdots \to g_n m_n p_n = 440,$$

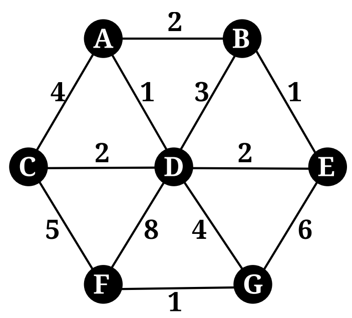
onde g_i, m_i e p_i denotam, respectivamente, os conteúdos (em litros) dos baldes *g*rande (de 8ℓ), *m*édio (de 5ℓ) e *p*equeno (de 3ℓ) após o i-ésimo despejamento. Se não é possível, explique sucintamente o porquê.

QUESTÃO 2 (2,5pt)

Complete o diagrama

Iter. #	Peso, Precursor A	B	C	D	E	F	G
0	$0, -$	$\infty, ?$	$\infty, ?$	$\infty, ?$	$\infty, ?$	$\infty, ?$	$\infty, ?$
1	$0, -$	$2, A$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

correspondente à execução do *Algoritmo de Prim* sobre o grafo a seguir.



QUESTÃO 3 (2,5pt)

Considere o *Algoritmo de Floyd-Warshall* para o cálculo das distâncias mínimas entre todos os pares de vértices de um grafo G dado.

a) Escreva a definição recursiva de $\delta^k(i,j)$ ="distância mínima de v_i a v_j em G, passando apenas por vértices intermédios em $\{v_1, \ldots, v_k\}$", utilizada no algoritmo, incluindo o caso base.

b) Escreva a matriz de PD ao final da execução do algoritmo sobre o grafo da Questão 2, sabendo que a matriz após a penúltima iteração está como na tabela a seguir.

	A	B	C	D	E	F	G
A	0	2	3	1	3	8	5
B	2	0	5	3	1	10	7
C	3	5	0	2	4	5	6
D	1	3	2	0	2	7	4
E	3	1	4	2	0	9	6
F	8	10	5	7	9	0	1
G	5	7	6	4	6	1	0

QUESTÃO 4 (2,5pt)

Responda os itens a seguir com base no pseudocódigo abaixo.

Algoritmo *whatdoido*
Entrada $W = (w_1, \ldots, w_n) \in \mathbb{R}_+^n$;
$V = (v_1, \ldots, v_n) \in \mathbb{R}_+^n$;
$K \in \mathbb{R}_+$
Saída $S \in \mathbb{R}_+$
devolva $func(W, V, K, 0, 0, 0)$

Função *func*
Entrada $W = (w_1, \ldots, w_n) \in \mathbb{R}_+^n$;
$V = (v_1, \ldots, v_n) \in \mathbb{R}_+^n$;
$K,\ SW,\ SV \in \mathbb{R}_+;\ i \in \mathbb{N}$
Saída $S \in \mathbb{R}_+$
se $SW > K$ **então**
  devolva 0
senão se $i = n$ **então**
  devolva SV
senão
  $l \leftarrow func(W,\ V,\ K,\ SW + W[i+1],$
    $SV + V[i+1],\ i+1)$
  $r \leftarrow func(W,\ V,\ K,\ SW,\ SV,\ i+1)$
  devolva $\max(l,\ r)$

a) Qual o problema resolvido pelo algoritmo?

b) Qual a complexidade assintótica do algoritmo no pior caso?

c) Ilustre a execução do algoritmo sobre a entrada $W = (4, 3, 1, 2)$, $V = (120, 80, 20, 70)$, $K = 5$.

d) Qual a técnica de desenho de algoritmos empregada?

e) Modifique ligeiramente o algoritmo de forma a transformá-lo num algoritmo do tipo *branch and bound*.

QUESTÃO 5 (1,0pt extra. Correção "*tudo ou nada*".)

Prove que o problema de decisão

VERTEX-COVER
Entrada: Um grafo $G = (V, E)$; $K \in \mathbb{N}$.
Propriedade: G possui uma cobertura $C \subseteq V$ de, no máximo, K vértices.

é NP-completo.

Obs.: Pode usar diretamente (sem precisar provar) a NP-completude de qualquer outro problema de decisão NP-completo comentado em aula, especificando-o adequadamente.

```
             \|/
            - * -
             /1\
              1
             101
           1000001
         10000000001
       100000000000001
          100000001
        1000000000001
     1000000000000000001
        1000000000001
     1000000000000000001
   10000000000000000000001
1000000000000000000000000000001
            10001
            10001

        BOAS FESTAS
          E FELIZ 2016
```