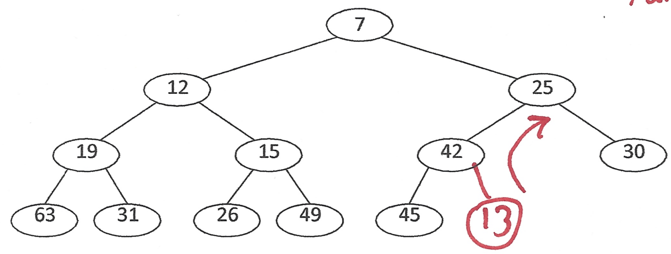
Phillip Escandon

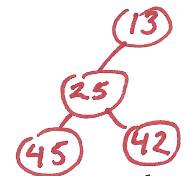
Part 2. Consider the following heap that implements a minimum-oriented priority queue:

Parent < child

13 swaps with 42

13 swaps with 25

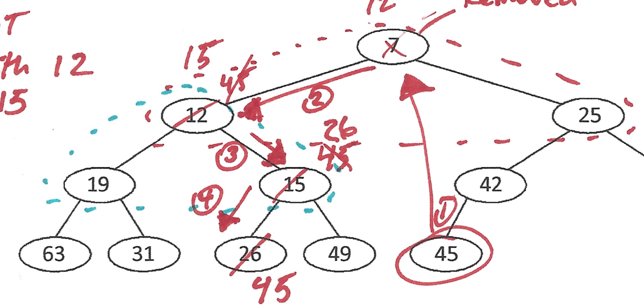


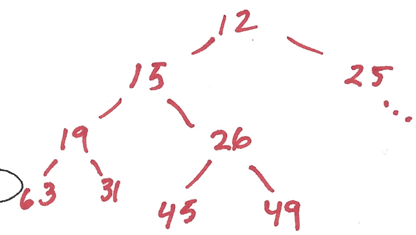


Suppose you insert 13 (or an entry with key = 13). Show the resulting heap. You must use the method we discussed in the class.

Part 3. Consider the following heap that implements a minimum-oriented priority queue:

1. 45 To Root
2. 45 swap with 12
3. 45 ⇄ ~~26~~ 15
4. 45 ⇄ 26





Suppose you perform the removeMin( ) operation. Show the resulting heap. You must use the method we discussed in the class.

14 mod 13
= 1

Part 4. Consider the following hash table, which contains some keys.

14

0	1	2	3	4	5	6	7	8	9	10	11	12
	79	14		17	98		59		22		37	

If you insert $k = 14$ to this hash table, where is it placed? Assume that collision is resolved using the open addressing method with linear probing.

Part 5. Suppose that your hash function resolves collisions using the open addressing method with double hashing. The double hashing method uses two hash functions h and h'.

Assume that the table size $N = 13$, $h(k) = k \bmod 13$, $h'(k) = 1 + (k \bmod 11)$, and the current content of the hash table is:

0	1	2	3	4	5	6	7	8	9	10	11	12
14	79			17	98		59		22		37	

14

h(K) = K mod 13

h'(K) = 1 + K mod 11

If you insert $k = 14$ to this hash table, where is it placed?

h(K) = 14 mod 13
= 1

h'(K) = 1 + 14 mod 11
= 1 + 3
= 4