

## Lista 8 – Árvore Rubro-Negra

ESTRUTURA DE DADOS – Pedro Nuno Moura  
Monitor: Celio Ferreira Camara Junior

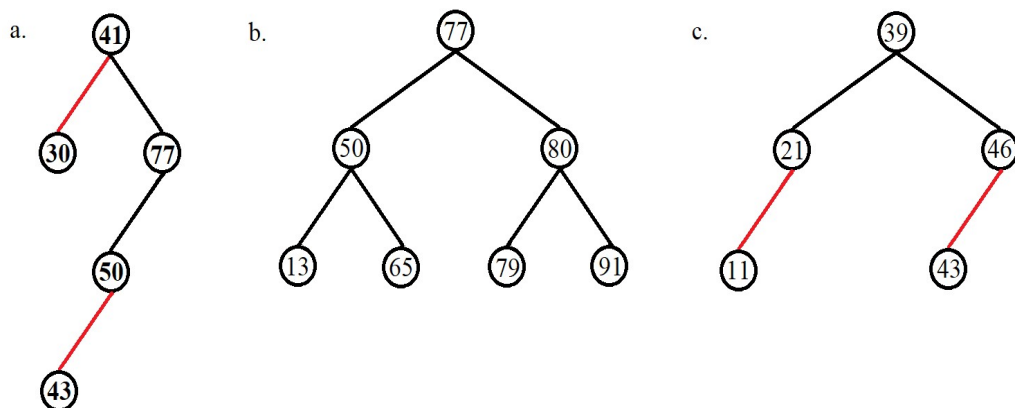
Para as questões abaixo, considere a implementação de Árvore Binária de Busca vista em sala de aula.

**ATENÇÃO:** Para todas as questões de implementação, deve ser informada e explicada a complexidade computacional alcançada.

1) Monte uma árvore binária de busca para cada uma das sequências abaixo. Repita com árvores rubro-negras. Compare os dois tipos de árvore para cada sequência.

- a. 70, 19, 11, 89, 13, 17
- b. 23, 15, 41, 11, 13, 14, 50, 38
- c. 1, 2, 3, 4, 5, 6, 7

2) Indique, para cada uma das árvores abaixo, se é ou não uma árvore rubro-negra. Justifique as que não forem.



3) Desenvolva um método que retorne a altura negra de uma árvore rubro-negra. A altura negra corresponde ao número de nós negros que são encontrados da raiz até uma folha qualquer. O método deve adotar o seguinte protótipo:

```
public int alturaNegra();
```

4) Modifique a implementação de árvore rubro-negra vista em aula para permitir que o método `get` possa obter o último nó acessado da árvore (inserido ou recuperado através do próprio `get`) em tempo constante, isto é,  $O(1)$ .

5) Faça uma implementação de árvore rubro-negra que não use uma variável para armazenar a cor do nó. Considere o seguinte: para tornar um nó vermelho, inverta seus filhos. Assim, para testar se um nó é vermelho, verifique se o filho esquerdo é maior que o filho direito.

6) Implemente uma árvore 2-3 diretamente utilizando dois tipos de nós diferentes (nós binários e nós ternários).