Universidade Estadual de Maringá – Departamento de Informática
1747 – Teoria dos Grafos e Análise de Algoritmos
Ciência da Computação
Profs. Marco Aurélio e Mauro Henrique Mulati

Simulado 1

1. (1,0) Para cada sentença abaixo, diga se ela é verdadeira ou falsa. Justifique sua resposta.
   (a) $\lg n = \Theta(\log n)$
   (b) $n! = \Omega((n+1)!)$
   (c) $4^n = O(3^{2n})$

2. (1,5) Resolva as seguintes recorrências:
   (a) $T(n) = 2T(n/2) + n \lg n$
   (b) $T(n) = 9T(n/3) + n^2$
   (c) $T(n) = 2T(n/3) + n$

3. (1,5) Dado dois conjuntos A e B de valores reais no intervalo de 0 a 10 com duas casas decimais, ambos de tamanho n, o seguinte algoritmo verifica quantos elementos os dois conjuntos têm em comum. Considerando este algoritmo, responda as perguntas (escreva as respostas em termos de n):
   (a) Em quais situações ocorre o pior caso? Quantas vezes exatamente a comparação da linha 4 é executada nestas situações?
   (b) Em quais situações ocorre o melhor caso? Quantas vezes exatamente a comparação da linha 4 é executada nestas situações?
   (c) Qual é o tempo de execução do algoritmo? Dê a sua reposta em termos da notação O com justificativa.

```
contar-os-elementos-em-comum(A, B, n)
1  cont = 0
2  for i = 1 to n
3    for j = 1 to n
4      if A[i] == B[j]
5        cont = cont + 1
6        break            # para a repetição da linha 3
7  return cont
```

4. (2,0) Em uma análise do tempo de execução do algoritmo heapsort, foi considerado que o tamanho do heap não é alterado durante a execução, o tempo de execução obtido foi de $O(n \lg n)$. Faça uma nova análise do tempo de execução do heapsort considerando que o tamanho do heap é alterado (na linha 4) a cada iteração. Existe diferença no tempo de execução encontrado? Qual é a intuição por trás deste resultado?

```
heapsort(A)
1  build-max-heap(A)
2  for i = n downto 2 do
3    trocar(A[1], A[i])
4    A.tamanho-do-heap = A.tamanho-do-heap - 1
5    max-heapify(A, 1)
```

5. (4,0) Proponha um algoritmo $\Theta(n)$ para o problema do exercício 3. Apresente a análise do tempo de execução e mostre que seu algoritmo funciona corretamente.