HW4 - hd2366 - Written

1) Weiss 5.1

$\{4371, 1323, 6173, 4199, 4344, 9679, 1989\}$

$h(x) = x \bmod 10$

a) Separate chaining hash table

$4371 \% 10 = 1$

$1323 \% 10 = 3$

0	
1	4371
2	
3	1323 → 6173
4	4344
5	
6	
7	
8	
9	4199 → 9679 → 1989

b) Linear Probing Hash Table : $h(x) = (hash(x) + f(i)) \% tableSize$

0	9679
1	4371
2	1989
3	1323
4	6173
5	4344
6	
7	
8	
9	4199

{4371, 1323, 6173, 4199, 4344, 9679, 1989}.

c) Hash table using Qudrictic Probing : $h(x) = (hash(x) + f(i^2))$ % table size

0	9679
1	4371
2	
3	1323
4	6173
5	4344
6	
7	
8	1989
9	4199

- first collision: $(6173 + 1^2)$ % $10 = 4$.
- 1st collision: $(4344 + 1^2)$ % $10 = 5$
- 1st collision: $(9679 + 1^2)$ % $10 = 0$
- 1st collision $(1989 + 1^2)$ % $10 = 0$

2nd collision. $(1989 + 2^2)$ % $10 = 3$

3rd collision $(1989 + 3^2)$ % $10 = 8$

d) $h_2(x) = 7 - (x \bmod 7)$. $f(i) = i \cdot hash_2(x)$

$h_2(x) = (hash(x) + f(i))$ % table size

$7 - (6173$ % $7) = 7 - 6 = (1 + 6173)$ % $10 = 4$

$h_2() = 7 - (4344$ % $7) = 3$

$h_2() = 7 - (9679$ % $7) = 2$

7 - 4199

0	
1	4371
2	
3	1323
4	6173
5	9679
6	
7	4344
8	
9	4199

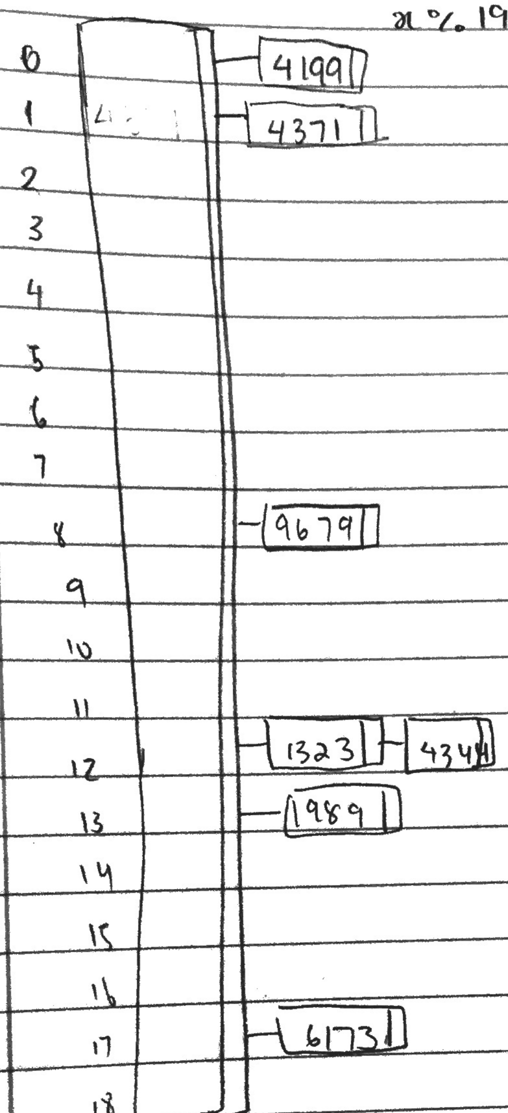
The # 1989 can't be entered in the table because positions 9, 5, 1, 7, 3 are filled.

$h_2(1989) = 6$

2) Weiss 5.2 - Rehash @ table size 19

$x \% 19$

a)



b) Show only non-null positions

0	4199
1	4371
8	9679
12	1323
13	1989
14	4344
17	6173

c)

0	4199
1	4371
8	9679
12	1323
13	1989
16	4344
17	6173

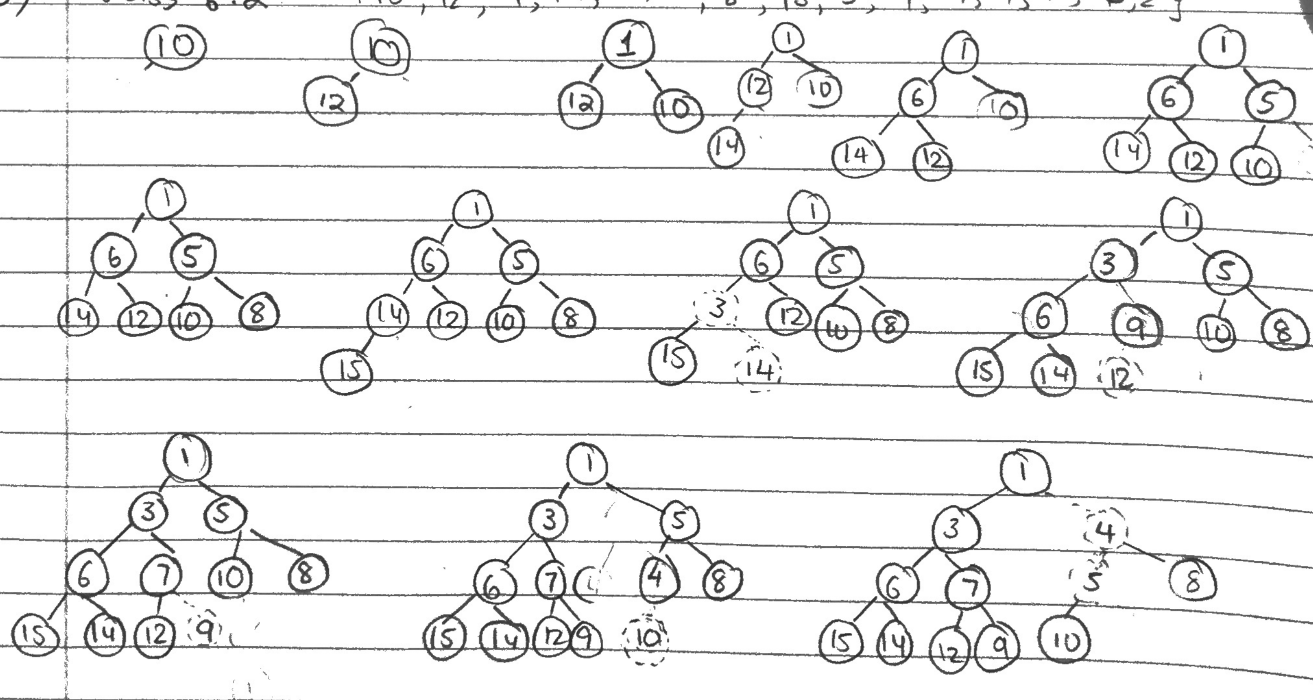
d)

0	4199
1	4371
8	9679
12	1323
13	1989
15	4344
17	6173

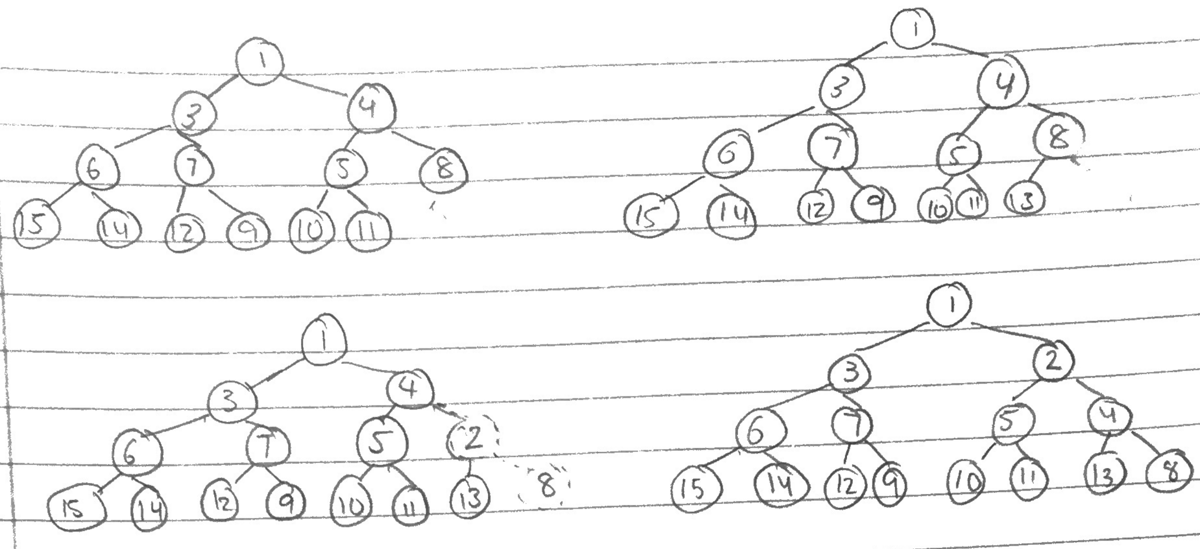
$h_2(4344) = 3$

$h_1(4344) = (12 * 3) \% 19 = 15$

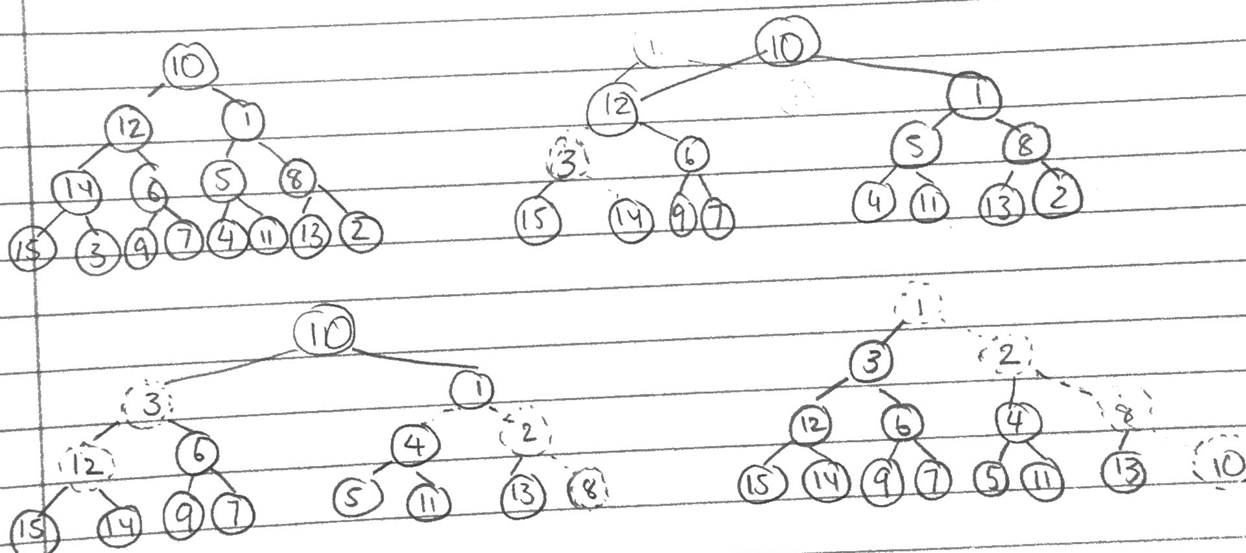
3) Weiss 6.2 {10, 12, 1, 14, 6, 5, 8, 15, 3, 9, 7, 4, 11, 13, 2}



=>



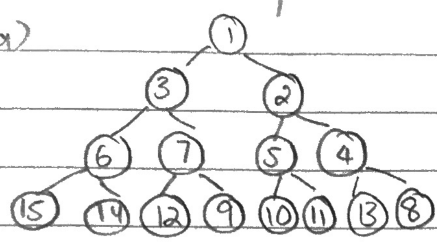
b) Result of using linear time algorithm to build binary heap :



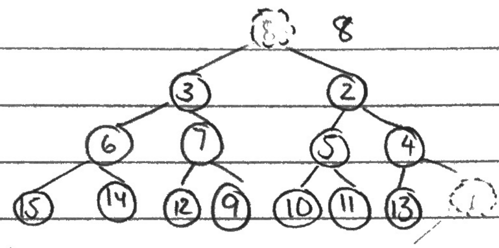
4) Weiss 6.3

3 delete min operations:

Heap a)



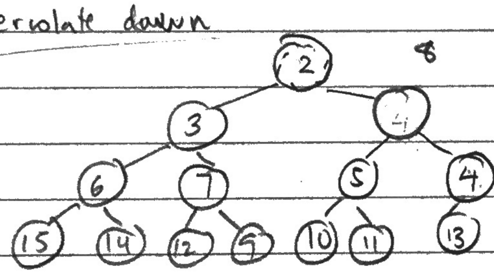
delete 1

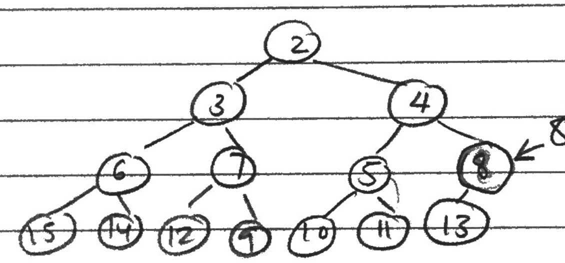


. Delete 1

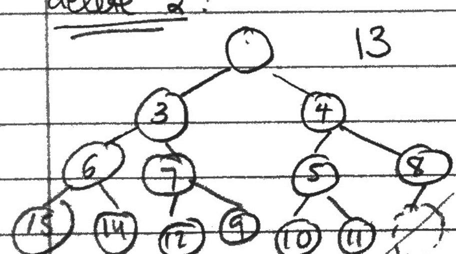
. Move 8 up.

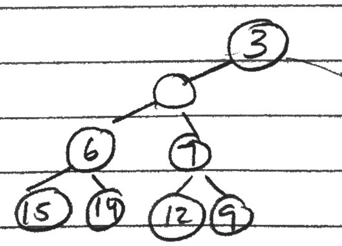
Percolate down





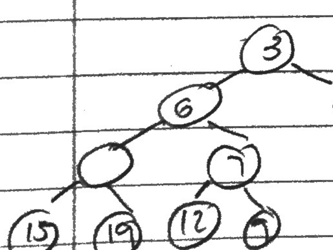
delete 2:



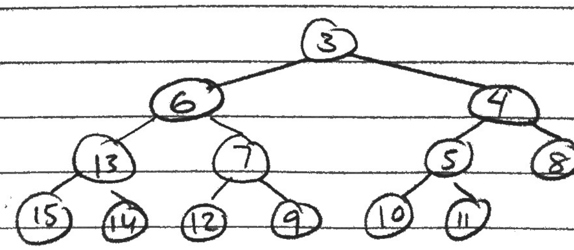


Same as previus

this side of the tree doesn't change

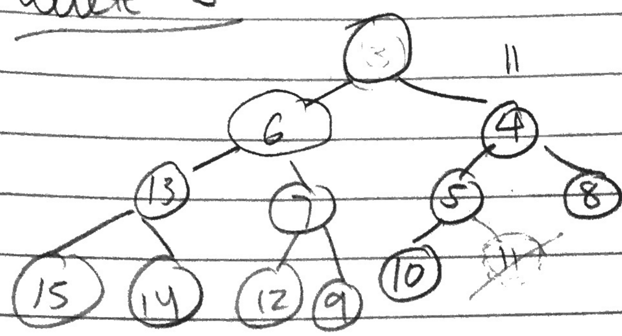


Same

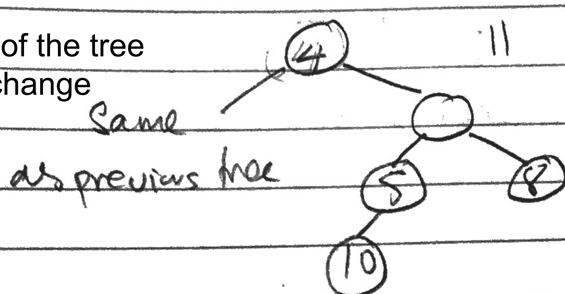


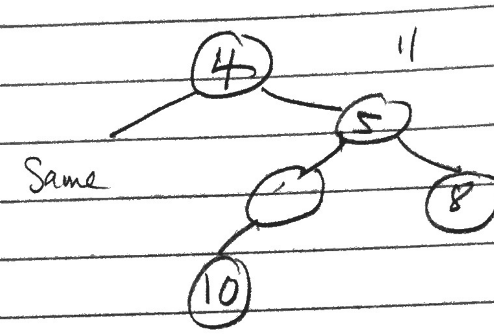
delete 3:

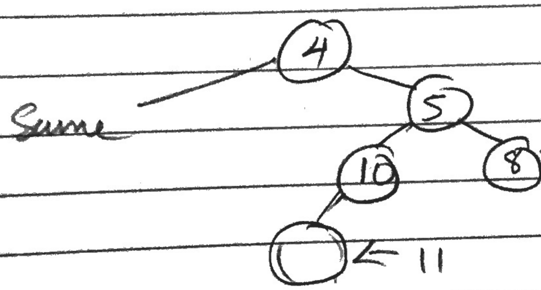
delete 3

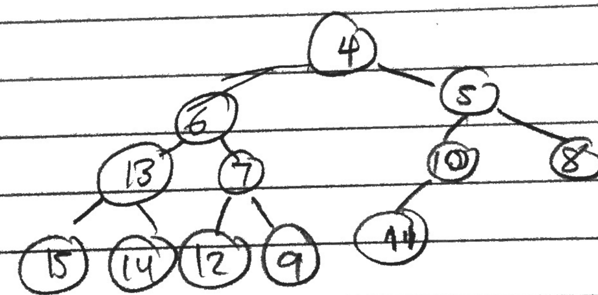


this side of the tree
doesn't change









5) Weiss 6.8.

a) maximum item of the heap is the leaf.

. Assume the max value is not a leaf (max value = m)

⇒ it has 1 child or 2 children

. Since m is the max value, it's larger than the children

⇒ ~~A~~ it violates the heap property and contradict the heap structure.

b) There must be exactly $\lceil N/2 \rceil$ leaves

- The heap property indicates it must be a comple binary tree.

which is full & the number of Nodes

$N = 2^k$

with 2^{k-1} leaves

$= 2^k - 2^{-1}$

$= N - 2^{-1} = N/2$ leaves.

c) Every leaves must be examined to find it (max value m)

. From part A, we know the max value of a heap must be a leaf

. Larger values than m can be inserted to be a leaf,

depending on the order of insertion, it could be placed at different location.

⇒ We need to search all leaves to find the max