

---

## Exame de Qualificação

27/02/2014

Nome:

**Dicas gerais:** Leia todas as questões antes de começar; sempre justifique a sua resposta.

1. (2.5 pontos) Teorema: A classe de funções recursivas parciais é idêntica à classe de funções Turing-computáveis.
  - a) Defina essas classes de funções e faça um esquema da prova deste teorema, explicando o que deve ser provado em cada passo.
  - b) Explique por que este teorema é importante.
2. (1.5 pontos) Disserte sobre redução de problemas, tanto no contexto de indecidibilidade de problemas quanto no contexto da definição de classes de complexidade.
3. (1 ponto) Explique intuitivamente o que diz o Teorema de Cook-Levin e por que ele é relevante para a Ciência da Computação? (Não precisa mostrar definições formais nesta questão.)
4. (1.4 pontos) Considere o pseudocódigo abaixo. O algoritmo recebe como entrada um nó  $s \in V$  e um grafo direcionado pesado  $G = (V, A, w)$  onde  $V$  representa o conjunto de nós (considere que estes estejam enumerados de 1 a  $|V|$ ),  $A$  representa o conjunto de arcos e  $w(u, v) \in \mathbb{R}^+$  são valores atribuídos a cada arco  $(u, v) \in A$ .

```
1 INPUT:  $G=(V,A,w)$ ,  $s \in V$ 
2 OUTPUT:  $d$ 
3 for each  $v \in V$  do
4    $d[v] := \infty$ ;
5 end
6  $d[s] := 0$ ;
7  $Q := V$ ; //  $Q$  is initialized with all nodes from  $V$ 
8 while  $Q \neq \emptyset$  do
9    $u := \text{extrair-min}(Q)$ ; // node  $u$  with the minimum value of  $d$  is removed from  $Q$ 
10  for each  $v$  adjacent to  $u$  do
11    if  $d[v] > d[u] + w(u,v)$  then
12       $d[v] := d[u] + w(u, v)$ ;
13    end
14  end
15 end
```

- 
- a) Qual problema o algoritmo resolve?
- b) Analise com detalhes a complexidade do algoritmo quando  $Q$  é implementado com as seguintes estruturas de dados:
- heap mínimo binário;
  - $Q$  é um vetor sem organização específica dos nós (lista comum).
5. (1.8 pontos) Escolha um problema que pode ser resolvido por divisão e conquista ou programação dinâmica. Enuncie o problema, apresente o pseudocódigo do algoritmo que o resolve via uma destas duas técnicas, e analise a complexidade do seu algoritmo.
6. (1.8 pontos) Considere o problema de sequenciamento de intervalos (*interval scheduling problem*). São fornecidos  $n$  intervalos, sendo que cada intervalo  $i$  possui um tempo de início  $s_i$  e um tempo de término  $t_i$ . Dois intervalos são ditos consistentes se eles não usam um mesmo recurso ao mesmo tempo. Selecione a maior quantidade de intervalos consistentes possível.

Exemplo (ver figura): Considere que apenas uma sala de reuniões esteja disponível, mas muitas solicitações de uso da sala foram feitas (intervalos a-o). Deseja-se atender o maior número de solicitações sem que haja sobreposição de horário. A solução ótima para o problema atende 8 solicitações: b,c,e,f,g,h,i,j.

Sobre este problema, resolva as seguintes questões:

- a) Resolva de forma exata o problema via técnica de algoritmos gulosos: apresente pseudocódigo do seu algoritmo e análise de complexidade do mesmo.
- b) Prove que o seu algoritmo sempre retorna a solução ótima.

