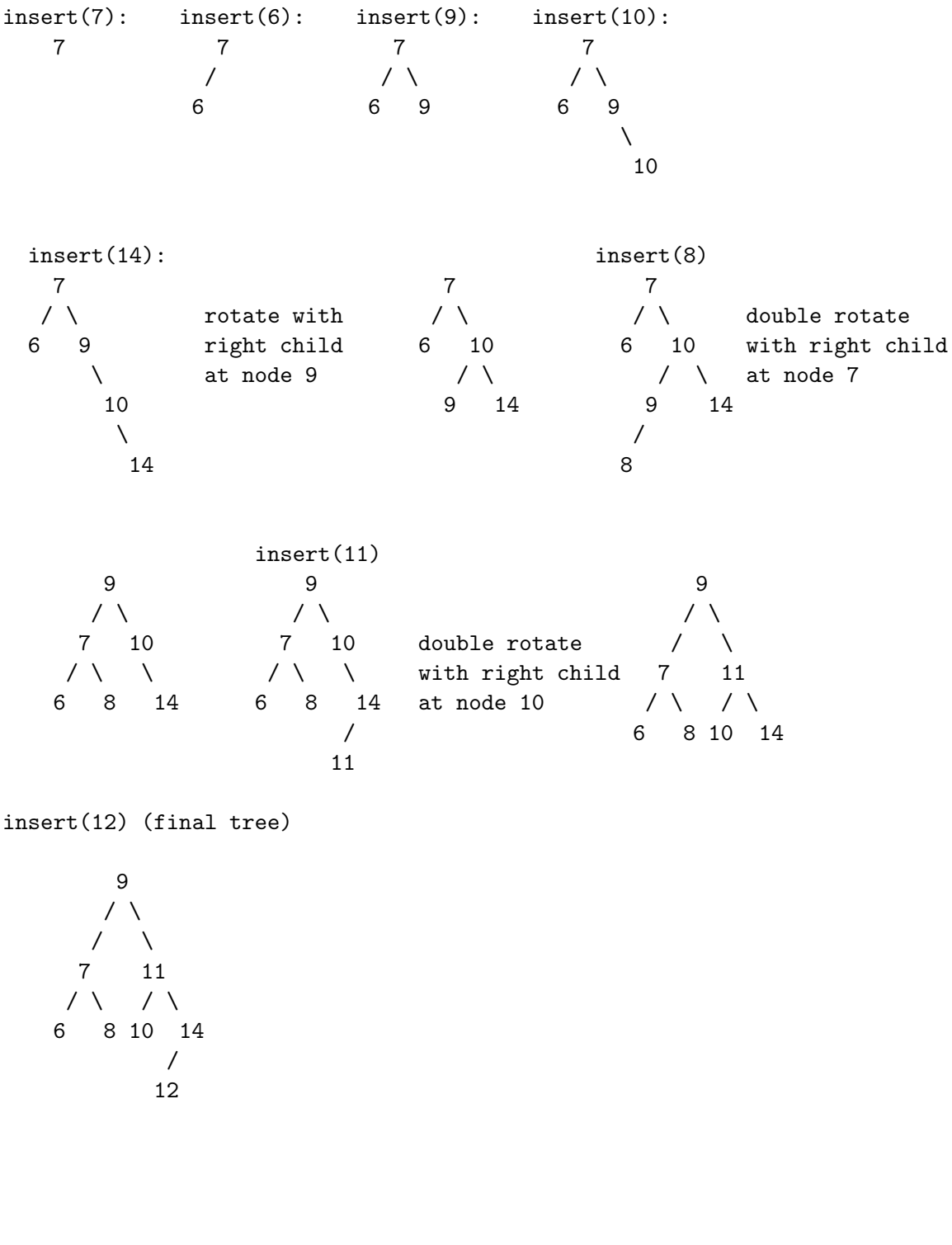
CSE 250 Spring 2011
Homework 5 Solutions
Total Points: 39

1. (5 points) Insert the key values 7, 6, 9, 10, 14, 8, 11, 12 (in this order) into an initially empty AVL tree. Show the resulting tree after each insert operation.

Solution:

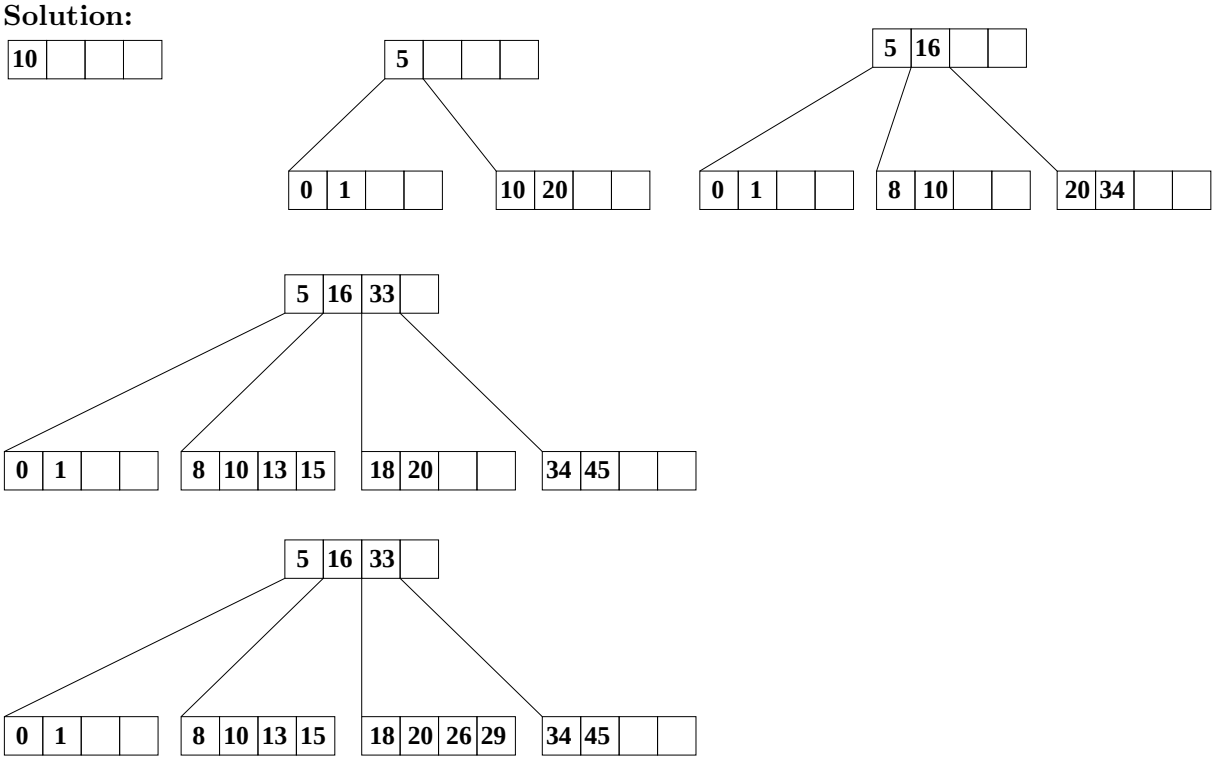


2. (6 points) Consider a B-tree with CAP = 5. (Namely, each node has at most 5 children, and each node contains between 2 and 4 key values). Insert the following sequence of key values:

10 20 0 1 5 8 34 16 18 13 15 33 45 26 34 29

into an initially empty B-tree. Show the resulting B-tree each time a new node is created.

Solution:

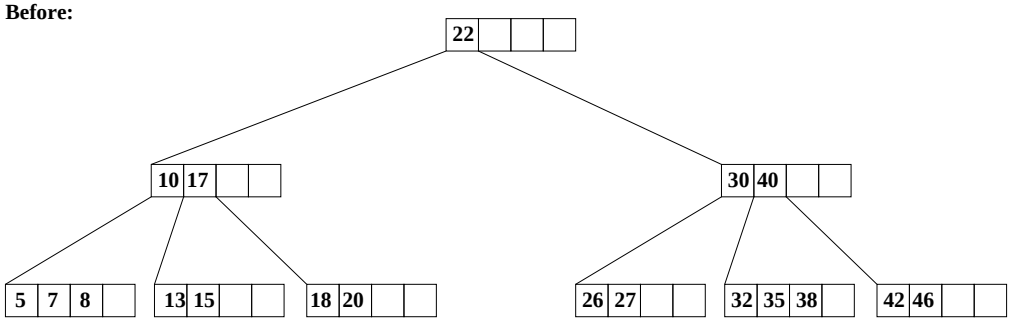


*Note: 34 is not inserted the second time when it is found to already be in the tree.

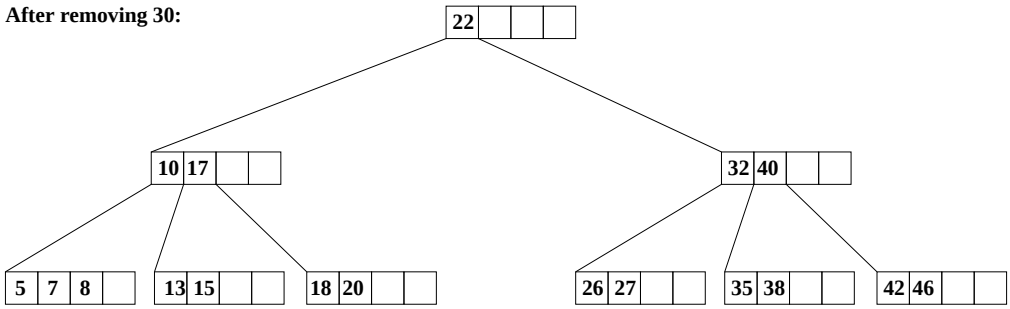
3. (4 points) Remove keys 30, 26, 15 and 17 from the B-tree shown in Fig 11.58. Show the resulting B-tree after each remove operation.

Solution:

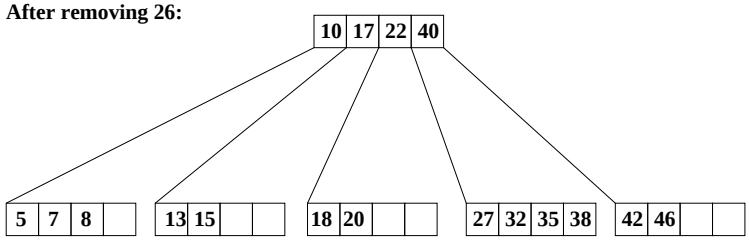
Before:



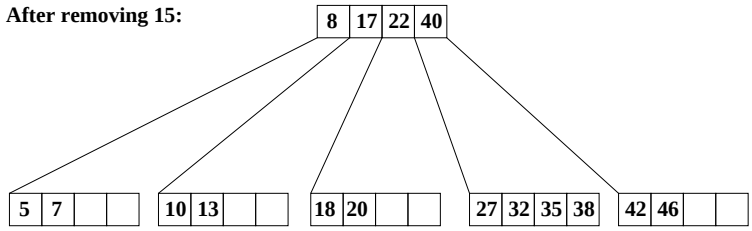
After removing 30:



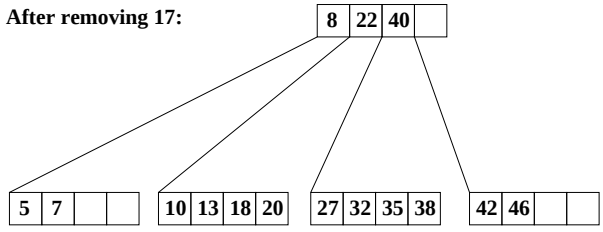
After removing 26:



After removing 15:



After removing 17:



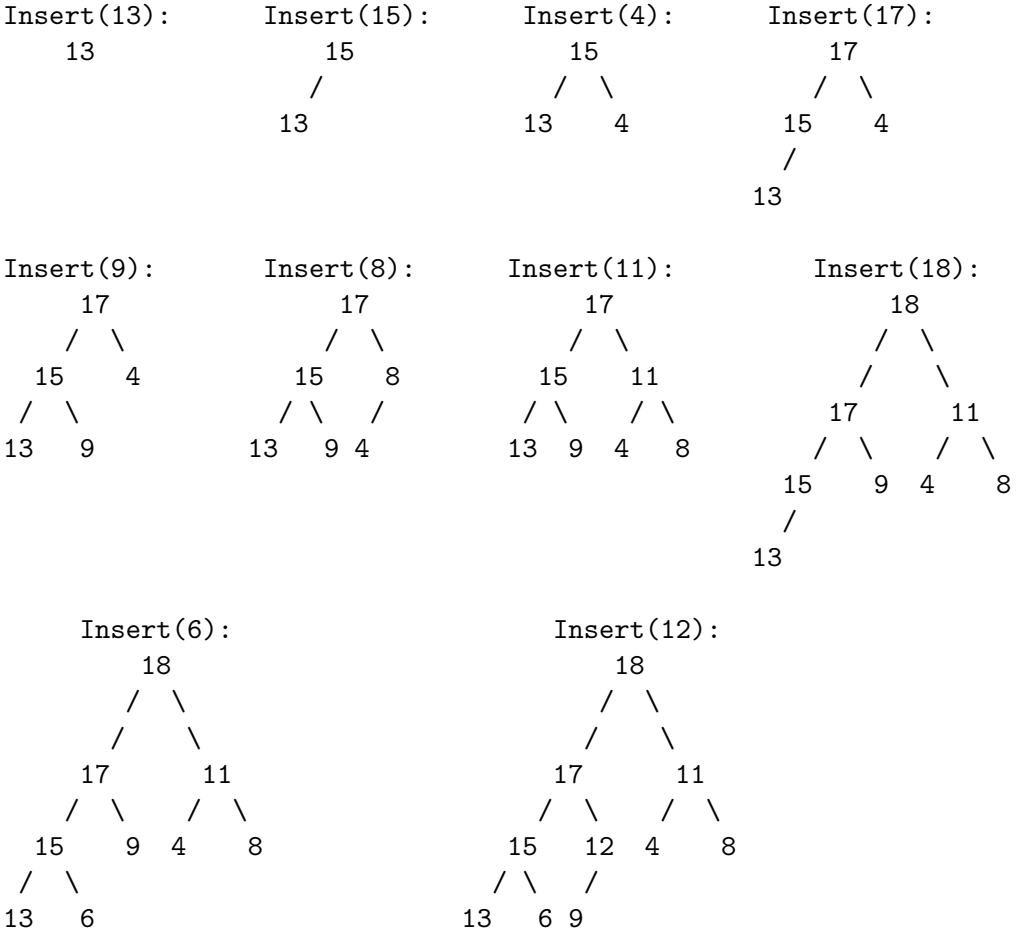
For the problems 4, 5, 6, draw the binary heap of the results, (not the array).

4. (6 points) Insert the sequence of keys:

   13, 15, 4, 17, 9, 8, 11, 18, 6, 12

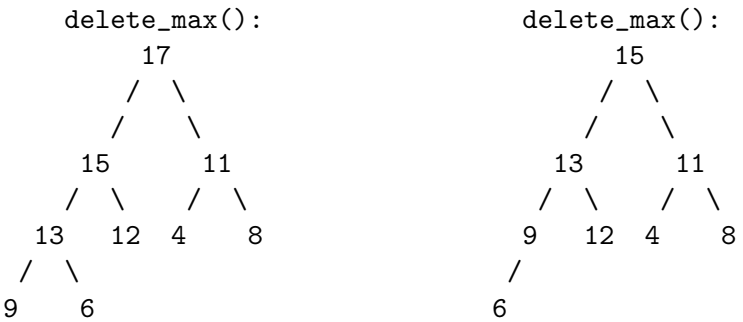
   into an initially empty (max) heap. Show the result after each insert operation.

   Solution:



5. (4 points) Perform the `delete_max()` operation twice to the heap in the previous problem. Show the result after each operation.

   Solution:



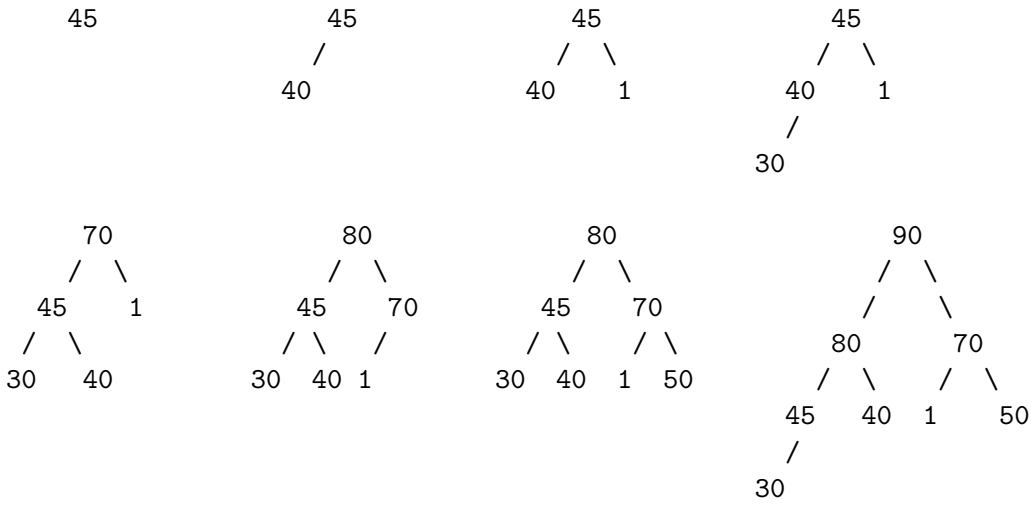
6. (4+4 = 8 points) Perform HeapSort on the following input array:

45, 40, 1, 30, 70, 80, 50, 90, 60, 10

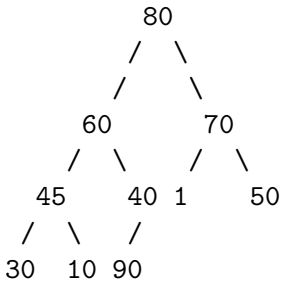
(a) Show, step-by-step, the result of `build_heap` function of the above array.

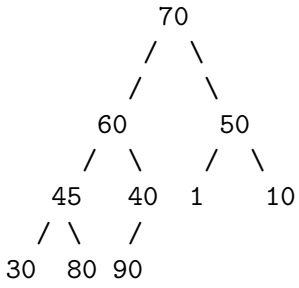
(b) Show, step-by-step, the result of `shrink_heap` function of the array.

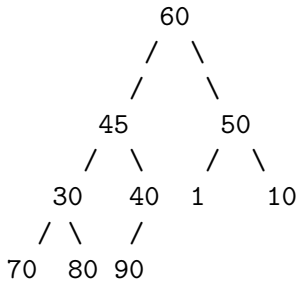
Solution:
(a)

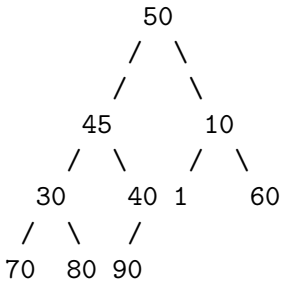


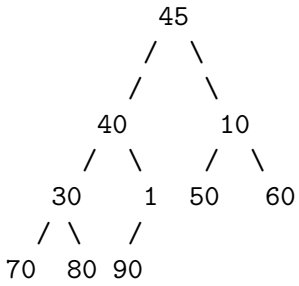
(b)

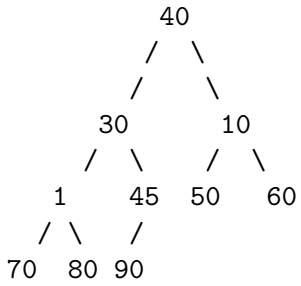

80
60 70
45 40 1 50
30 10 90

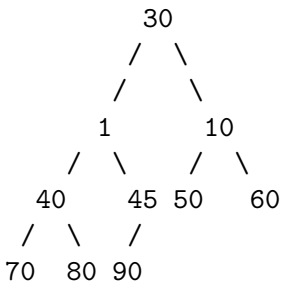

70
60 50
45 40 1 10
30 80 90

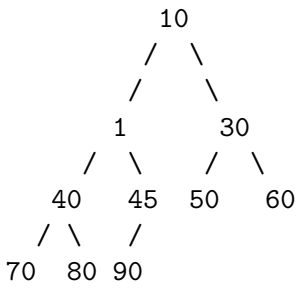

60
45 50
30 40 1 10
70 80 90

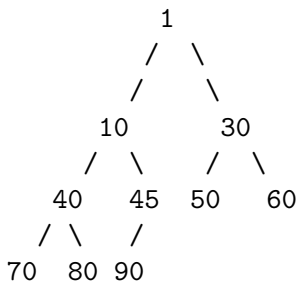

50
45 10
30 40 1 60
70 80 90


45
40 10
30 1 50 60
70 80 90


40
30 10
1 45 50 60
70 80 90


30
1 10
40 45 50 60
70 80 90


10
1 30
40 45 50 60
70 80 90


1
10 30
40 45 50 60
70 80 90

7. (2+2+2=6 points). Given the input:

371, 323, 173, 199, 344, 679, 989

and the hash function: `h(x) = x (mod 10)`, show the resulting hash table by using:

(a) Separate chaining hash table

(b) Open addressing hash table using linear probing

(c) Open addressing hash table using quadratic probing

Solution:

x	h(x)
371	1
323	3
173	3
199	9
344	4
679	9
989	9

(a)

Key	Value
0	null
1	pointer to list: {371}
2	null
3	pointer to list: {323, 173}
4	pointer to list: {344}
5	null
6	null
7	null
8	null
9	pointer to list: {199, 679, 989}

(b)

Key	Value
0	679
1	371
2	989
3	323
4	173
5	344
6	null
7	null
8	null
9	199

(c)

Key	Value
0	679
1	371
2	null
3	323
4	173
5	344
6	null
7	null
8	989
9	199