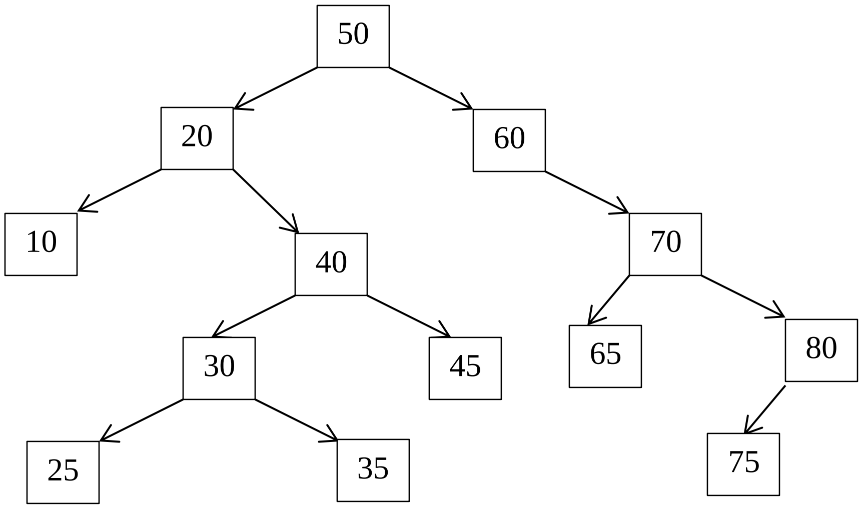
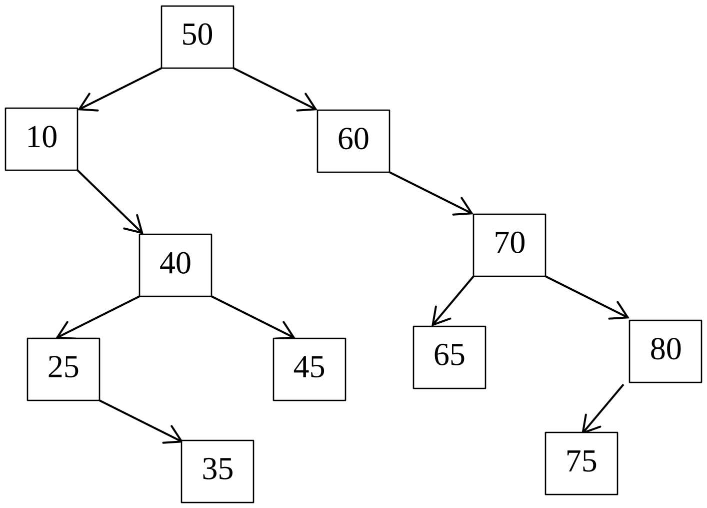
Nathan Tung
Com Sci 32 (Lec 2)
March 13, 2012

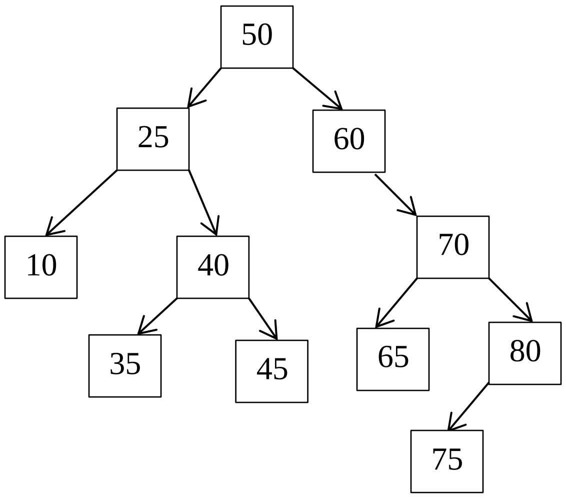
CS32 Homework 5

1a. Using the simplest binary search tree insertion algorithm, show the tree that results after inserting into the above tree the nodes 80, 65, 75, 45, 35, and 25 in that order.



1b. After inserting the nodes mentioned in part a, what is the resulting BST after you delete the node 30, then the node 20?





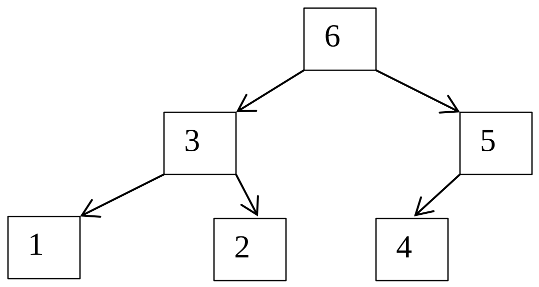
(Either one works depending on which subtree the replacement comes from.)

1c. After inserting the nodes mentioned in part a, what would be printed out by in-order, pre-order, and post-order traversals of the tree?

In-order Traversal:	10 20 25 30 35 40 45 50 60 65 70 75 80
Pre-order Traversal:	50 20 10 40 30 25 35 45 60 70 65 80 75
Post-order Traversal:	10 25 35 30 45 40 20 65 75 80 70 60 50

Nathan Tung
Com Sci 32 (Lec 2)
March 13, 2012

2a. Show the resulting heap.

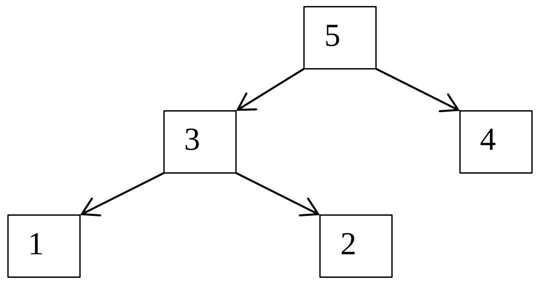


2b. Show how your heap from part a would be represented in an array.

6	3	5	1	2	4

(Front) (Back)

2c. Remove the top item from the heap and show the resulting array after the removal operation.



5	3	4	1	2

(Front) (Back)

3a. Show a C++ structure/class definition for a binary tree node that has both child node pointers and a parent node pointer. Assume the data stored in each node is an int.

```
struct Node
{
      Node* parent;
      Node* left; //child
      Node* right; //child
      int data;
};
```

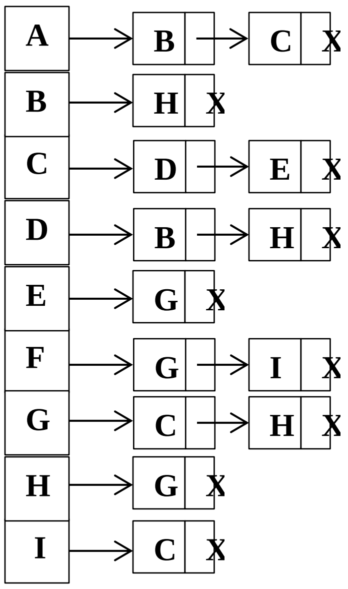
3b. Write pseudocode to insert a new node into a binary search tree with parent pointers.

```
insert(Node* node, Node* root)
      if root is NULL, then set node to the root and node's parent pointer to NULL
      else if node's value is smaller than root's value
            if root has no left child
                  set root's left pointer to node and node's parent pointer to root
            else pass in the root's left child as root in another call to insert
      else if node's value is greater than root's value
            if root has no right child
                  set root's right pointer to node and node's parent pointer to root
            else pass in the root's right child as root in another call to insert
```

Nathan Tung
Com Sci 32 (Lec 2)
March 13, 2012

4a. Show an adjacency matrix and an adjacency list for the following graph.

	A	B	C	D	E	F	G	H	I
	0	1	2	3	4	5	6	7	8
0	0	1	1	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1	0
2	0	0	0	1	1	0	0	0	0
3	0	1	0	0	0	0	0	1	0
4	0	0	0	0	0	0	1	0	0
5	0	0	0	0	0	0	1	0	1
6	0	0	1	0	0	0	0	1	0
7	0	0	0	0	0	0	1	0	0
8	0	0	1	0	0	0	0	0	0



4b. If you perform a depth-first traversal through this graph starting from vertex E, what vertices are visited, and in what order? (List all the answers.)

EGCDBH
EGCDHB
EGHCDB

One comprehensive traversal (including all steps) where the path is chosen alphabetically is as follows. Note that this is not the answer to 4b, but one such process of calculating the answer.

E
EG
EGC
EGCD
EGCDB
EGCDBH
EGCDB
EGCD
EGCDH
EGCD
EGC
EGH
EG
E