CS 121 Final Examination Review

1. **Programming with Recursive**

   Recursive function general case and base case; For the linked list, show the contents of the run-time stack when Insert(listNode, x) is executed, the content of listNode->info = y; Recursive statement; if statement in recursive function.

2. **Binary Search Trees**

   Given a binary search tree and traversal node sequences, try to determine the order; Building a binary search tree based on given elements, determine one element's successor; Given a tree, trace the run-time stack content with related function such as R1 = Insert(tree>left) and R2 = Insert(tree->right); Draw a binary search tree with elements, determine the level number; Child location of an array based binary tree.

3. **Heaps, Priority Queues, and Heap Sort**

   After steps of applying the heap sort algorithm to an unsorted array with given elements, what can we find the array change(s)? Why heap sort does not use an object of type HeapType? The Big-O complexity of the Dequeue operation on a priority queue which is being implemented using a linked list of values sorted by priority.

4. **Tree Plus**

   How to justify a tree is balanced? The AVL trees' performance (complexity). How to insert an element into a B-Tree?

5. **Sets, Maps, and Hashing**

   Disruption operation(s) of a balanced binary search tree. Chain used in a hash. Binary search tree implementation of a map.

6. **Graphs**

   What are the components of a graph? Write down the graph nodes in breadth-first order.