

Estruturas de Dados II (DEIN0083) 2017.2
Curso de Ciência da Computação
1ª avaliação

Prof. João Dallyson Sousa de Almeida

Data: 09/10/2017

Aluno: Pedro Thiago

Matrícula:

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Regras durante a prova:

- É vetada: a consulta a material de apoio, conversa com colega e a utilização de dispositivos eletrônicos. A não observância de algum dos itens acima acarretará a anulação da prova.

I. (1.0pt) Para cada função $f(n)$ abaixo, dê um limite superior assintótico usando a notação "Big-O". Você deve dar o limite mais próximo possível (Não tente usar 100^n como resposta em todas as opções).

- a) $f(n) = 200n^3 - 5n^3 + 8n^2$
b) $f(n) = 50n^4 - 50n^4 + 4n^2$
c) $f(n) = (\log n)(n^2 + n^3)$
d) $f(n) = 0.003n + 200 \cdot 3^n$

II. (1.0pt) Determine o tempo de execução do algoritmo abaixo. Justifique sua resposta.

```
int k=1, count=0;
for (int i=0; i<n; i++) {
    k=2*k;
}
for (int j=k; j>1; j=j/3) {
    count++;
}
```

III. (1.0pt) Sobre a análise assintótica indique se cada afirmativa é verdadeira ou falsa e justifique sua resposta:

- a) $7n^2 + 2n = \Theta(n)$ b) $5n + 20\log n = \Omega(n)$

IV. (1.0pt) Utilize o teorema Mestre para analisar assintoticamente as recorrências a seguir:

Teorema 4.1 (Teorema mestre)

Sejam $a \geq 1$ e $b > 1$ constantes, seja $f(n)$ uma função e seja $T(n)$ definida sobre os inteiros não negativos pela recorrência

$$T(n) = aT(n/b) + f(n),$$

onde interpretamos n/b com o significado de $\lfloor n/b \rfloor$ ou $\lceil n/b \rceil$. Então, $T(n)$ pode ser limitado assintoticamente como a seguir.

1. Se $f(n) = O(n^{\log_b a - \epsilon})$ para alguma constante $\epsilon > 0$, então $T(n) = \Theta(n^{\log_b a})$.
2. Se $f(n) = \Theta(n^{\log_b a})$, então $T(n) = \Theta(n^{\log_b a} \lg n)$.
3. Se $f(n) = \Omega(n^{\log_b a + \epsilon})$ para alguma constante $\epsilon > 0$, e se $af(n/b) \leq cf(n)$ para alguma constante $c < 1$ e para todo n suficientemente grande, então $T(n) = \Theta(f(n))$.

- a) $T(n) = 4T(n/2) + n^2 \log n$ b) $T(n) = 8T(N/2) + n \log n$

- 2.0
- V. (2.0pt) Utilize o algoritmo de ordenação HeapSort para ordenar o vetor [9, 2, 3, 8, 1, 7, 5]. Apresente, passo a passo (árvore intermediária) a estrutura da Heap (MaxHeap) após a construção. Apresente a solução da ordenação mostrando passo a passo (ilustrando a árvore e o vetor em cada iteração).
- 1.5
- VI. (2.0pt) Escreva o método particiona do algoritmo de ordenação Quicksort. Considere a escolha do pivô como a mediana de três elementos (primeiro, central e o último).
- 1.5
- VII. (2.0pt) Explique e utilize o algoritmo do RadixSort para ordenar a sequência de chaves a seguir: [337, 192, 785, 124, 537, 912]. Apresente a saída após utilizar: InsertionSort (DDX), CountSort (DXD) e o ShellSort (XDD) como segundo algoritmo para ordenação dos dígitos. Apresente o conteúdo do vetor a cada passo intermediário. (exibir execução passo a passo). É recomendado utilizar o ShellSort? Justifique sua resposta.