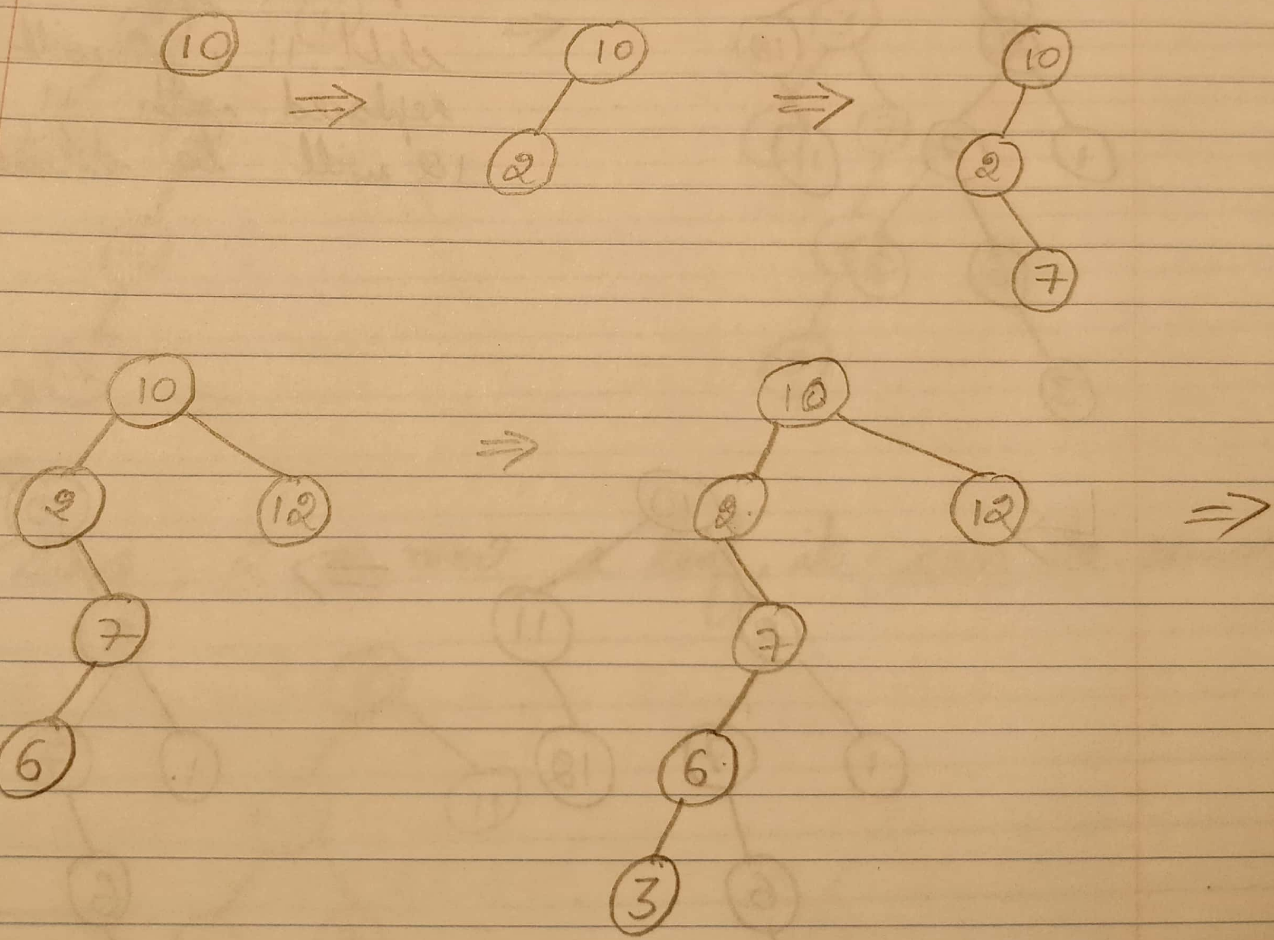
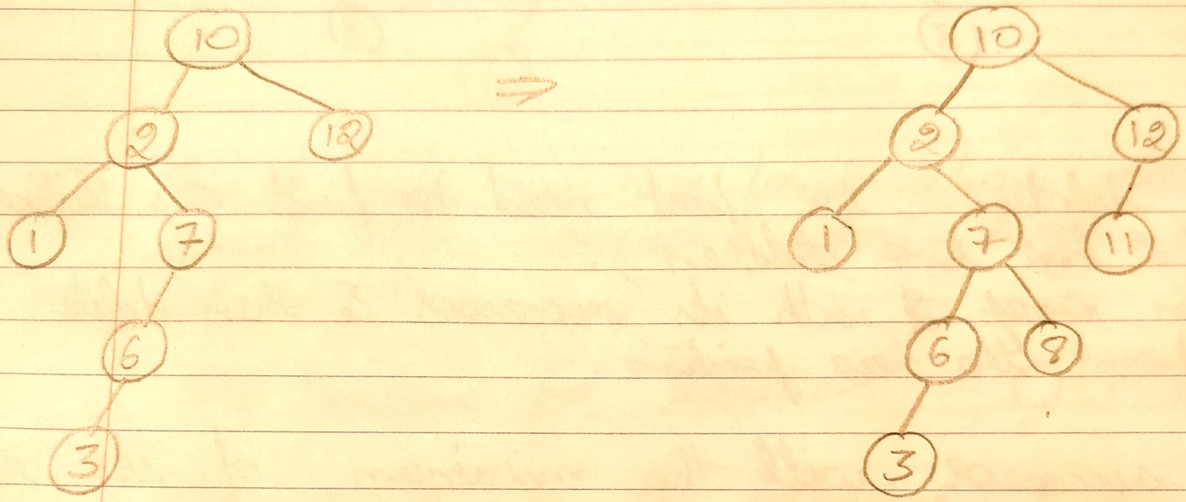
HOMEWORK 5 => ARSHITHA BASAVARAJ

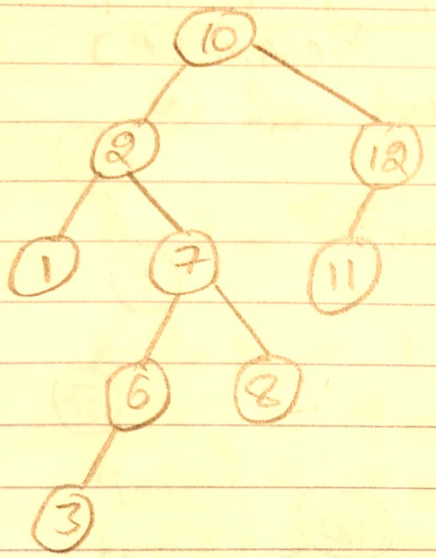
2) a)



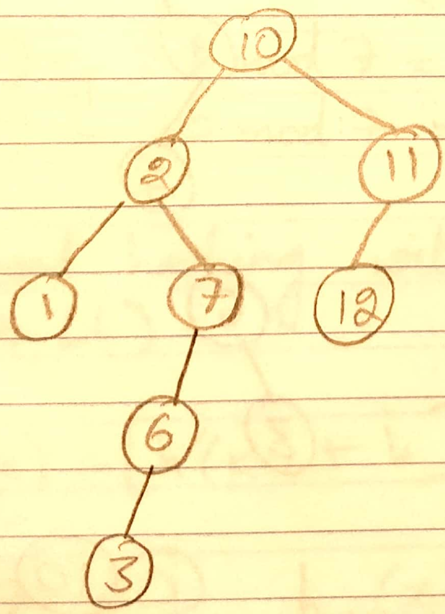


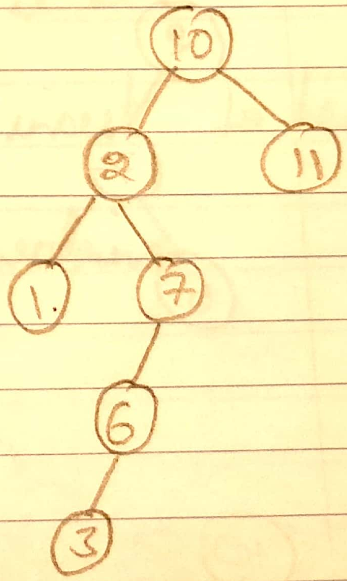
Binary Search Tree after insertion

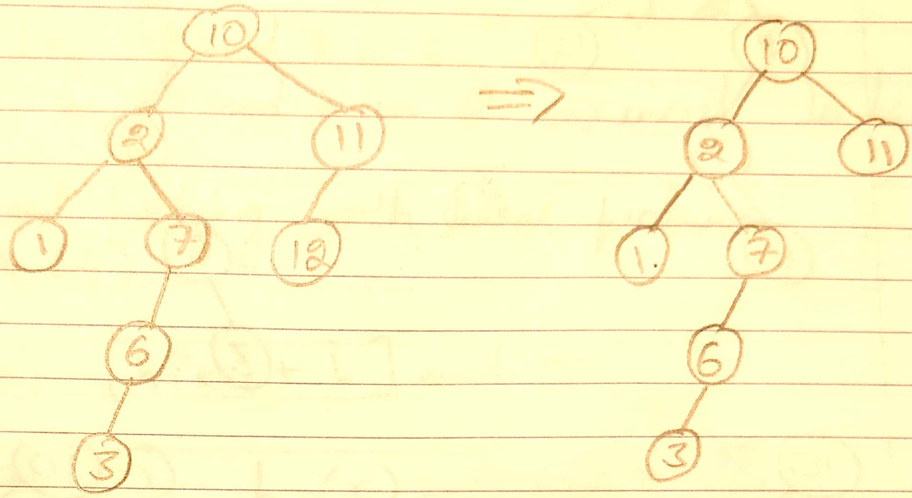
b)



∵ 12 has one child 11, 12 will be replaced with 11 & then 12 will be deleted

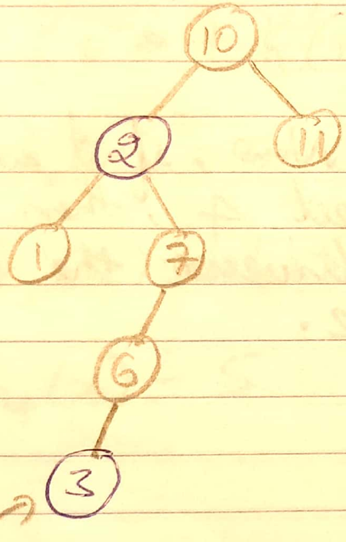




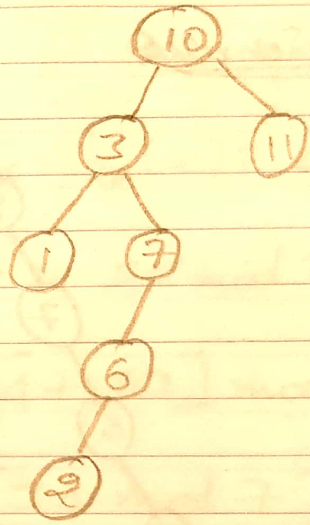
↳ 

c> To delete 2, we first need to find 2's successor (∵ 2 has two children).
Then swap 2 with its successor & then delete 2 from the new position.

2's successor will the minimum of its right subtree.

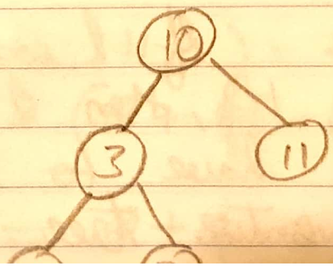


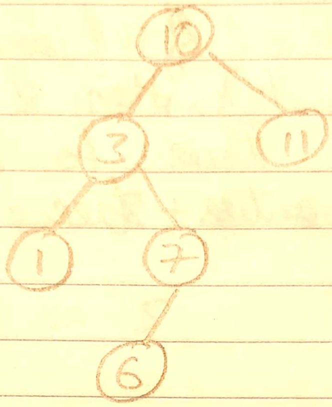
⇒



successor of 3

Since, 2 is now a leaf, it can be removed.



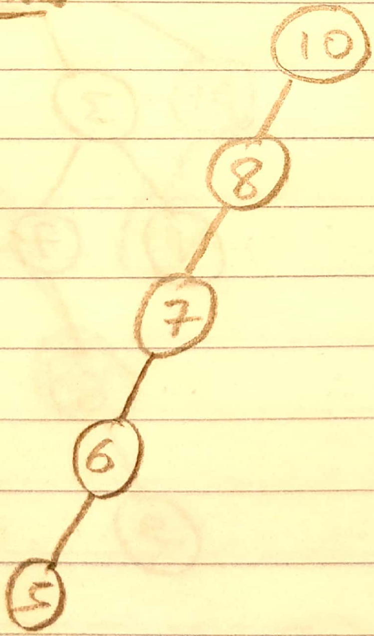


3> Worst-case run time for insertion into Binary Search Tree occurs when height of the tree = n.

$\Rightarrow$ Runtime for insertion $= O(n)$ (worst-case)

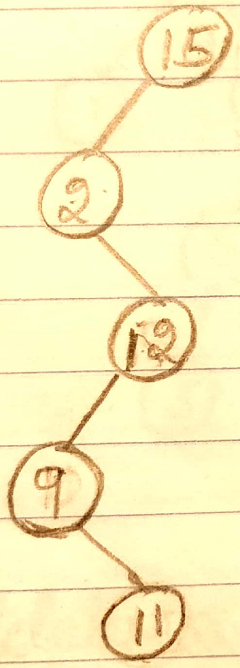
This occurs in the following two scenarios:

1st Scenario



Now, if I wish to insert 4, then I to traverse the entire tree.

2nd scenario



If I want to insert 10, then I'll have to traverse the entire tree.

a) 4) 7, 9, 18, 51, 33, 30

$h_1(x) = x \bmod 7$

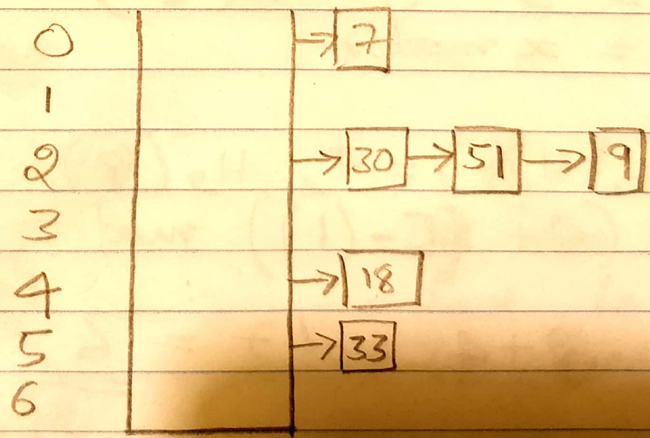
$h(7) = 7 \bmod 7 = 0$; $h(9) = 9 \bmod 7 = 2$

$h(18) = 18 \bmod 7 = 4$; $h(51) = 51 \bmod 7 = 2$

$\Uparrow$ insert to head

$h(33) = 33 \bmod 7 = 5$;

$h(30) = 30 \bmod 7 = 2 \Rightarrow$ insert to head



b) Closed hashing with linear probing.

$H(x) = (h(x) + i) \bmod 7$ $\quad i$ = no. of collision

$H(7) = 0;\ H(9) = 2;\ H(18) = 4$

$H(51) = (51 \bmod 7 + i) \bmod 7 = (2+1) \bmod 7 = 3$

$H(33) = 5$

$H(30) = (30 \bmod 7 + i) \bmod 7 = (2+2) \bmod 7$

$= 4$

Again a collision

$\therefore\ H(4) = (4+1) \bmod 7$

$= 5 \Rightarrow$ again a collision

$H(5) = (5+1) \bmod 7 = \underline{\underline{6}}$ (no collision)

0	7
1	
2	9
3	51
4	18
5	33
6	30

resulting hash table

c) $H(x) = [h(x) + i^2 c_2] = [h(x) + i^2] \bmod 7$

$H(7) = 0$; $H(9) = 2$; $H(18) = 4$;

$H(51) = [51 + 1^2] \bmod 7 = 52 \bmod 7 = 3$;

$H(33) = 5$; $H(30) = [30 + 2^2] \bmod 7$

$= 34 \bmod 7 = 6$;

c) $H(x) = [h(x) + i^2 c_2] = [h(x) + i^2] \bmod 7$

$H(7) = 0$; $H(9) = 2$; $H(18) = 4$;

$H(51) = [51 + 1^2] \bmod 7 = 52 \bmod 7 = 3$;

$H(33) = 5$; $H(30) = [30 + 2^2] \bmod 7$
$= 34 \bmod 7 = 6$;

0	7
1	
2	9
3	51
4	18
5	33
6	30

resulting hash table

d) $H(x) = [h_1(x) + i h_2(x)] \bmod 7$

$h_1(x) = x \bmod 7$
$h_2(x) = 5 - (x \bmod 5)$

$H(7) = 7 \bmod 7 = 0$; $H(9) = 9 \bmod 7 = 2$

$H(18) = 4$; $H(51) = [51 \bmod 7 + \{5 - (51 \bmod 5)\}] \bmod 7$

$= [2 + \{5 - 1\}] \bmod 7$

$H(51) = 6 \bmod 7 = 6$

$H(33) = 5$;

$H(30) = [30 \bmod 7 + 2\{5 - (30 \bmod 5)\}] \bmod 7$

$= [2 + 2\{5 - 0\}] \bmod 7 = 12 \bmod 7 = 5$

$= [2 + 2\{5-0\}] \bmod 7 = 12 \bmod 7 = 5$

There's a clash

$\therefore \Rightarrow H(5) = [5 \bmod 7 + 1\{5-0\}] \bmod 7$
$= (5+5) \bmod 7$
$= 10 \bmod 7 = 3$

$\Rightarrow H(30) = 3$

0	7
1	
2	9
3	30
4	18
5	33
6	51

resulting hash table

Scanned by CamScanner

1) If an element occurs more than $\lceil n/2 \rceil$ times then that element would also be the median.

∴ ① Using the SELECT (k, n, A) algorithm we'll get the median.

ⓐ Divide i/p into $n/5$ groups of 5 elements each.

ⓑ Find the median of each group

ⓒ Find the median of medians by calling SELECT$(n/10, n/5)$ recursively

② Let median $= m$,

$k=0$

for i to length.Array

if $(A[i] == m)$

$k = k+1$

② Let median = m,
k=0
for i to length.Array
if (A[i] == m)
k=k+1;
if ($k > \lceil n/2 \rceil$)
then m occurs $\lceil n/2 \rceil$ times

else
it doesn't

Both ① & ② run in $O(n)$ time

∴ This algorithm runs in $O(n)$.