You will submit your solution to this assignment to the Curator System (as `HW1`). Your solution must be either a plain text file (e.g., NotePad) or a typed MS Word document; submissions in other formats will not be graded.

Partial credit will only be given if you show relevant work.

---

1. [30 points] Write an algorithm to perform `int BST.RangeSearch(lower, upper)` as efficiently as possible. It should print all values in the binary search tree that are in the range [lower, upper] inclusively, and return the number of such values found. You may use Java syntax to express your solution, and you are encouraged to test your solution.

```
int BST.RangeSearchHelper(node, lower, upper)
   // base case:
   if (node == null)  return 0;
   // recursive case:
   int n = 0;
   if (node.value > lower)    // prune left branch if too small
      n += RangeSearchHelper(node.left, lower, upper);
   if (lower <= node.value <= upper)
      print node.value;
      n++;
   if (node.value <= upper)   // prune right brach if too large
      n += RangeSearchHelper(node.right, lower, upper);
   return n;
```

2. [30 points] Prove the following by induction: For every k >= 0, there are no more than 2^k nodes in level k of a binary tree.

Let B be a binary tree.
Let n(k,B) = number of nodes at level k in binary tree B. We are trying to prove that n(k,B) <= 2k.
Induct on k
Base case: k = 0 (assuming level 0 is the root level)

A binary tree at level 0 contains only the root node.
Thus, $n(k,B) = n(0,B) = 1 = 2^0 <= 2^k$.
[10 points – base case]

Induction step:

Assume theorem is true for all i where i < k; now prove its true for k.
By the induction hypotheses: $n(k-1,B) <= 2^{k-1}$
Each node at level k-1 has at most 2 children in level k. There are no other nodes at level k besides these.
Thus, $n(k,B) <= 2 * n(k-1,B) <= 2 * 2^{k-1} = 2^k$

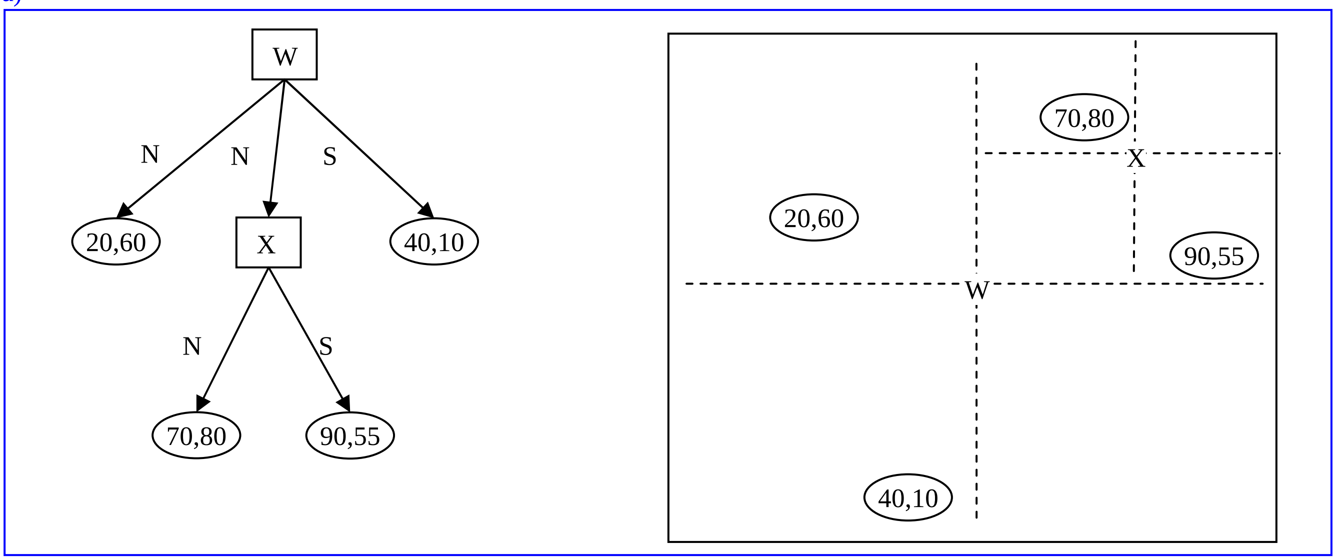
3. [40 points] Show the quadtrees (and their corresponding world maps) that would result from inserting the values given below into:

a) a region quadtree that uses key space splitting to always split regions into equally sized quadrants

b) a point quadtree that uses key value splitting to always split regions at the inserted values.

See sections 2.1 and 3.1 of the paper by Samet for discussions of region and point quadtrees. Use bucket size = 1. World coordinates are (0,0) – (100,100). Insert values in this order: A(20,60), B(70,80), C(40,10), D(90,55).

a)



b)