Questão 01

A) $n + (\log n) = \theta(n)$
   $0 \leq c1 \times n \leq n + \log n \leq c2 \times n$
   $n = 2$ temos que:
   $0 \leq 2c1 \leq 2 + 1 \leq 2c2$
   $0 \leq 2c1 \leq 3 \leq 2c2$
   Se $c1 = 1$ e $c2 = 2$:
   $0 \leq 2 \leq 3 \leq 4$

B) $n^2 = o(n^3)$
   $\lim_{n \to +\infty} n^2 / n^3 = 0$

C) $(n + 1)^2 = O(2n^2)$
   $0 \leq (n + 1)^2 \leq 2cn^2$
   $0 \leq n^2 + 2n + 1 \leq 2cn^2$
   $n = 5$ temos que:
   $0 \leq 25 + 10 + 1 \leq 50c$
   $36 \leq 50c$
   $c > 50/36$

D)
   $f(n) = n - 300$. : $\Omega(300\ n)$ e $O(300\ n)$
   $\Omega(g(n)) = 0 \square c\ 1\ g(n) \square f(n) \square c\ 2\ g(n)$
   $0 \square c\ 1\ 300n \square n - 300 \square c\ 2\ 300\ n$
   $c\ 1 = 0$ $n = 400$ $c\ 2 = 1$
   $0 \square 0 \square 100 \square 120000$
   . :ok

Questão 02

Temos que se existe um vetor com elementos ordenáveis, então temos o problema da ordenação onde se deseja organizar esses elementos de algum jeito, dado um vetor de x1 até xn, para que estes dados sejam ordenados em suas posição várias repetições são feitas até o objetivo ser atingido. exemplo: caso o vetor (10 ,2 ,1) seja ordenado em ordem crescente é preciso que sejam realizadas n^2 comparações porque o vetor está em ordem decrescente. No caso de um vetor (1, 2, 10) é preciso que sejam realizadas apenas n comparações, onde o vetor é passeado checando se estão todos em seus respectivos lugares. Os limites superior e inferior são respectivamente $O(n^2)$ e $O(n)$.

Questão 03

A)

$$\sum_{l=1}^{10000} \sum_{i=1}^{n-5} \sum_{j=i+2}^{n/2} \sum_{k=1}^{n} 1 = 10000n^2 - 50000n$$

B)

$$\begin{cases} 1 \quad se \quad n=1 \\ 3T\left(\frac{n}{3}\right) + \frac{5n}{3} - 2 \quad se\, n>1 \end{cases}$$

Após as recorrências se encontra:

$$3^k T\left(\frac{n}{3^k}\right) + k\frac{5n}{3} - 2\sum_{i=0}^{k-1} 3^i$$

$$sendo \quad k = \log_3 n, \quad se\, obtém:$$

$$\frac{n}{2} \; - \; n\; long_{3}n\; \frac{5}{3} + \frac{3}{2}$$

C)

$$\sum_{i=1}^{n-2} \sum_{j=i-1}^{n} \sum_{k=1}^{j} 1 = \frac{n^3}{3} - \frac{3n^2}{2} + \frac{17n}{2} - 1$$

D)

$$T(n) = \begin{cases} 1 \quad se \quad n=0 \\ 2T(n-1)+1 \quad se \quad n>1 \end{cases}$$

Após as recorrências se encontra:

$$2^k T(n-k) + \sum_{i=0}^{n-1} 2^i$$

$$sendo \quad k=n, \quad se\, obtém:$$

$$2^{n+1} - 1$$

Questão 04

Os algoritmos de divisão e conquista consistem em dividir as tarefas grandes em tarefas menores, realizando chamadas recursivas em cada uma das partes até que o objetivo seja atingido.

```
int maior(int * v, int s, int e) {
  if (s == e)
    return v[e];
  int meio = (e - s)/2 + s ;
  int a = maior(v, s, meio);
  int b = maior(v, meio+1, e);if (a > b) return a;
  return b;
}

int menor(int * v, int s, int e) {
  if (s == e)
    return v[e];
int meio = (e - s)/2 + s ;
int a = menor(v, s, meio);
int b = menor(v, meio+1, e);
if (a > b) return b;
  return a;
}
```

Questão 05

Quick sort:

Pior caso - O(*n log 2n*);

Melhor caso - O(*log n*);

Merge sort:

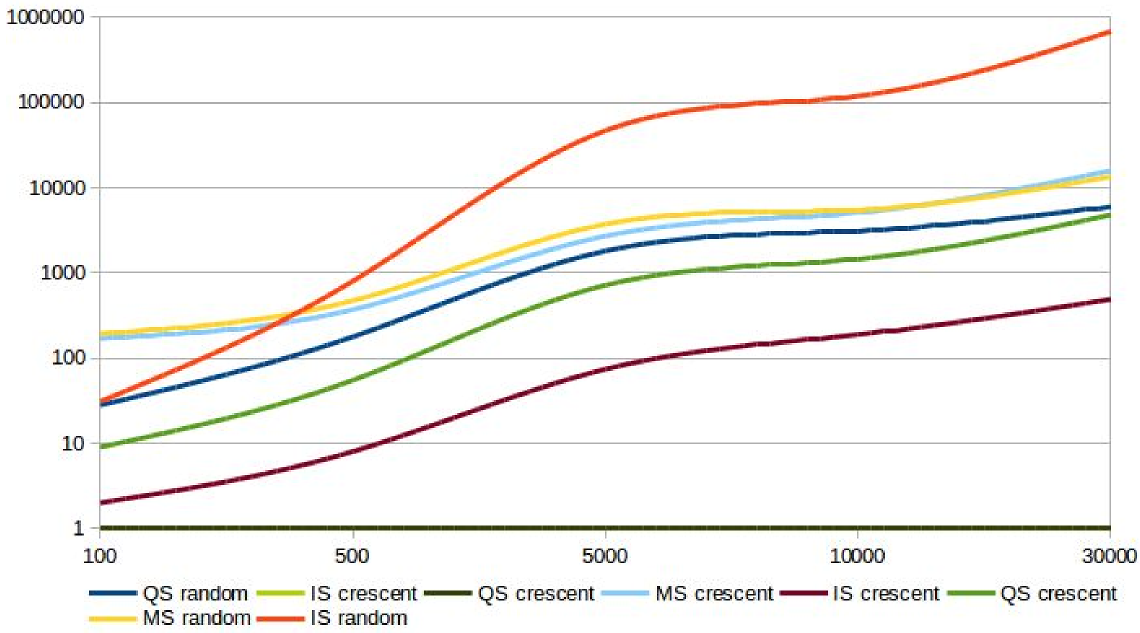
Pior caso - O(*n log n*);

Melhor caso - O(*n*);

Insertion sort:

Pior caso - O(n^2);

Melhor caso - O(*n*);



Questão 06

Escolhe um pivô, coloca todos os elementos menores que o pivô para à esquerda e os maiores à direita, depois o mesmo se repete até que o vetor esteja ordenado.

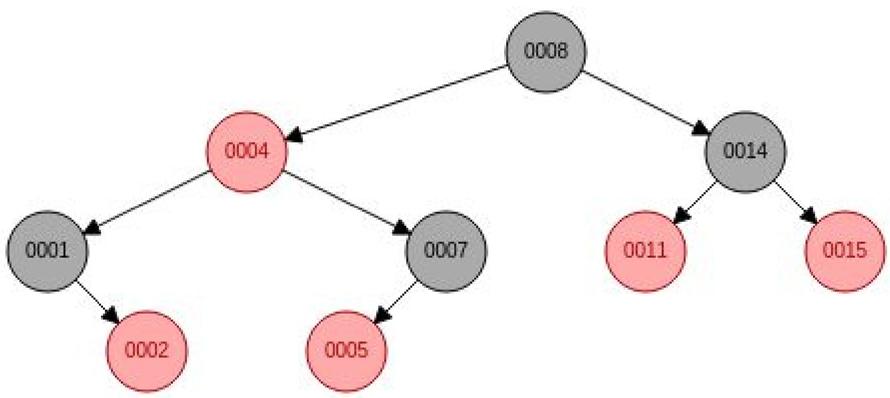
Questão 07

as propriedades da árvore rubro negra são:

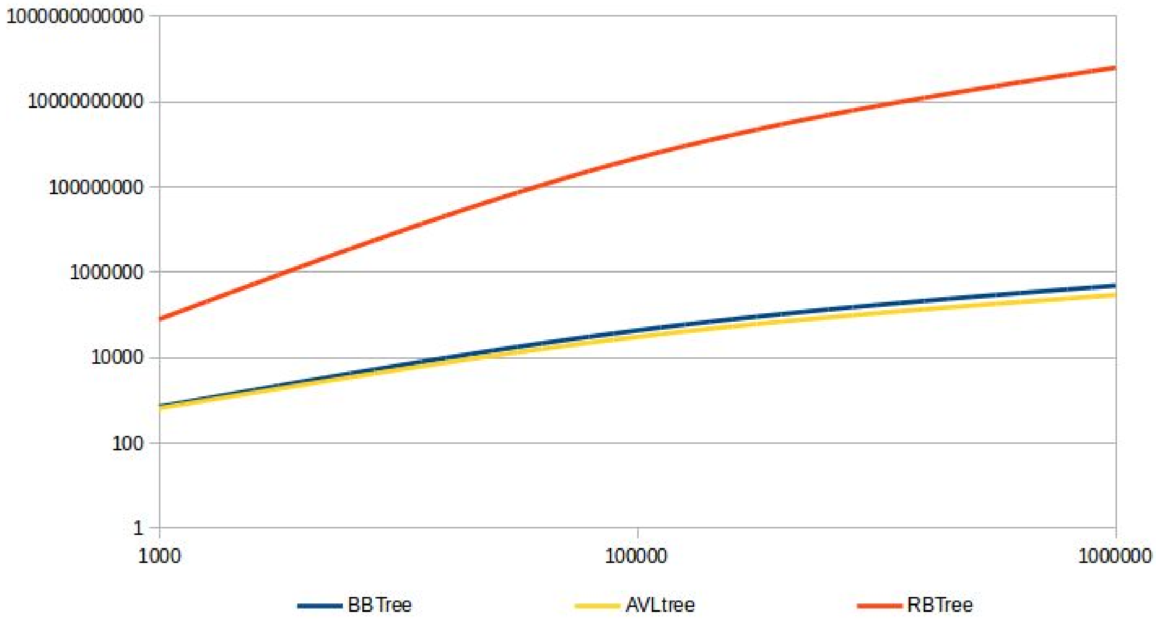
1 - toda raíz é preta;

2 - todo filho de nó vermelho é preto;

3 - a quantidade de nós pretos em cada caminho da árvore é a mesma.



Questão 08



Questão 09

Series harmônicas são séries divergentes infinitas

$$\sum_{i=1}^{\infty} \frac{1}{n} = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \dots$$

O nome desta série vem do conceito de sobretons, que na música são chamados de harmônicas, são
comprimentos de onda de sobretons de uma corda vibrando que são 1/2, 1/3, 1/4...O espaçamento das notas em uma escala musical segue a série harmônica em relação aos sobretons
de cada nota, o que garanta um som harmônico na escala.

Uma série harmônica é uma sequência de sons onde a frequência base de cada som é múltipla da integral da frequência base mais baixa