1. Draw the binary tree whose inorder traversal is 12345678 and whose preorder traversal is 43128657. Is it a binary search tree?

2. Which permutations of \{1, 2, 3, 4\} may be a sequence of nodes examined during a search in a binary search tree? (E.g. 3,4,2,1 is one of the excluded sequences. See a similar exercise 12.2-1, page 293.) For extra points you can derive a general formula for the number of such permutations of \{1, \ldots, n\}.

3. Devise an algorithm that takes two values \(a\) and \(b\) such that \(a \leq b\) and which prints all keys \(x\) of a binary search tree such that \(a \leq x \leq b\). The running time of your algorithm should be \(O(m + h)\), where \(m\) is the number of keys printed and \(h\) is the height of the tree.

   Write the algorithm in a pseudocode and argue its time complexity.

4. How many different permutations of the values 1 through 10, when inserted in that order, will yield the following tree?

   ![Binary Tree Diagram]

   Hint: Derive a recurrent formula.