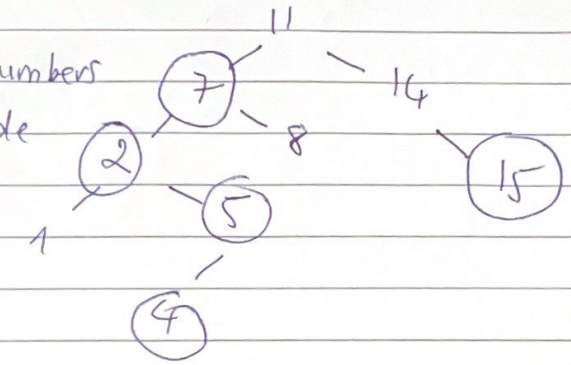
Q1: 0, 1, 2, 3, 4, 6, 7, 9, 10, 13

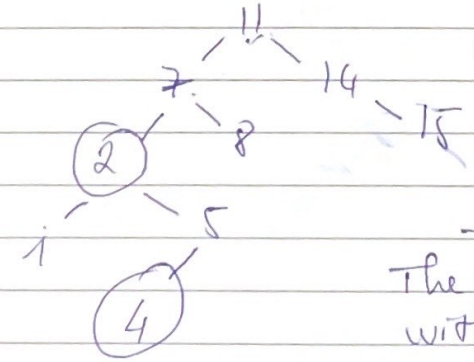
Q2:

1. The Tree is violated as this 2 rules:
   - if a node is red, then both its children are black
     7, 2, 5, 4 are red but 4 is child of 5
     5 is child of 2
     2 is child of 7
   - For each node, all paths from that nodes to descendant leaves contain the same number of black nodes.
     node 4 is the leaves and has the height of 1
     when 1, 8, 15 are leaves but height are 2.

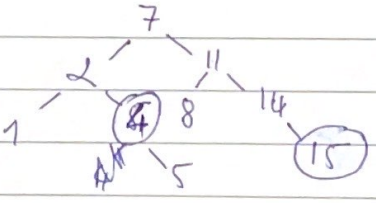
- origin tree
The circle numbers are red node




remove red node to black not at value 15, 7, 5
we have, or we rotate at 7 and apply case D



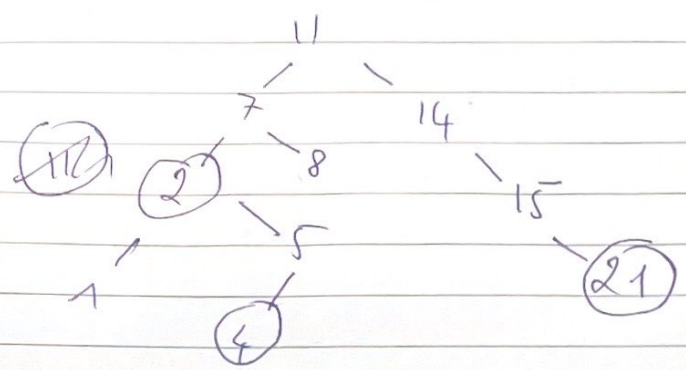

or




- So we can maintain The rule of Red-black tree with all black height is 3.

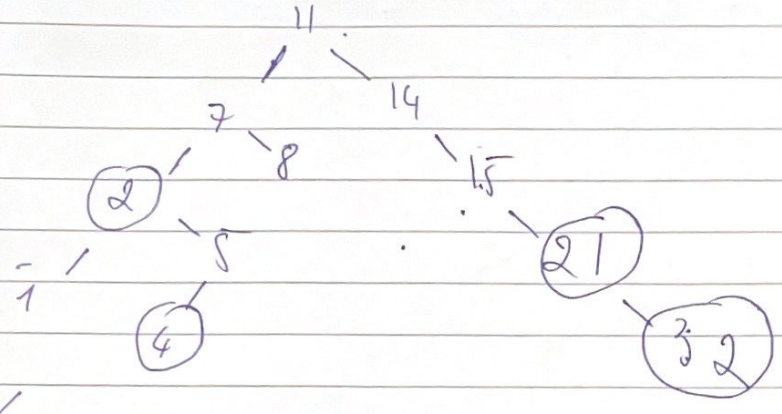
Q2.2:

insert 21



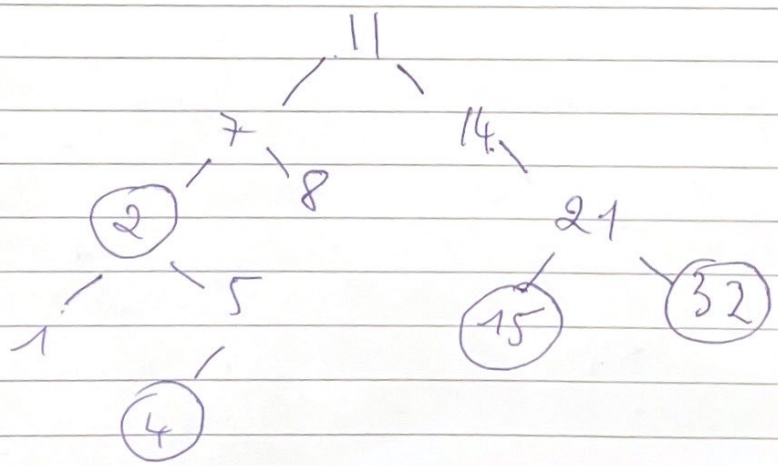

Not Violated any rule

insert 32:

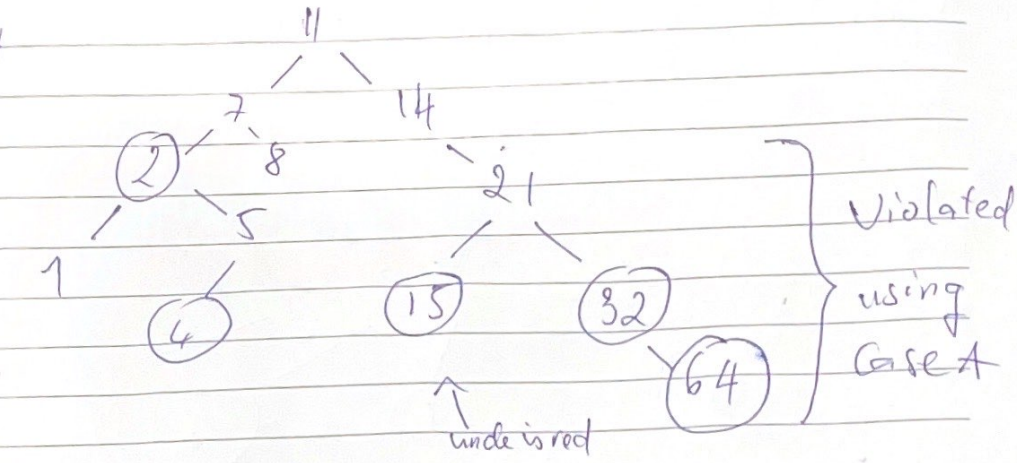



Violated using case D, because uncle of 32 is black

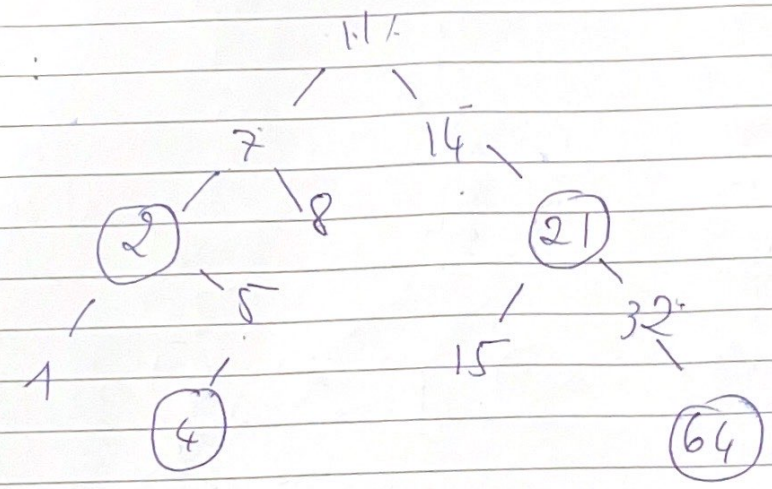
~~the~~ ~~case A~~ case D



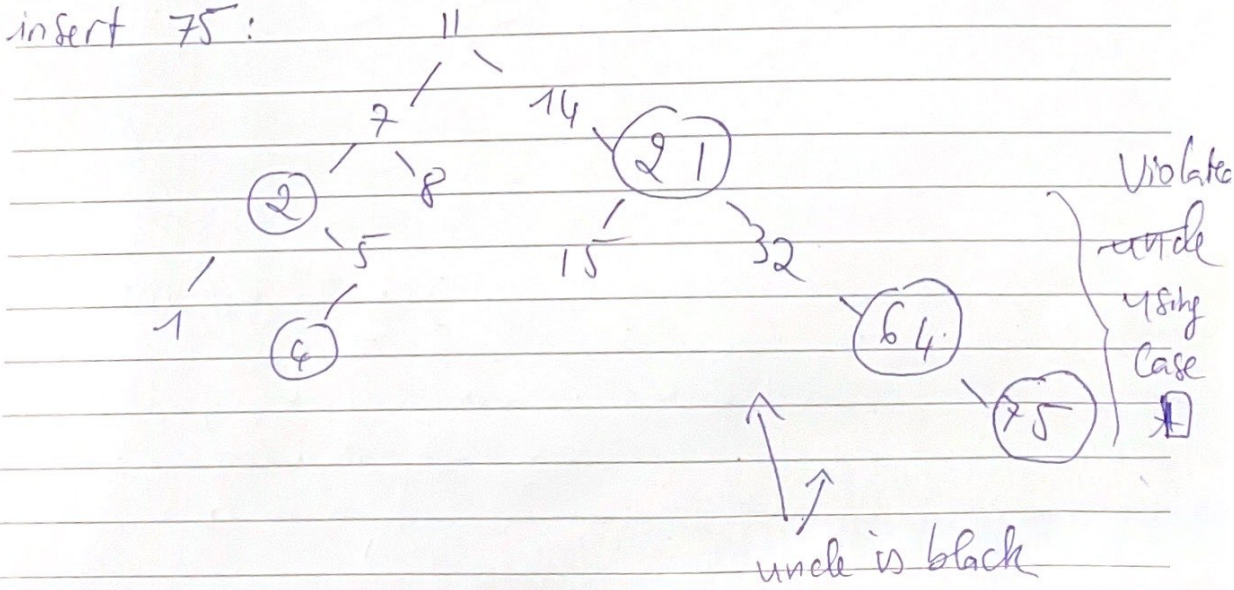
insert 64



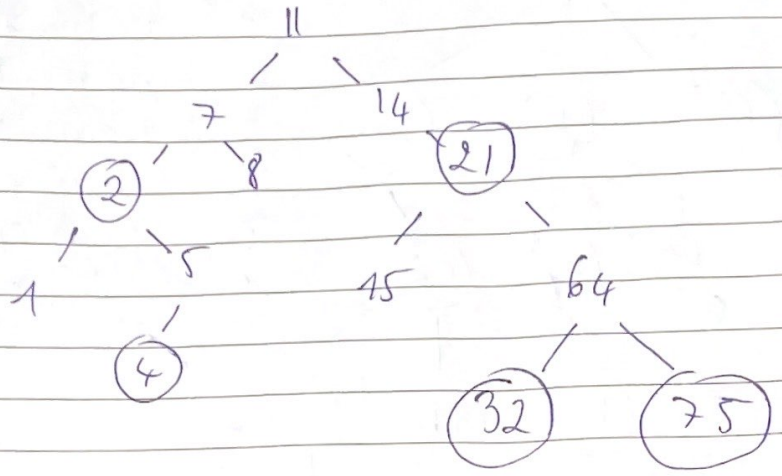

Case A:



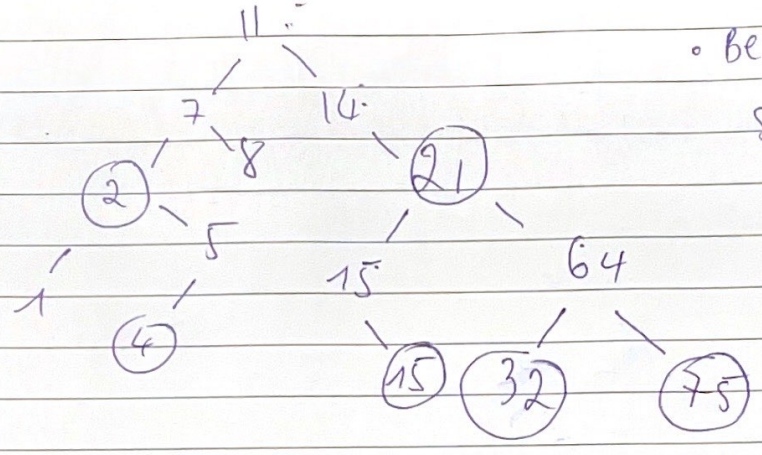

insert 75:



Case b:




insert 15:



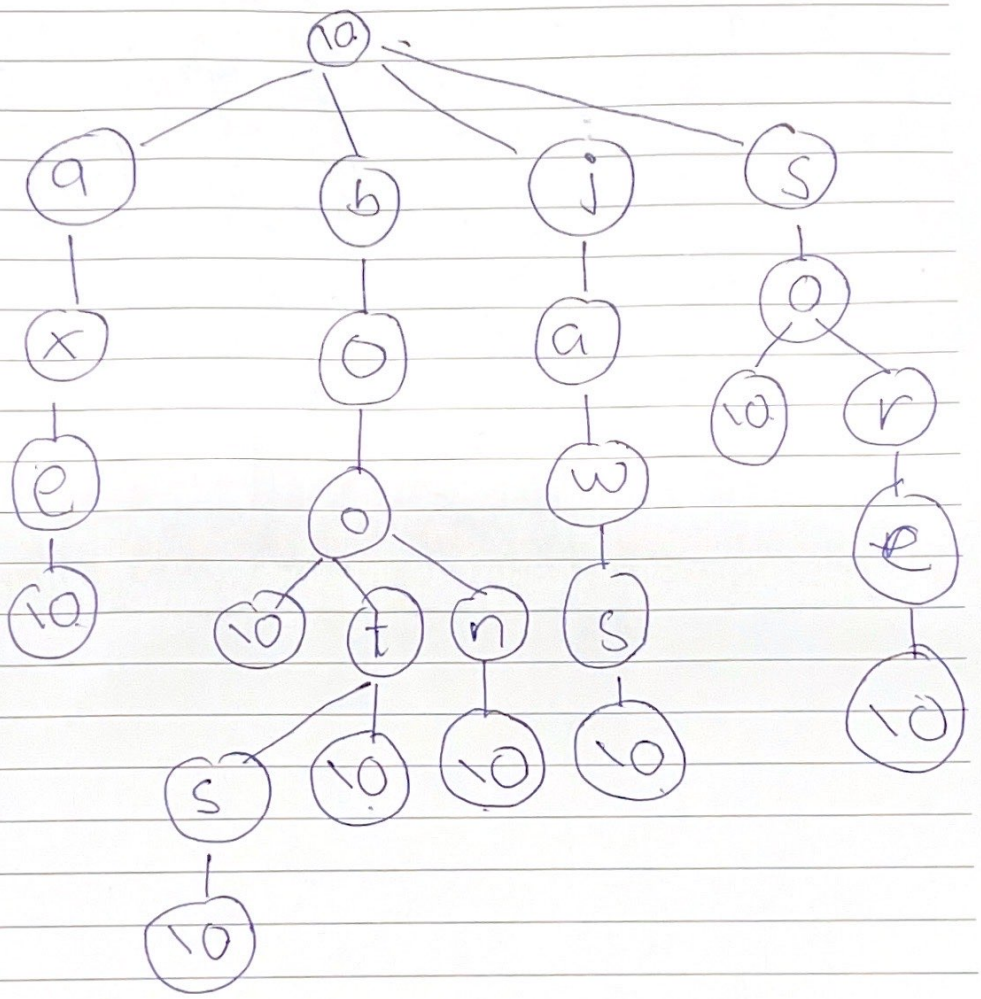

- Because $15 \geq 15$ so put node 15 in

- There is no violation in the R-B tree rule.

Q3.1: using tree to represent the trie

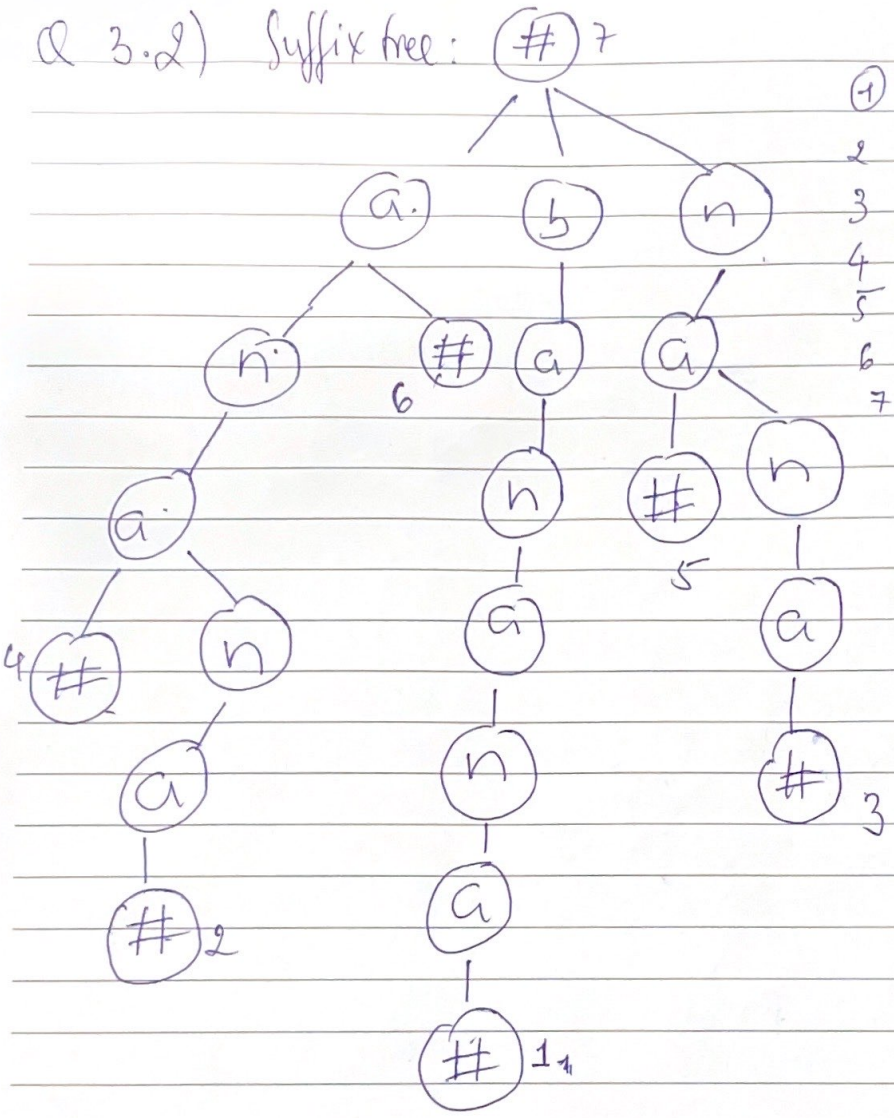
start at empty node:

axe
boo
boot
boon
boots
jaws
so
sore



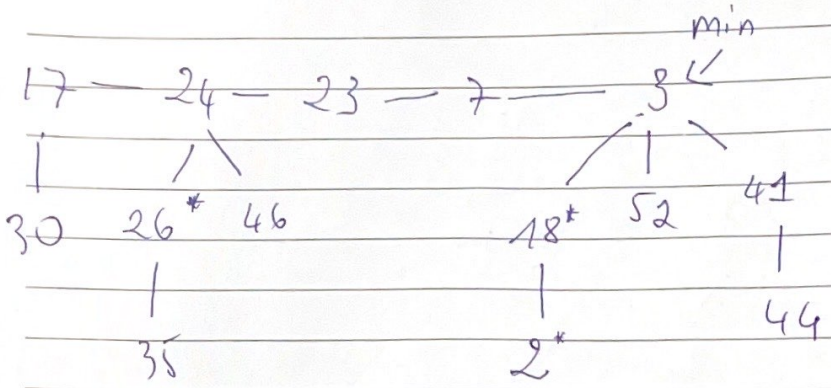

The order of insertion may matter since the efficient of the search for the tree. for example the word initial can be found faster in array a, b, j, s by using binary search. in a large data, it will give a good performence in the run time of $O(\log n)$ where n is the sibling node (in this case is a, b, j and s)

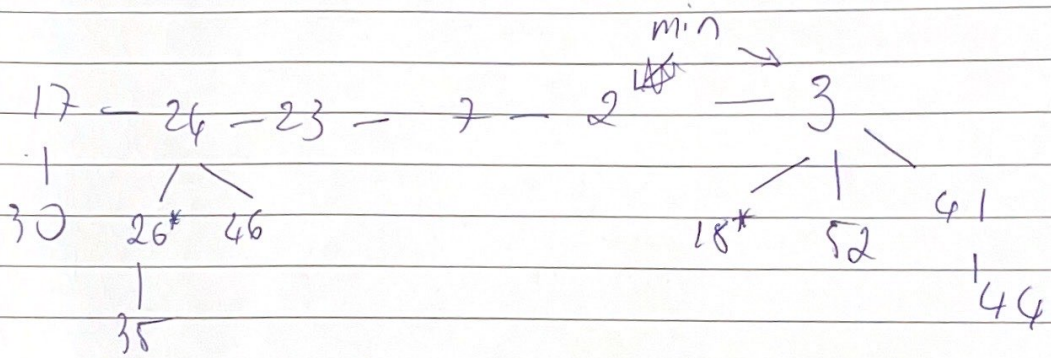
Q 3.2) Suffix tree:

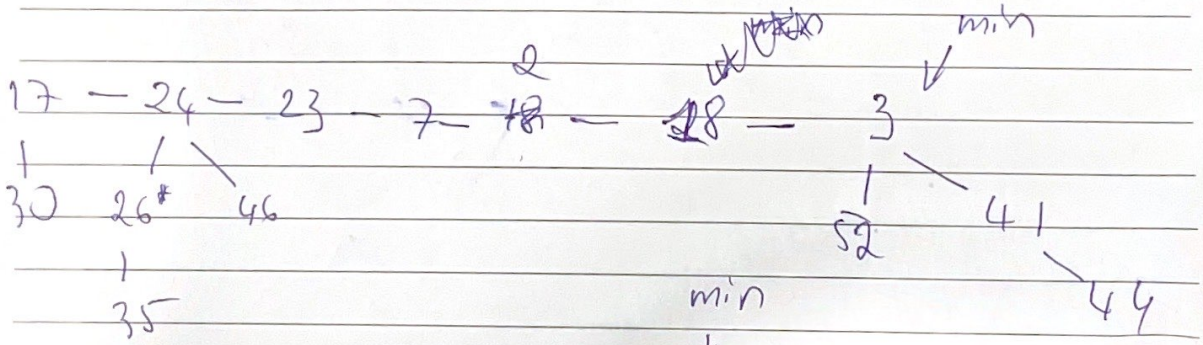


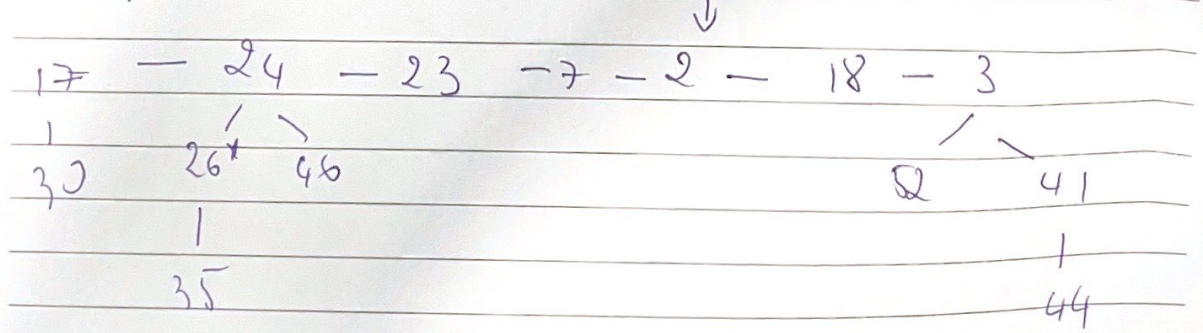

① banana #
2 anang #
3 nana #
4 ana #
5 na #
6 a #
7 #

4) Decrease 39 to 2

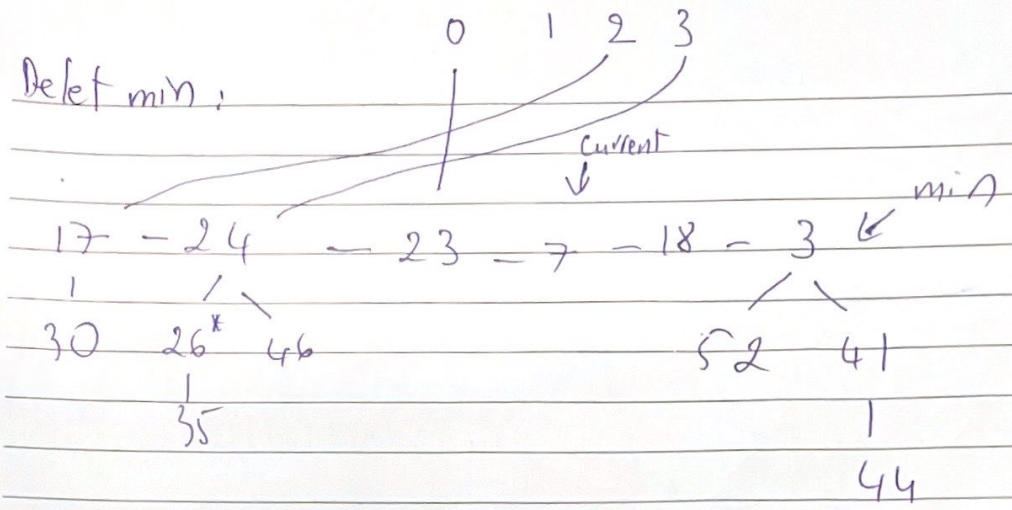


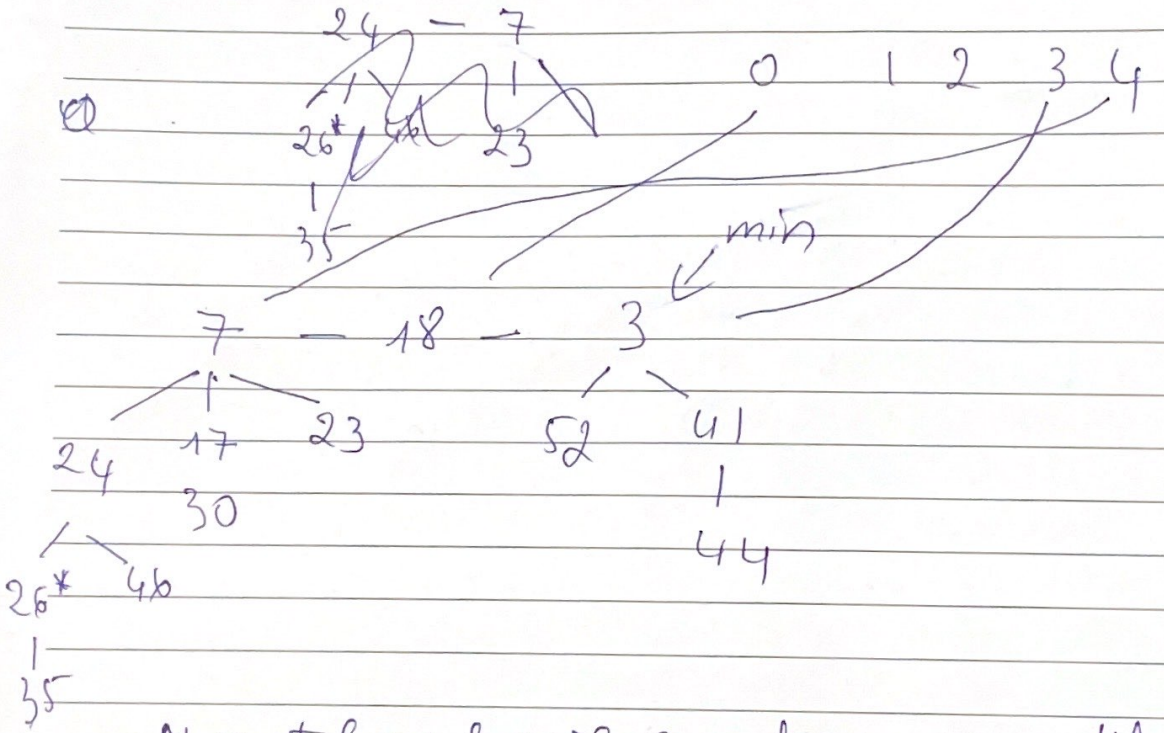



Delet min:



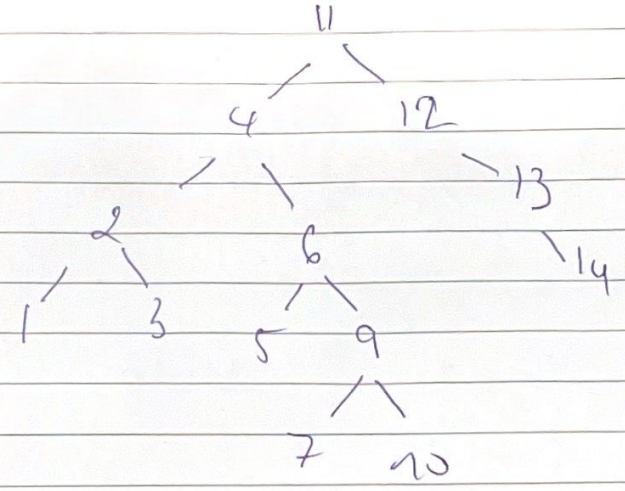

merge current to 0, Then to 2 ; Then to 3



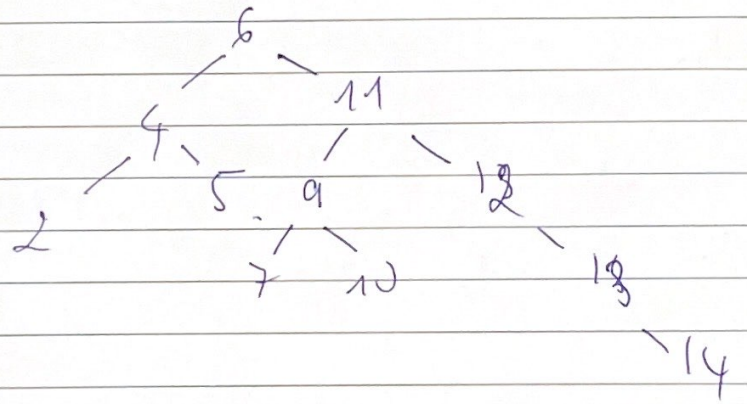

. No root have the same degree consolidation is finished.

Q5)

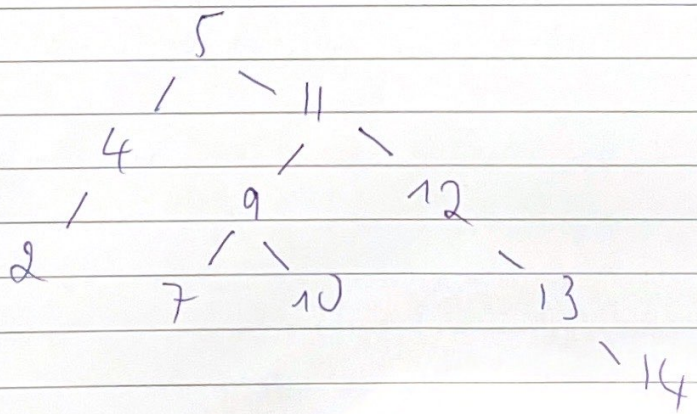
1)



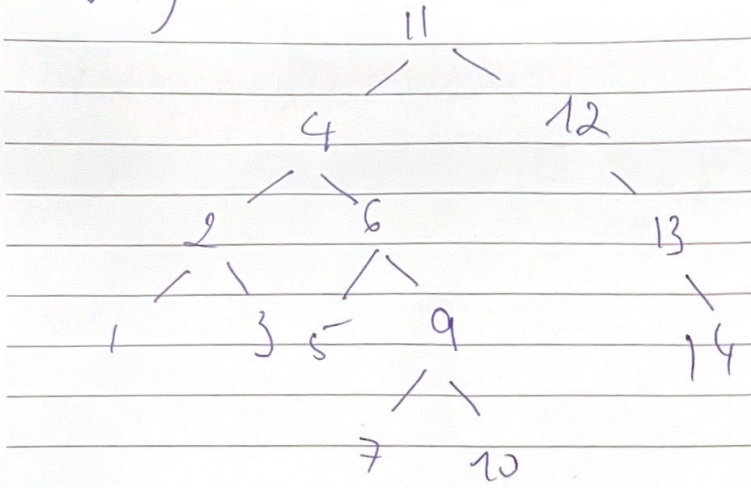

(zig zag

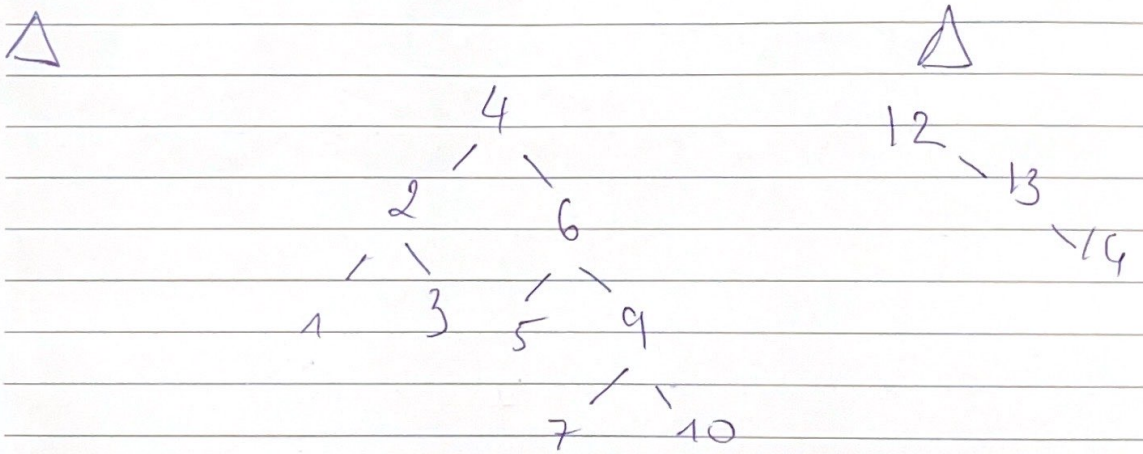


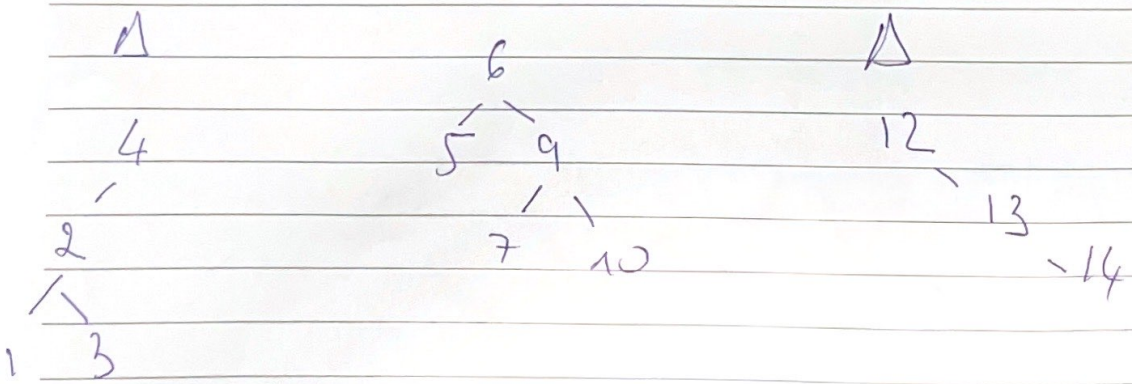

Delete 6:

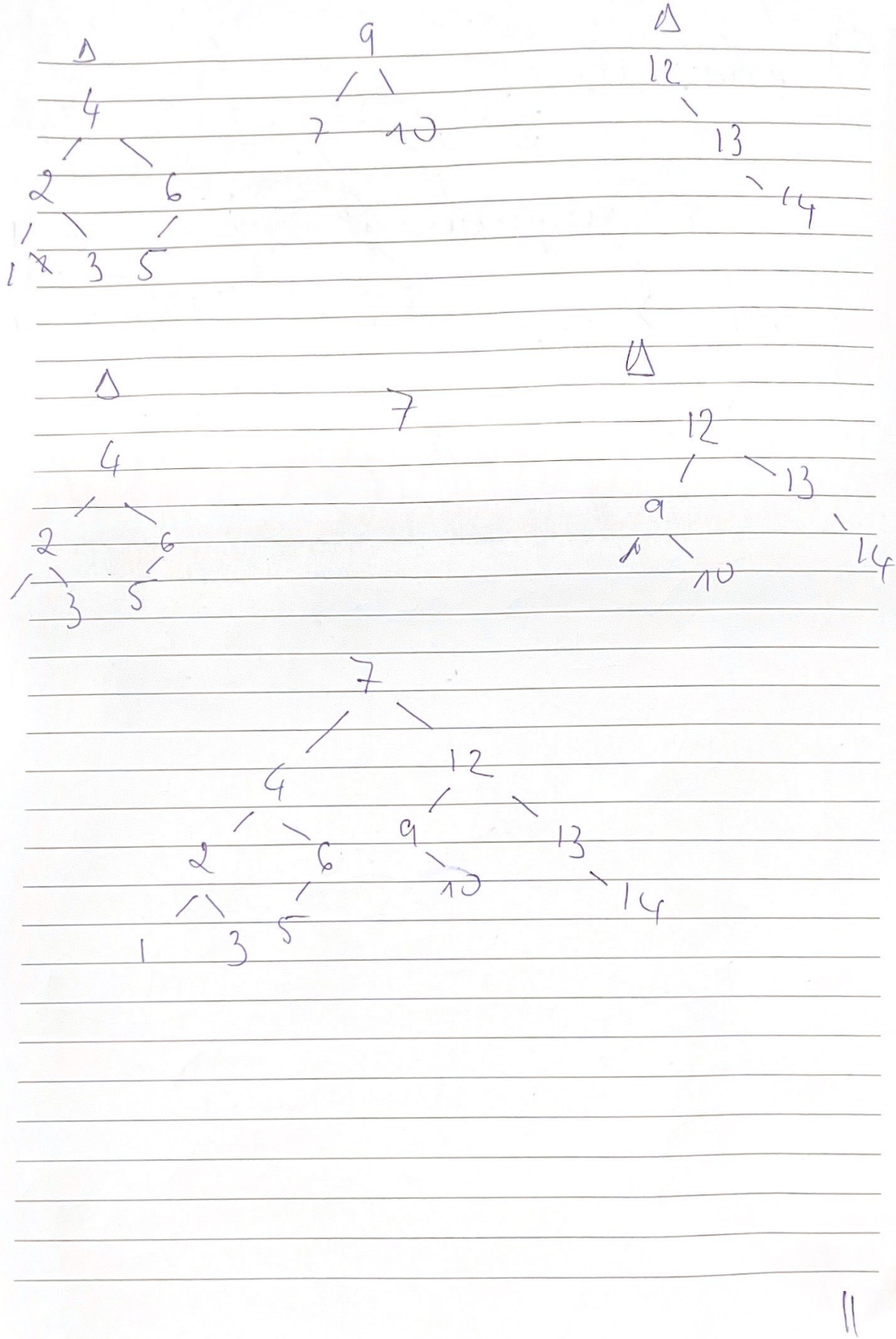


Q5.2)


11
4
12
2
6
13
1
3
5
9
14
7
10

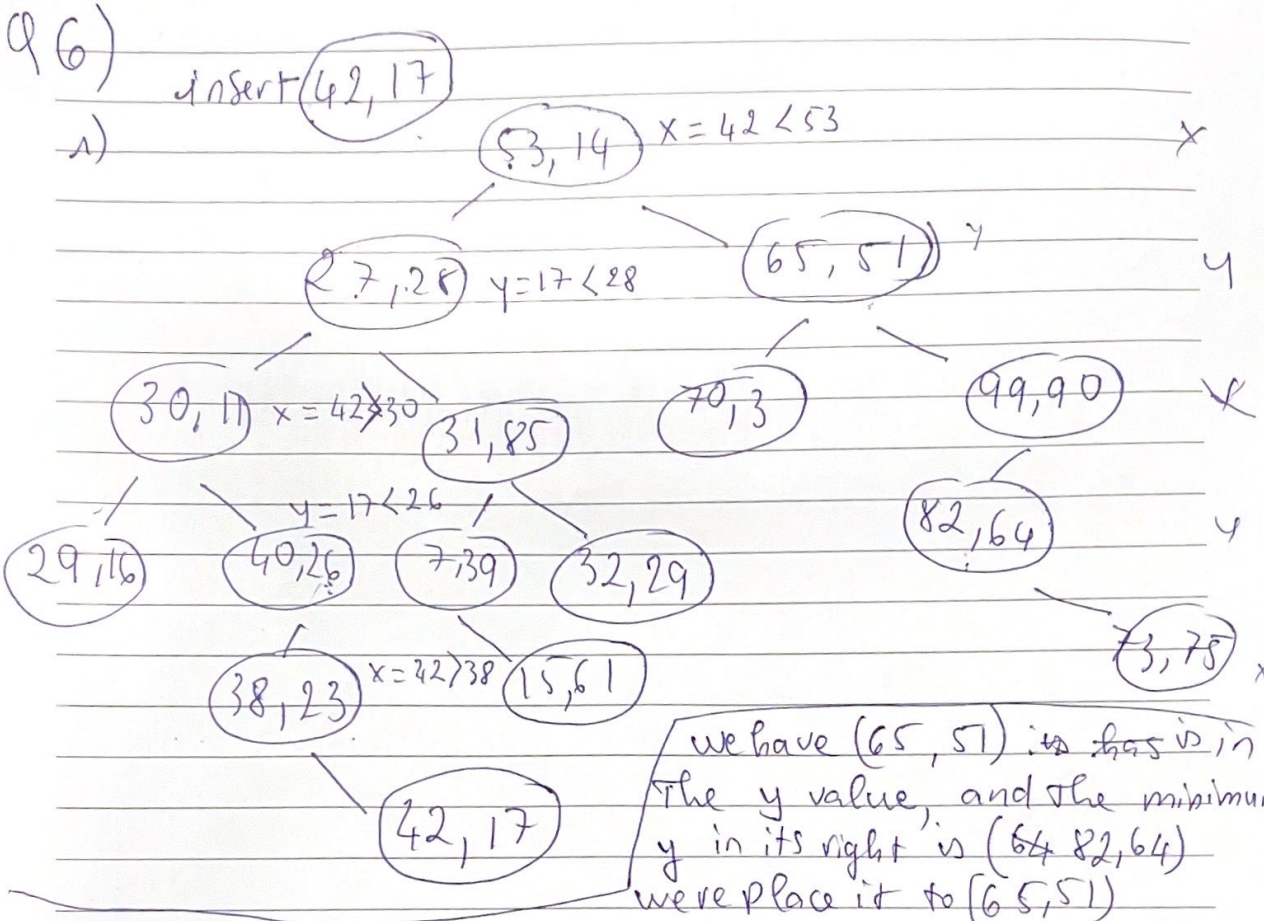

4
2
6
1
3
5
9
7
10
12
13
14


4
2
1
3
6
5
9
7
10
12
13
14


4
2
6
1
3
5
9
7
10
12
13
14
7
4
12
2
6
9
13
1
3
5
10
14

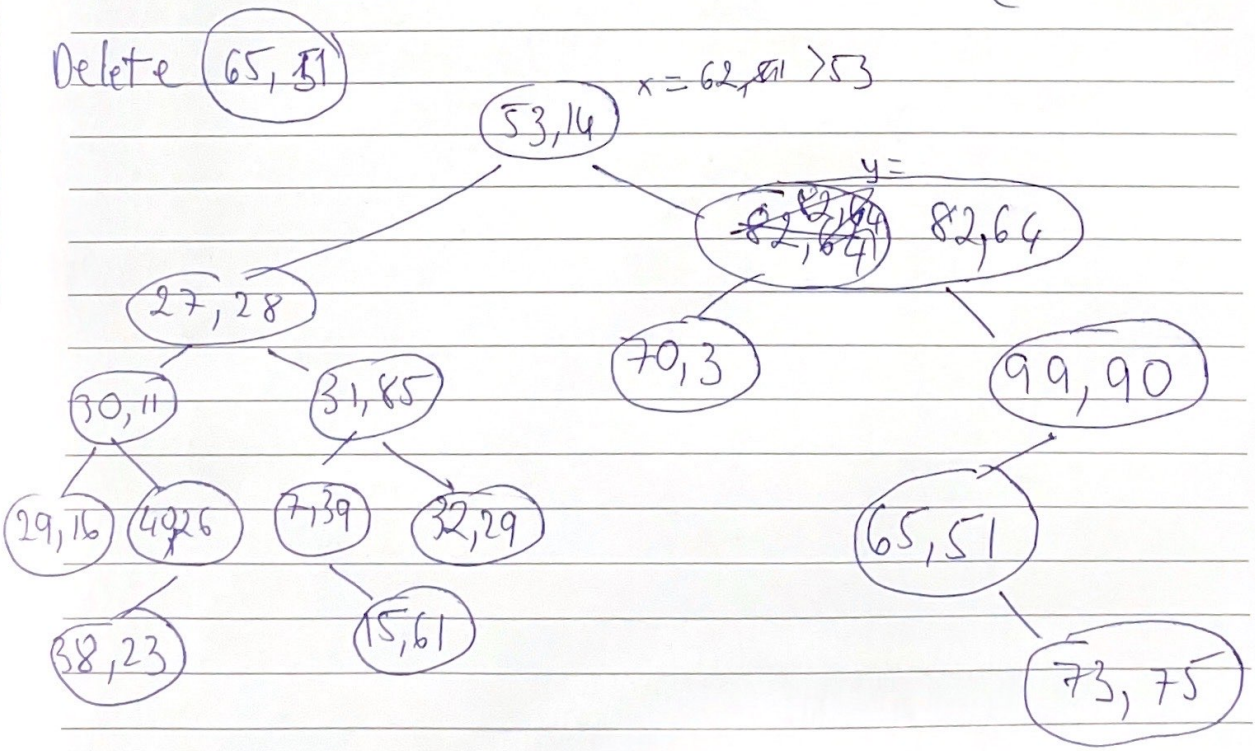
9 6)

a) insert (42,17)




we have (65,51) ~~is~~ ~~has~~ is in the y value, and the minimum y in its right is (~~64~~ 82,64) we replace it to (65,51)

Delete (65,51)



To perform de deletion we still have (73,75) in the right of (65,51), we replace with (65,51) and then, (65,51) become a leave, we can remove it from the tree.