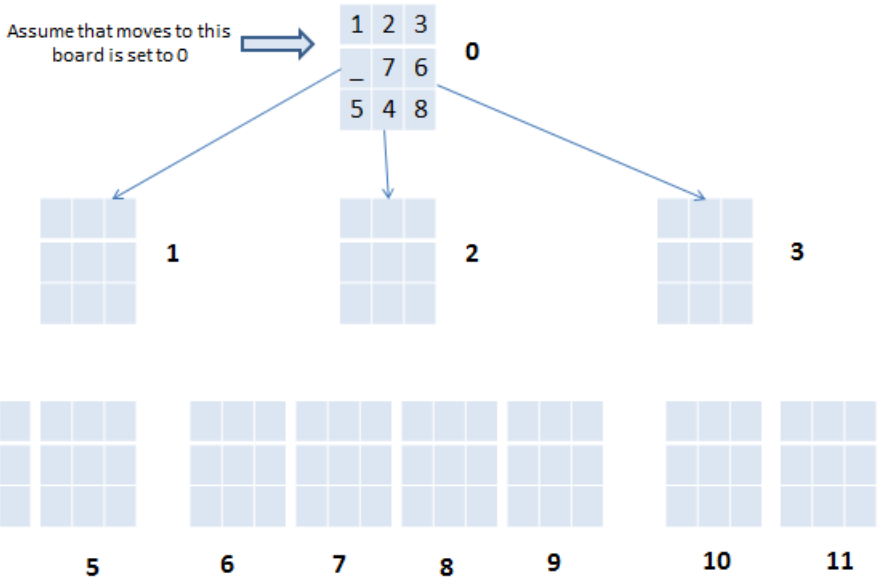
COS 226 Data Structures and Algorithms
Computer Science Department
Princeton University
Spring 2016

Week 4 Activity

1. **8-Puzzle**
   The 8-puzzle problem is a puzzle played on a 3-by-3 grid with 8 square tiles labelled 1 through 8 and a blank square. Your goal is to rearrange the tiles so that they are in order, using as few moves as possible. You are permitted to slide tiles horizontally or vertically into the blank square. Given below is a state of an 8-puzzle algorithm.

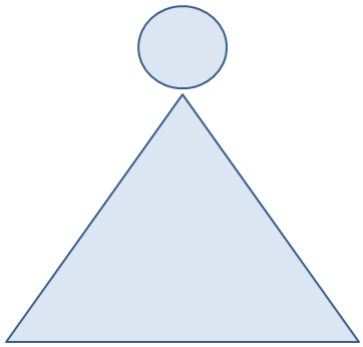
   (a) Complete the unfilled boards in the following partial game tree for 8-puzzle.



   (b) The *Manhattan priority function* is defined as the sum of the Manhattan distances (sum of the vertical and horizontal distances) from the tiles to their goal positions, plus the number of moves made so far to get to the search node. Compute Manhattan distance(M) and priority(P) for each board and Write M and P next to each board in the tree shown in(a).

   (c) Circle the boards that will not be considered further (i.e the critical optimization).

(d) Boards are numbered in the game tree starting with 0 (initial board) and show the order in which boards are inserted (and deleted) into and out of priority queue (PQ) during the execution of the A* algorithm. Note that not all boards are inserted to the PQ at the same time. When two boards have the same priority, pick the one with the smallest board index. You only need to show the board with the minimum priority (in the circle) and place the other board indices in any order inside the triangular part.



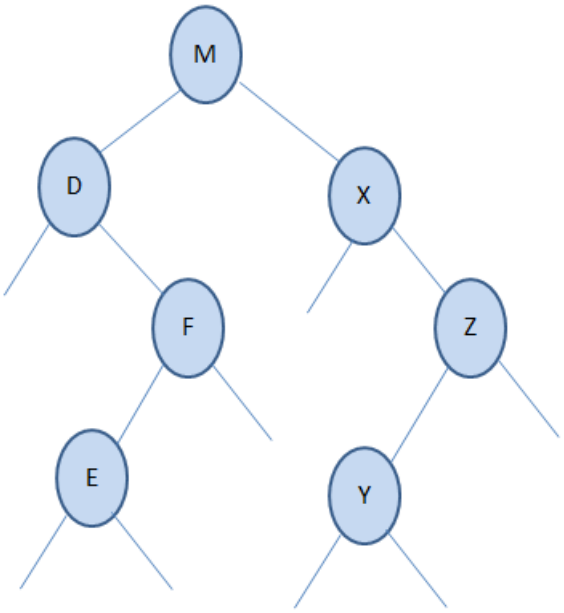
(e) We note that not all 8-puzzle boards are solvable. Show that the following puzzle is unsolvable. What is the order of growth of your algorithm?

1	_	3
2	5	4
7	6	8

2. **Binary Search Trees**

(a) Draw all possible binary search trees that can be generated using the comparable keys A,B and C. Use the alphabetic order for ASCII characters.

(b) An inorder traversal of a binary tree is obtained by recursively visiting its left subtree, processing the root and recursively visiting the right subtree. A preorder traversal of a binary tree is obtained by processing the root, and then recursively visiting left and right subtrees. Write down the order of the keys visited in the following BST, during an inorder traversal? A preorder traversal?.



inorder:_____

preorder:_____

(c) Draw a Binary Tree (not a BST) that produces the following order traversals.
preorder traversal: $ADBCPMNTQ$
inorder traversal: $DCBANMTPQ$.

3. **Priority Queues (Bonus Problem)**

Suppose we have the following code that uses a binary heap-based MaxPQ. Assume $N > k$.

```
int k = 10;
MaxPQ<Integer> pq = new MaxPQ<Integer>();
int N = a.length;

for (int i = 0; i < N; i++) {
   pq.insert(a[i]);
   if (pq.size() > k) pq.delMax(); /* MARK */
}

for (int i = 0; i < k; i++)
   System.out.println(pq.delMax());
```

(a) What does this code output?

(b) What is the run time of the code in terms of N and k?

(c) Suppose we remove the entire line marked with /* MARK */. What does the code output?

(d) What is the run time of the code in terms of N and k?