



Universidade Federal de Roraima  
Departamento de Ciência da Computação  
Análise de Algoritmos



DISCIPLINA: Análise de Algoritmos – DCC606

1ª Lista

Prazo de Entrega: 30/06/2016

ALUNO(A): \_\_\_\_\_ NOTA: \_\_\_\_\_

**ATENÇÃO:** Descrever as soluções com o máximo de detalhes possível, no caso de programas inclusive a forma como os testes foram feitos. Todos os artefatos (relatório, código fonte de programas, e outros) gerados para este trabalho devem ser adicionados em um repositório no site `github.com`. Para as questões que requisitarem a escrita/implementação de programas devem ser feitas na linguagem de programação C e deve ser apresentado: o modo de compilar/executar o programa; a linha de comando para executar o programa; e um exemplo de entrada/saída do programa.

**[QUESTÃO – 01]** Para cada afirmação, indique se a mesma é falsa ou verdadeira, justificando sua resposta:

(A)  $n + (\log n) = \Theta(n)$

(B)  $n^2 = o(n^3)$

(C)  $(n + 1)^2 = O(2n^2)$

(D) Se  $f(n) = n - 300$  então  $f(n)$  é  $\Omega(300n)$  e  $f(n)$  é  $O(300n)$

**[QUESTÃO – 02]** Para o problema de ordenação: especificar o problema (descrição, instância, entrada, saída, objetivo), dar um exemplo gráfico e citar o limite inferior (LI) e superior (LS) do problema.

**[QUESTÃO – 03]** Obtenha a função de custo e a complexidade de tempo para os códigos apresentados abaixo:

(A)

```
1.VerificaAlgo (n: int);
2.   i, j, k, l: int;
3.   para l := 1 TO 10.000 faça
4.       para i := 1 TO n-5 faça
5.           para j := i+2 TO n/2 faça
6.               para k := 1 TO n faça
7.                   {Inspecione elemento}
```



(B)

```
1. Ordena (A: vet[1..n] de int; i, j: int); { n uma potência de 3 }
2.   se i < j então
3.       m := ((j-i) + 1)/3;
4.       Ordena(A, i, i+m-1);
5.       Ordena(A, i+m, i+2m-1);
6.       Ordena(A, i+2m, j);
7.       Merge(A, i, i+m, i+2m, j);
8.       { Merge intercala A[i..(i+m-1)], A[(i+m)..(i+2m-1)] e A[i+2m..j] em A[i..j] a um custo
      (( 5n/3 ) - 2 ) }
```

(C)

```
1. void FazAlgo (int n) {
2.   int i, j, k;
3.   FOR (i= 1; i<n - 1; i++) {
4.     FOR (j= i + 1; j<= n; j++) {
5.       FOR (k = 1; k<=j;k++) {
6.         Algum comando de custo O(1)
7.       } } }
8. }
```

(D)

```
1. Hanoi(n, Origem, Destino, Auxiliar){
2.   se n > 0{
3.     Hanoi(n-1,Origem,Auxiliar, Destino)
4.     move o disco da Origem para o Destino
5.     Hanoi(n-1, Auxiliar, Destino, Origem)
6.   }
7. }
```

**[QUESTÃO – 04]** Descreva a técnica de divisão e conquista. Implemente um algoritmo utilizando divisão e conquista para encontrar o maior e o menor elemento em uma lista.

**[QUESTÃO – 05]** Implemente os algoritmos de ordenação: Insertion Sort; QuickSort; e MergeSort. Apresente as complexidades dos algoritmos. Apresente um estudo empírico para analisar o tempo de execução dos algoritmos. Cada algoritmo de ordenação deve ser executado com entradas de diferentes tamanhos: 100, 500, 5000, 10000 e 30000. Também se deve utilizar diferentes configurações para cada tamanho de entrada para a ordenação: números aleatório; em ordem crescente; e em ordem decrescente. Crie um gráfico de linhas para cada configuração de entrada com os tempos de ordenação para comparar os algoritmos, onde cada linha do gráfico irá representar um algoritmo.





**[QUESTÃO – 06]** Descreva os passos para ordenação de um vetor usando o algoritmo Quick Sort.

**[QUESTÃO – 07]** Descreva as regras de balanceamento em uma árvore vermelho e preto (red and black). Adicionalmente, apresente de forma gráfica a inserção dos seguintes valores em uma árvore vermelho e preto: 11; 7; 8; 14; 4; 15; 1; 2; 5.

**[QUESTÃO – 08]** Implemente a operação de inserção da árvore AVL e árvore vermelho e preto. Apresente um estudo empírico para obter custos de inserção na medida em que o número de elementos da árvore aumenta. Gere gráficos para mostrar o custo médio de inserção para tamanhos distintos de  $N$  (exemplo: de 10 a 1000000). Apresente uma análise de comparação entre árvore AVL e árvore vermelho e preto em relação ao tempo de execução. Adicionalmente, apresente a complexidade da operação de inserção da árvore AVL e árvore vermelho e preto.

**[QUESTÃO EXTRA]** Pesquise e apresente a relação entre a série harmônica e a escala musical.

