

## Prova P2 de Estruturas de Dados (INF1010)

Profs Augusto Baffa e Luiz Fernando Seibel

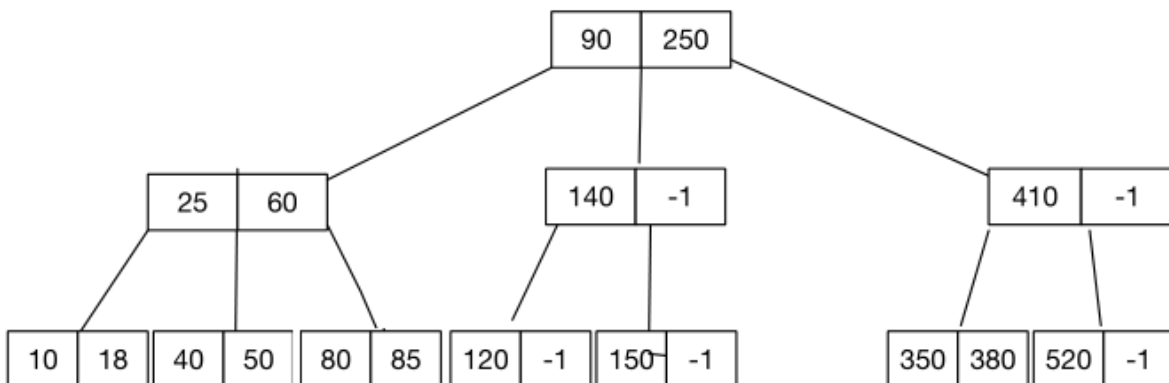
08 de novembro de 2021

- Responda as questões em uma folha de papel (utilizando sua caligrafia). As respostas podem ser escritas a lápis ou caneta.
- Indique claramente a questão que está respondendo e, quando cabível, os passos que seguiu para chegar a sua resposta.
- Utilize seu celular para bater fotos da folha de respostas e verifique se a foto está legível.
- Anexe a foto da resposta na tarefa relativa.

1. Desenhe uma árvore rubro-negra através da inserção das chaves 600,200,110,520, 180, 610. Indique uma sequência de chaves que provoque rotação dupla à esquerda e outra que provoque apenas a mudança de cores. (2.5 pontos)

2. Dada a árvore B a seguir, como ficará a árvore se realizarmos cada uma das operações abaixo (as operações são acumulativas): (2.5 pontos)

- Obs: os valores -1 demonstram nós que tiveram valores removidos anteriormente. Assim, a árvore manteve a configuração apresentada. O -1 representa que o espaço está vazio.



- a) inserção com chave 180 (0.5 ponto)
  - b) inserção com chave 45 (0.5 ponto)
  - c) remoção com chave 80 (0.5 ponto)
  - d) remoção com chave 410 (0.5 ponto)
3. Uma tabela de dispersão de tamanho 11 é implementada com encadeamento interior, utilizando todo o espaço de endereçamento para tratar colisões (ou seja sem área de overflow). A função de dispersão é a seguinte: (2 pontos)

$$h(x) = ( (x \% 11) + k) \% 11, \text{ onde } k=0,1,2,3,\dots \text{ tentativas}$$

- a) Desenhe a estrutura de dados após a inserção das chaves (nesta ordem): (1 ponto)

9, 10, 17, 3, 2, 13, 15, 25, 14

- Explique como cada colisão foi tratada.
  - Inicialmente as posições da tabela são inicializadas com -1.
  - O bit de exclusão é inicializado com 0.
- b) Explique, com base no exemplo anterior, como executar sucessivamente as seguintes operações: (1 ponto)
- i. Remoção da chave 3.
  - ii. Inserção da chave 4.

4. Considere o heap com a seguinte estrutura: (3 pontos)

```
typedef struct _heap Heap;
struct _heap {
    int max; /* tamanho maximo do heap */
    int pos; /* proxima posicao disponivel no vetor */
    int* info; /* vetor dos elementos do heap */
};
```

- a) Escreva uma função em C que retorne o número de elementos menores que um elemento de valor x de um heap min, sendo heap e x enviados como parâmetros da função. A função deve examinar o menor número possível de elementos. (1.5 pontos)
- b) Desenhe a estrutura da heap após a inserção da sequência 10-50-20-5-60-80-15 (0.5 ponto)
- c) Execute o algoritmo heapsort na heap criada durante a questão 4.b e explique como o resultado é organizado em cada um dos seus passos (1 ponto)