



Curso de Tecnologia em Sistemas de Computação  
Disciplina: Estrutura de Dados e Algoritmos  
AP3 - Primeiro Semestre de 2018

Nome -

Assinatura -

---

Observações:

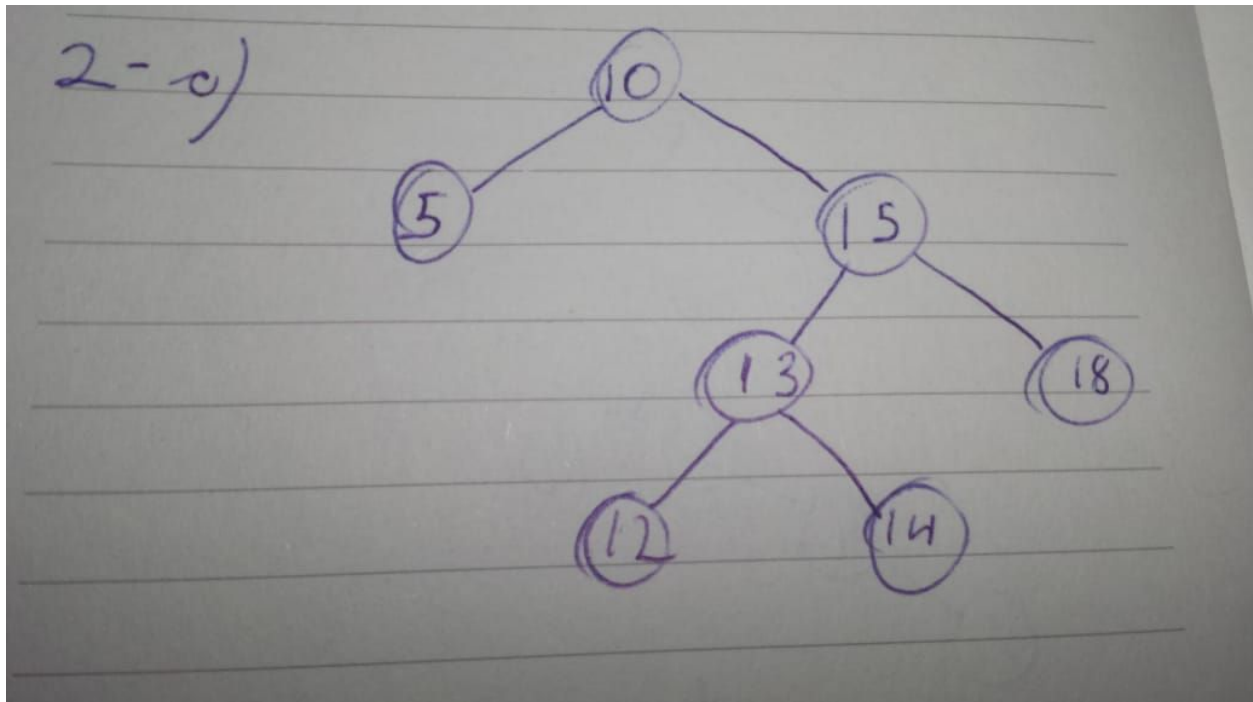
1. Prova sem consulta e sem uso de máquina de calcular.
  2. Use caneta para preencher o seu nome e assinar nas folhas de questões e nas folhas de respostas.
  3. Você pode usar lápis para responder as questões.
  4. Ao final da prova devolva as folhas de questões e as de respostas.
  5. Todas as respostas devem ser transcritas nas folhas de respostas. As respostas nas folhas de questões não serão corrigidas.
-

1. Forneça as definições dos seguintes conceitos:
  - (a) (1,0) Árvore binária de busca.
  - (b) (1,0) Algoritmo ótimo para um problema.
  - (c) (1,0) Limite inferior de um problema.
2. Para cada um dos itens abaixo, desenhe uma árvore binária de busca  $T$  de **altura 4** (colocando valores de chaves nos respectivos nós), atendendo às condições de cada caso:
  - (a) (0,5)  $T$  é uma árvore estritamente binária, porém não completa.
  - (b) (0,5)  $T$  é uma árvore completa, porém não estritamente binária.
  - (c) (0,5)  $T$  é uma árvore zigue-zague.
3. (1,5) Desenhe uma árvore  $B$  de ordem 3 e altura 3 que contenha um número mínimo de chaves. Os valores das chaves ficam à sua escolha. (Desenhe detalhadamente os ponteiros, de acordo com a definição.)
4. Para cada sequência abaixo, responda se ela corresponde ou não a um *heap* (lista de prioridade). Justifique brevemente.
  - (a) (1,0) 33 32 27 31 29 28 25 30 26
  - (b) (1,0) 33 32 27 31 29 26 25 30 28
5. (2,0) Desenhe a árvore AVL resultante da inclusão das chaves com valores respectivamente iguais a 5, 4, 3, 10, 8 e 15, a partir de uma árvore, inicialmente vazia. Detalhe todos os passos do processo.

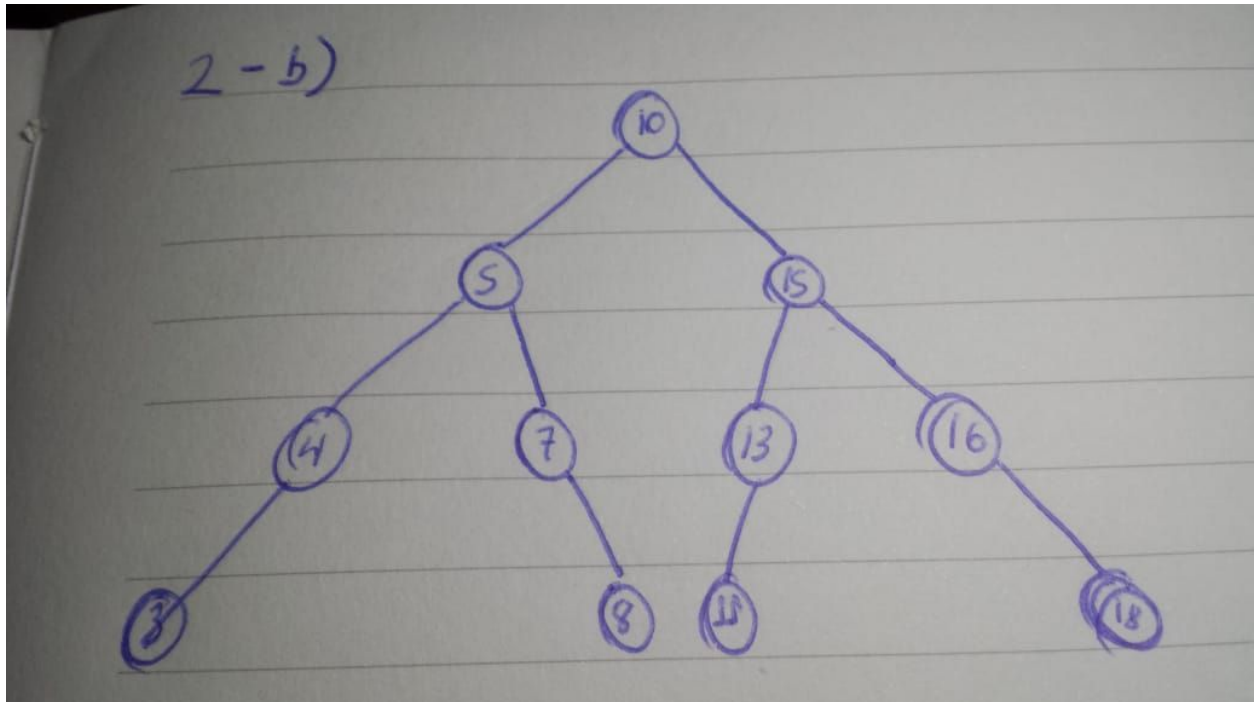
1 -

- a) Uma árvore binária de busca é aquela em que os nós da subárvore esquerda possuem um valor menor que a raiz e todos os nós da sua subárvore direita tem um valor maior que a raiz.
- b) Um algoritmo ótimo apresenta a menor complexidade dentre todos os possíveis algoritmos para o mesmo problema.
- c) Um limite inferior para P é uma função  $I$ , tal que a complexidade de pior caso de qualquer algoritmo que resolva P é  $\Omega(I)$ .

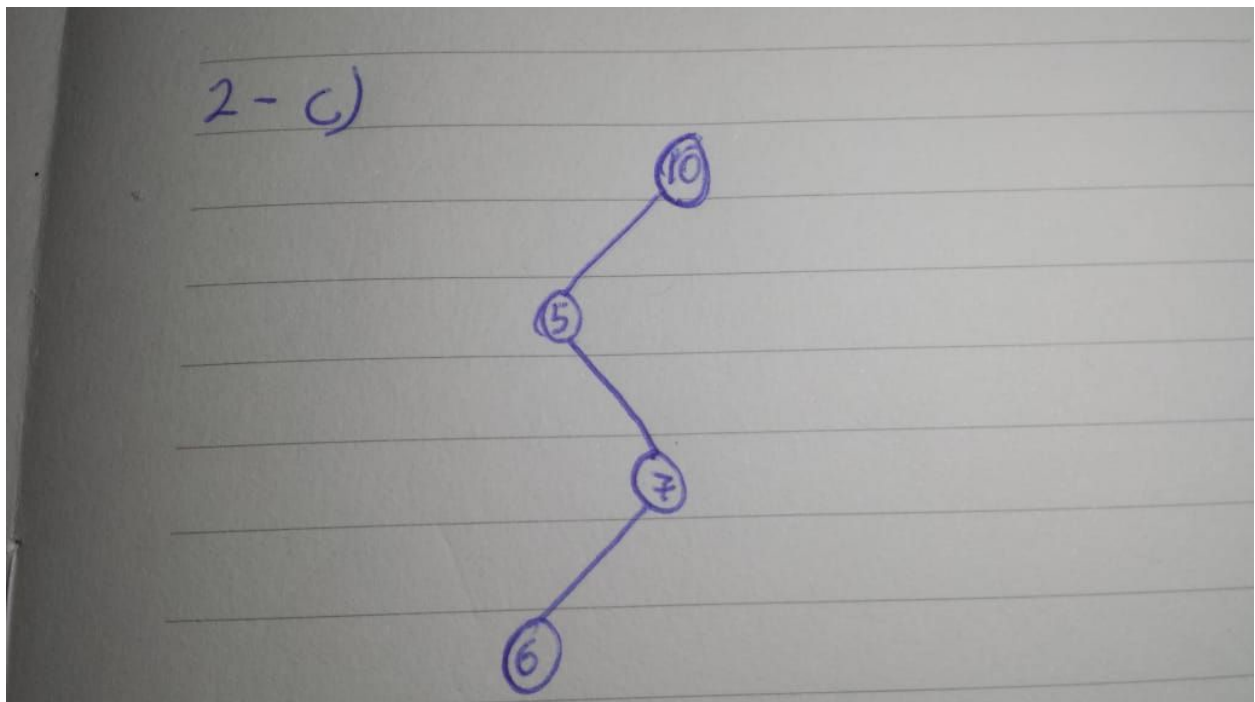
2 - a)

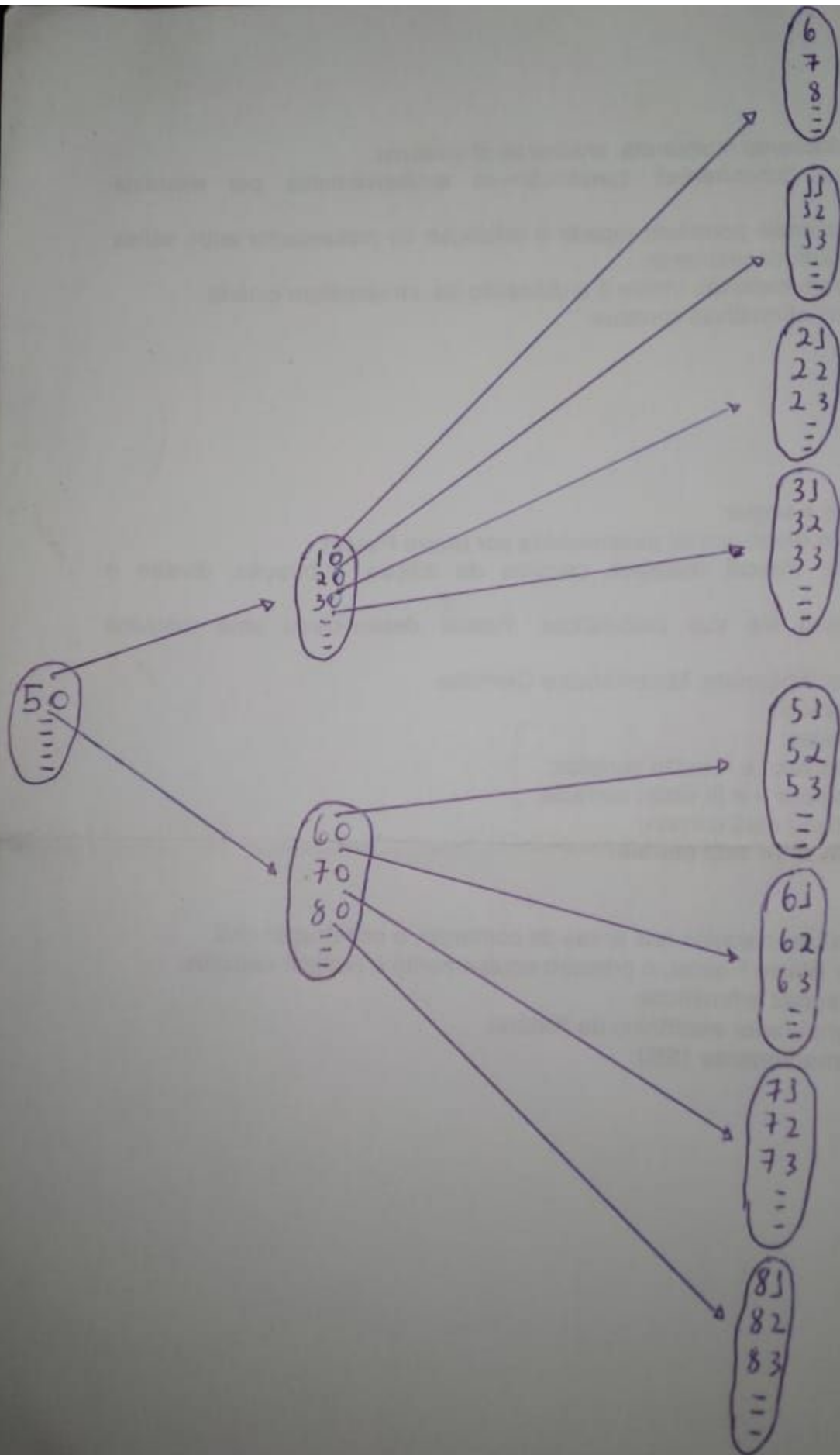


2 - b)



2 - c)





4 - a)  $s_6 > s_3$ , não é heap.

4 - b) É Heap.

5)

