Name: _____

You may refer to the textbook, your notes, the assigned reading, and any other material you wish during this exam, but the material must not be shared. You may use your computer and the Internet, but only as a reference library. However, the exam is designed to be answered without using any such references.

1. (1 pt.) **Important note** for the questions on tree operations below.
   - You must use precisely the algorithms described in the textbook.
   - You must depict the intermediate states of the tree including at least the states before and after each insertion and deletion, and before and after each rotation or color-changing operation.
   - The type of each rotation and color-changing operation, and the root of the subtree to which it is applied, must be clearly identified.
   - Answers that do not include these details clearly are likely to receive zero credit.

   Write your name in the space provided above.

2. (9 pts.) Depict the *AVL tree* resulting from the insertion of the following keys, in the order presented, into an empty tree.

   50, 75, 60, 70, 55, 56, 57

[additional space for answering the earlier question]

3. (10 pts.) Repeat Question 2, but use a *red-black tree* with *bottom-up* insertions. As in class, use boxes for black nodes and circles for red nodes.

[additional space for answering the earlier question]

4. (10 pts.) Repeat Question 3, but use *top-down* insertions.

[additional space for answering the earlier question]

5. (10 pts.) Depict the tree resulting from the *top-down deletion* of 75 and 70 (in that order) from the final tree of Question 4.

6. (10 pts.) This question is based on the *range maxima* problem defined in Saxena's paper.[1] Provide Java code fragments, or pseudocode at a similar level of detail, to answer range maxima queries in $O(1)$ time (worst case). Clearly separate the preprocessing actions from the actions at query time.

   What is the worst case time required for preprocessing? What is the worst case space used during preprocessing? Express these answers using asymptotic notation as a function of the input size n.

```
preprocess:
for(int i = 0; i < n; i++) {
    for(int j = i; j < n; j++) {
        if(i==j) P[i][j] = A[i];
        else if(A[j] > P[i][j-1]) P[i][j] = A[j];
        else P[i][j] = P[i][j-1];
    }
}

query(a,b):
return ( a <= b ? P[a][b] : P[b][a] );
```

[1] Sanjeev Saxena, "Dominance made simple," *Information Processing Letters* 109/9 (2009).

[additional space for answering the earlier question]

7. (10 pts.) Using the "(n, l, r)"-style notation for binary trees from class and Homework 1, provide concise functional specifications for each of the four cases of AVL rotation applied to a tree rooted at n.