Universidade Federal de Pernambuco
Centro de Informática
Bach. em Ciência da Computação

IF672cc Algoritmos e Estr. de Dados 2014.1
Profs. Katia Guimarães, Paulo Fonseca

Prova Final — 08 de Agosto de 2014

- Esta prova tem 07 questões divididas em duas partes.
- Responda cada parte em uma folha separada, sem esquecer de assinar cada folha.
- A duração da prova é de 02h00min.

PARTE I

QUESTÃO 1 (Análise de algoritmos – 1,0pt)
Assinale V (verdadeiro) ou F(falso). Não é preciso justificar. Cada alternativa errada anula uma certa.

a)
b)
c)
d)
e)

QUESTÃO 2 (Tabelas de dispersão – 1,0pt)
Represente a tabela de dispersão de tamanho $m = 11$ resultante da inserção das chaves

$$24,\ 98,\ 33,\ 50,\ 80,\ 35,\ 109,$$

nesta ordem, utilizando a política de *hashing fechado* com resolução de colisões por *sondagem linear* (*linear probing*) e utilizando a função de dispersão $h(k) = k \mod m$.

QUESTÃO 3 (Árvores binárias – 1,0pt)
Enumere em (a) *pré-ordem* e (b) *pós-ordem* a árvore AVL resultante das inserções sucessivas das chaves

$$3, 2, 1, 5, 4, 7, 6, 8,$$

nesta ordem.

QUESTÃO 4 (Heaps binárias – 1,0pt)
Represente na *forma de array*, a heap (fila com prioridades) resultante da inserção sucessiva das chaves a seguir.

1986-03-21, 1988-08-30, 1990-05-05, 1986-03-28, 1990-04-17,
1990-12-31, 1987-11-27, 1985-09-11, 1990-06-24.

As chaves representam datas de nascimento no formato `YYYY-MM-DD` e a relação de prioridade é: os mais velhos têm prioridade sobre os mais novos.

QUESTÃO 5 (Union-find – 1,0pt)
A estrutura de dados de floresta para conjuntos disjuntos constituidos por elementos num universo $\mathcal{A} = \{a_1, \ldots, a_n\}$ pode ser representada por um array $\mathtt{P} = (\mathtt{p}[1], \ldots, \mathtt{p}[n])$, onde $\mathtt{p}[i]$ representa o 'pai' do elemento a_i.

Considerando uma partição inicial `P` $= (10, 1, 10, 4, 5, 5, 7, 9, 9, 10)$ de $\mathcal{A}$, represente o vetor `P` após as operações

`Union(5,8) e Union(2,9),`

nesta ordem, assumindo as heurísticas de *união ponderada* com *compressão de caminhos*. Lembre-se que `Union`(i, j) é implementado como `Link`$($`Find`(i),`Find`$(j))$. Assumimos que, em caso de empate, `Link` deve escolher o representante do primeiro conjunto para representante da união.

PARTE II

▪ QUESTÃO 6 (2,5pt)

Considere a seguinte lista de Adjacências:

$1 \to 3, 4, 5, 6$	$4 \to 1, 5, 3$	$7 \to 9, 8$
$2 \to 8, 5, 6$	$5 \to 1, 2, 5, 8$	$8 \to 2, 5, 7, 3$
$3 \to 1, 8, 4$	$6 \to 2, 1$	$9 \to 7$

Preencha a tabela abaixo, *indicando a ordem de visita às arestas do grafo* especificado pela estrutura de Listas de Adjacências acima, durante uma *Busca em Profundidade*, iniciando no vértice 1.

Ordem	Extremoa 1	Extremo 2	Tipo (A/R)	Conteúdo da Pilha
1	1	3	A	1
2				
⋮	⋮	⋮	⋮	⋮
26				

Na tabela acima:

Extremo 1 = Vértice v_1 correspondente à entrada do array de listas de adjacências sendo visitada;
Extremo 2 = Elemento da lista de adjacências de v_1 sendo considerado no momento;
Tipo (A/R) = Tipo de aresta (se é uma aresta da Árvore ou uma aresta de Retorno);
Conteúdo da Pilha = Configuração atual da pilha de vértices.

▪ QUESTÃO 7 (2,5pt)

Dê exemplo de um grafo $G = (V, E)$, conexo e não-direcionado, com peso nas arestas dado pela função custo: $E \to \mathbb{Z}$ tal que:

i. $7 \leq |V| \leq 8$
ii. $|E| \geq 10$
iii. A função custo tem valores inteiros positivos e negativos, sendo que pelo menos 4 arestas têm peso positivo e pelo menos 3 arestas têm peso negativo.
iv. Não há em G ciclos de custo negativo (custo ciclo = soma dos custos das arestas no ciclo).
v. O Algoritmo *Dijkstra*(1) para calcular as distâncias do vértice 1 a cada um dos vértices do grafo calcula errado a distância de 1 a pelo menos dois vértices de G.

Você deve indicar claramente:

1) Os vértices cuja distância a 1 são calculados de forma errada,
2) As distâncias corretas de 1 a cada vértice do grafo e
3) O valor da distância que o Algoritmo Dijkstra encontra para cada vértice.

■ ■ ■