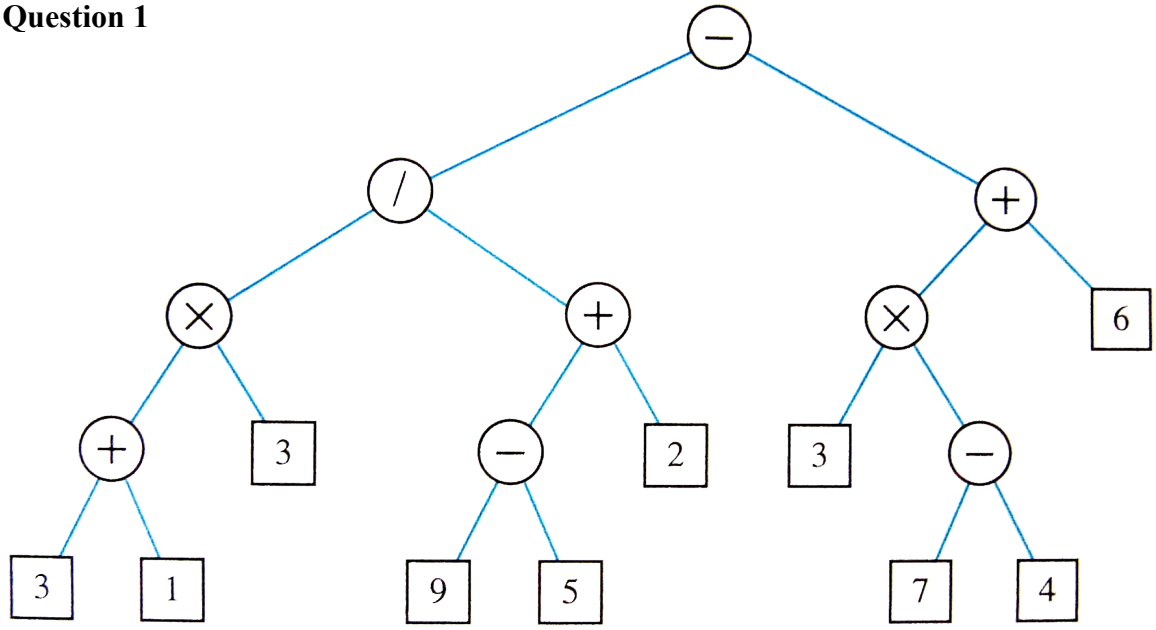
COMP3506/COMP7505—Algorithms and Data Structures

School of Information Technology and Electrical Engineering

Week 6 Tutorial

Question 1



a) Is this a proper binary tree?

b) What is the depth of the root node?

c) What is the depth of the node with a value of '6'?

d) What is the height of the node with a value of '/'?

e) What is the height of the tree?

f) If a post-order traversal of the tree is performed, what is the order of nodes visited?

g) What is the level of the nodes containing the *multiply* operators.

Question 2

Refer to the tree from question 1.

a) If an array representation were used to store the binary tree, how large would the array need to be?

b) Show the contents of the tree stored in an array i.e. ['a']['b']['c']…

Question 3

Show the contents of the underlying heap (as a binary tree) after each of the following operations is performed on a heap-based Priority Queue:

1. insert(5, A)
2. insert(4, B)
3. insert(7, I)
4. insert(1, D)
5. removeMin()
6. insert(3,J)
7. insert(6,L)
8. removeMin()
9. removeMin()
10. insert(8,G)
11. removeMin()
12. insert(2,H)
13. removeMin()
14. removeMin()

Question 4

At which nodes of a heap can an entry with the largest key (lowest priority) be stored?

Question 5

Let T be a complete binary tree such that node v stores the entry $(p(v),0)$, where $p(v)$ is the level number of v. Is tree T a heap? Why or why not?

Question 6

Explain why the case where node r has a right child but not a left child was not considered in the description of down-heap bubbling.

Question 7

a) Show how to implement the Stack ADT using only a priority queue and one additional integer instance variable.

b) Show how to implement the (standard) Queue ADT using only a priority queue and one additional integer instance variable.