

Lista de exercícios P1 – ED II
Profa. Valeria Farinazzo

1. Percorra a árvore em pré-ordem, em-ordem e pós-ordem.

a) Pré-ordem:

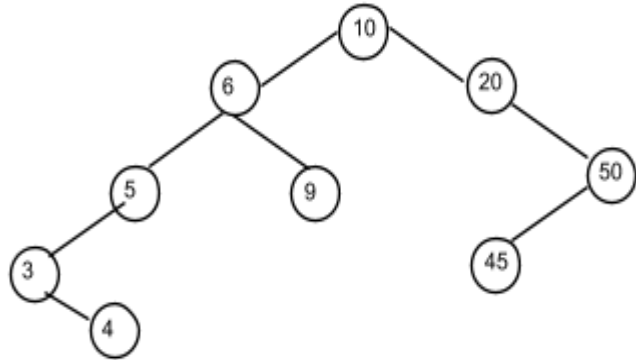
R: 10 - 6 - 5 - 3 - 4 - 9 - 20 - 50 - 45

b) Em-ordem:

R: 3 - 4 - 5 - 6 - 9 - 10 - 20 - 45 - 50

c) Pós-ordem:

R: 4 - 3 - 5 - 9 - 6 - 45 - 50 - 20 - 10



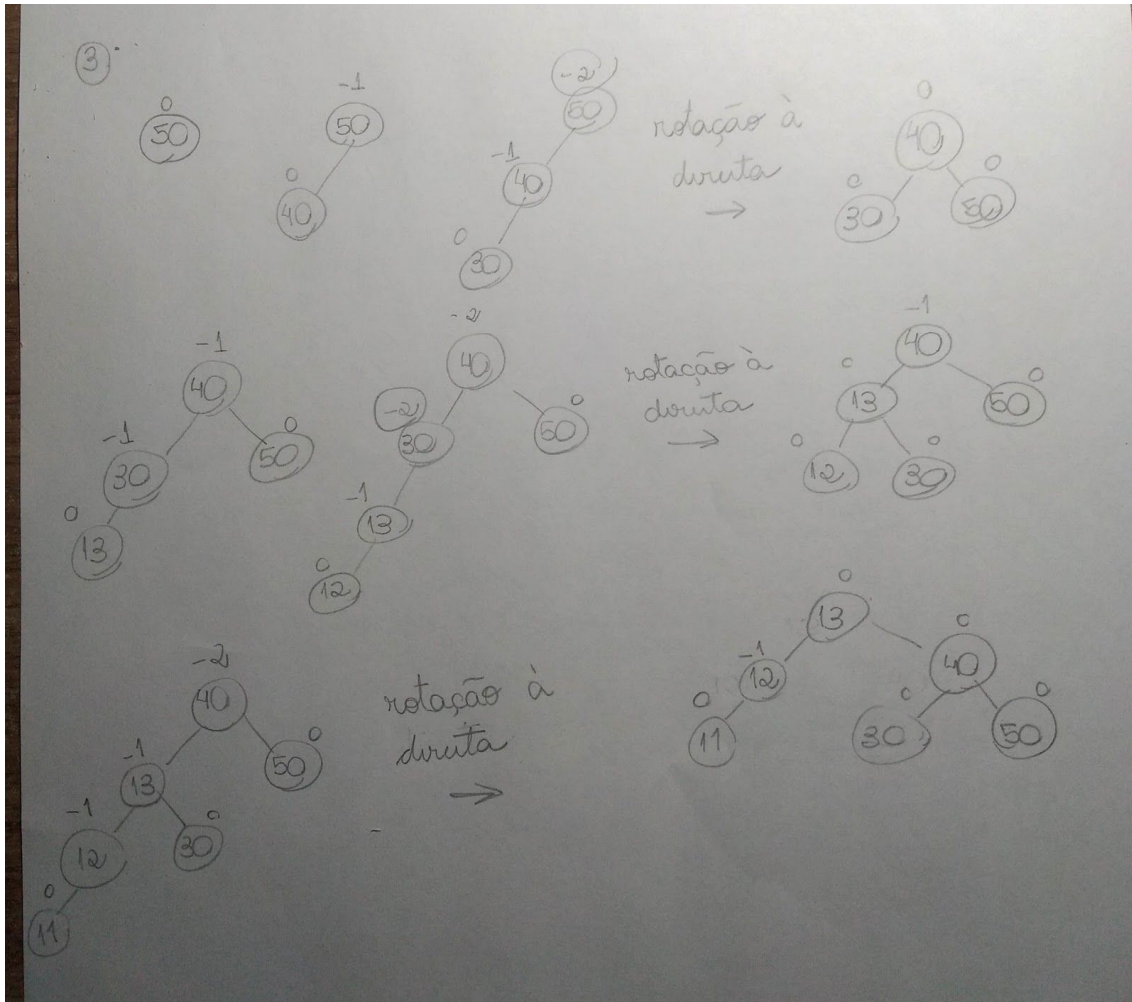
2. A partir do código para imprimir uma árvore binária de busca, de maneira recursiva, escreva o código para imprimir de maneira decrescente).

```
void AVL_tree::printrecursive(AVL_node *r)
{
    if(r!=NULL){
        cout<<r->key;
        printrecursive(r->left);
        printrecursive(r->right);
    }
}
```

```
void AVL_tree::printInDecreasingOrder(AVL_node *r) {
    if (r != NULL) {
        printInDecreasingOrder(r->right);
        cout << r->key;
        printInDecreasingOrder(r->left);
    }
}
```

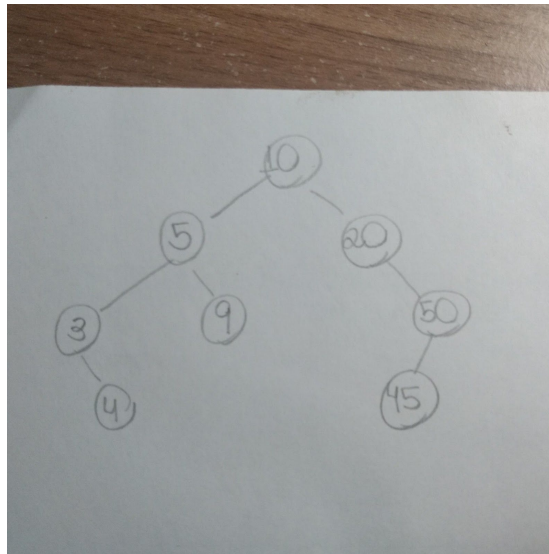
Lista de exercícios P1 – ED II
Profa. Valeria Farinazzo

3. Construa uma árvore AVL, de maneira que ela sempre fique balanceada (isto é, se a inserção de um elemento causar o desbalanceamento da árvore, é necessário balanceá-la novamente). Apresente a árvore passo-a-passo. Os dados são (nesta ordem): 50, 40, 30, 13, 12, 11

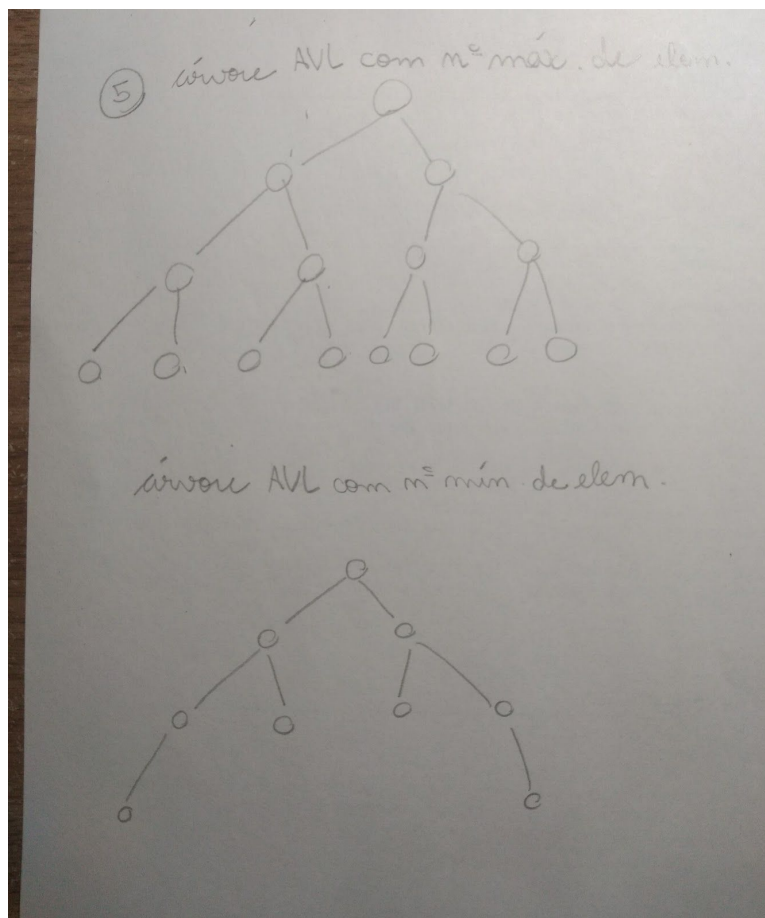


Lista de exercícios P1 – ED II
Profa. Valeria Farinazzo

4. Dada a árvore do exercício 1, remova o elemento 6.



5. Considere uma árvore de busca binária e uma árvore AVL de altura 4. Mostre como seria a estrutura destas duas árvores com o número mínimo de nós e com o número máximo de nós.



Lista de exercícios P1 – ED II
Profa. Valeria Farinazzo

6. Mostre os resultados de inserir as chaves a seguir em uma árvore B de ordem 2 inicialmente vazia: 14, 39, 1, 6, 41, 32, 8, 38, 43, 3, 36. Após essas inserções, mostre os resultados de remover as chaves ímpares nessa estrutura.