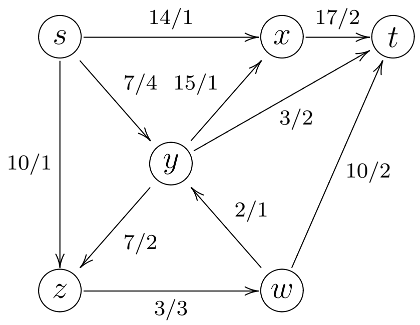
Departamento de Ciência de Computadores FCUP

Desenho e Análise de Algoritmos (CC211) 2013/14

2ºTeste (18.12.2013) *duração: 3h*

N.º Nome

1. Considere a rede de fluxo representada, em que c/f são pares capacidade/fluxo.



a) Apresente a rede residual correspondente.

b) Qual é o valor do fluxo que se pode obter se se realizar apenas uma iteração no algoritmo de Edmonds-Karp, a partir da situação indicada? Explique.

2. Considere o problema da determinação de uma árvore geradora mínima de um grafo $G = (V, E, d)$ não dirigido, conexo e com valores (inteiros) nos ramos. Que propriedade garante a correção das estratégias ávidas (*greedy*) seguidas nos algoritmos de Prim e de Kruskal?

3. Seja G um grafo dirigido $G = (V, E)$ com $V = \{v_1, v_2, v_3, \ldots, v_{10}\}$, $|V| = 10$, e tal que G tem exatamente quatro componentes fortemente conexas, cada uma com pelo menos dois vértices, e v_1 é acessível de v_5 e de v_{10}, mas nem v_5 nem v_{10} são acessíveis de v_1, o nó v_5 não é acessível de v_{10} nem v_{10} de v_5, o nó v_3 é acessível de v_1, mas v_1 não é de v_3.

a) Dê exemplo de um grafo nas condições indicadas e identifique as componentes.

b) Na sua determinação pelo algoritmo de Kosaraju-Sharir, por que ordem seriam obtidas? Explique.

4. Em cada alínea, dê exemplo, ou justifique a não existência, de uma *heap binária de mínimo* que contenha as chaves $-5, 7, 8, 3, 10, 1, 5, 12, -4, -9$ (possivelmente por outra ordem) e tal que:

a) inserir um elemento com chave -14 requer exatamente duas operações de SWAP.

b) remover o elemento com a chave -9 requer no máximo uma operação de SWAP na chamada HEAPIFY(1).

5. Na definição dada nas aulas para uma fila de prioridade Q suportada por uma *heap binária de mínimo* foram usados dois *arrays* (`a` e `pos_a`). Com que objetivo? De que modo são usados pelas operações EXTRACTMIN(Q) e DECREASEKEY($Q, w, dist[w]$) no algoritmo de Dijkstra?

6. Dê exemplo, ou justifique a não existência, de um grafo $G = (V, E, d)$, com $d(e) \in \mathbb{Z}^+$, não dirigido e conexo, em que existem dois nós x e y tais que, na aplicação do algoritmo de Dijkstra (para determinação de caminhos mínimos com origem num nó s), o nó x é retirado da fila de prioridade antes do nó y e o valor final de $dist[x]$ é maior do que o valor final de $dist[y]$.

7. Seja M um conjunto de n pessoas e T um conjunto de m tarefas. O grafo bipartido $G = (M \cup T, E)$ indica os pares $\langle x, y \rangle$ tais que a pessoa x pode realizar a tarefa y (ou colaborar na sua realização). Cada pessoa indicou o número máximo de tarefas em que pode estar envolvida. Algumas tarefas terão de ser realizadas por grupos de pessoas, sendo conhecido quantas pessoas são necessárias para cada tarefa. Não há outro tipo de restrições além destas. Pretendemos escrever um algoritmo para verificar se é ou não possível realizar todas as tarefas.

a) Dê exemplo de duas instâncias do problema (uma com solução e outra não), com oito pessoas e cinco tarefas, em que nem todas as tarefas requerem apenas uma pessoa.

b) Apresente (em linhas gerais mas com rigor) um algoritmo polinomial para resolver o problema. Explique.

N.º Nome

8. Seja $G = (V, E, t)$ um grafo dirigido finito, com $V = \{1, 2, \ldots, n\}$, que serve de modelo a uma rede de transportes, e em que $t(e) \in \mathbb{R}^+$ define a temperatura no ramo e, para todo $e \in E$. Seja $T(\gamma) = \text{maximo}\{t(e) \mid e \in \gamma\}$, para cada percurso γ. Para todos os pares $(i, j) \in V \times V$, com $i \neq j$, pretendemos encontrar **um percurso** γ_{ij} **tal que** $T(\gamma_{ij})$ **seja mínimo** quando considerados todos os percursos possíveis de i para j.

Tendo por base o algoritmo de Floyd-Warshall, apresente um algoritmo para calcular $T(\gamma_{ij})$ para cada par (i, j) nas condições do enunciado, bem como o segundo vértice no percurso γ_{ij} obtido. Justifique a correção do algoritmo que apresentou.

N.º ____ Nome ____

9. A função INVERTE(v,i,j) inverte o segmento $v[i], v[i+1], \ldots, v[j]$ de um vetor v de n elementos, **quando** $0 \leq i \leq j < n$. Após a chamada da função, o estado desse segmento é $(b_j, b_{j-1}, \ldots, b_{i+1}, b_i)$ se o estado inicial for $(b_i, b_{i+1}, \ldots, b_{j-1}, b_j)$.

```
INVERTE(v, i, j)
  Enquanto (i < j) fazer
      aux ← v[i];
      v[i] ← v[j];
      v[j] ← aux;
      j ← j − 1;
      i ← i + 1;
```

```
TRANSFORMA(v, n)
  d ← 0;
  m ← ⌊n/2⌋;
  Enquanto (d ≤ m) fazer
      INVERTE(v, d, n − 1 − d);
      d ← d + 1;
```

a) Indique um invariante de ciclo que permita demonstrar a correção de INVERTE(v,i,j). Prove-o por indução matemática e conclua que a função está correta.

b) Admita que a instrução $m \leftarrow \lfloor n/2 \rfloor$ é executada em tempo $O(1)$. Indique a complexidade temporal de TRANSFORMA(v,n). Justifique sucintamente mas com rigor e averigue se existe um algoritmo assintoticamente mais eficiente para efetuar a mesma transformação.

(Fim)