1.

a) a c e d q n r w s

b) q e c a d r n s w

c) a c d e n w s r q

(2)

1. 65

2. 65
/
13

3. 65
/
13
\
16

double rotate

4. 16
/ \
13 65

5. 16
/ \
13 65
\
52

6. 16
/ \
13 65
\
52
/
28

Single rotate

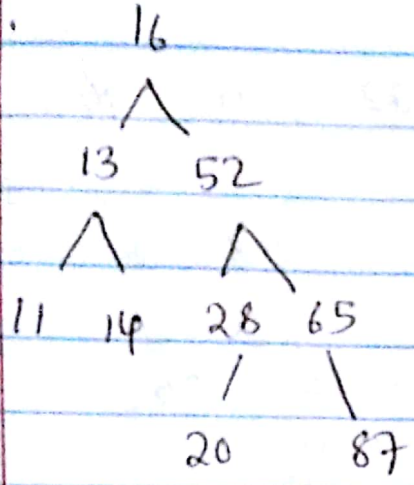
7. 16
/ \
13 52
/ \
28 65

8. 16
/ \
13 52
/ \ /
11 28 65

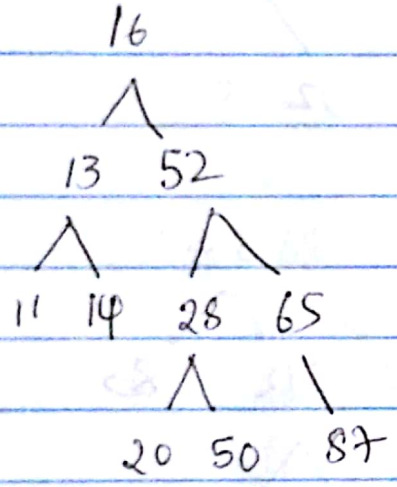
9. 16
/ \
13 52
/ / \
11 28 65
/
20

10. 16
/ \
13 52
/ \ / \
11 14 28 65
/
20

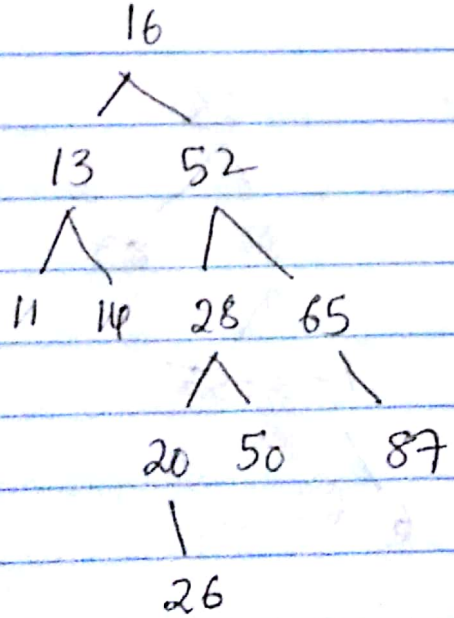
11.



12.

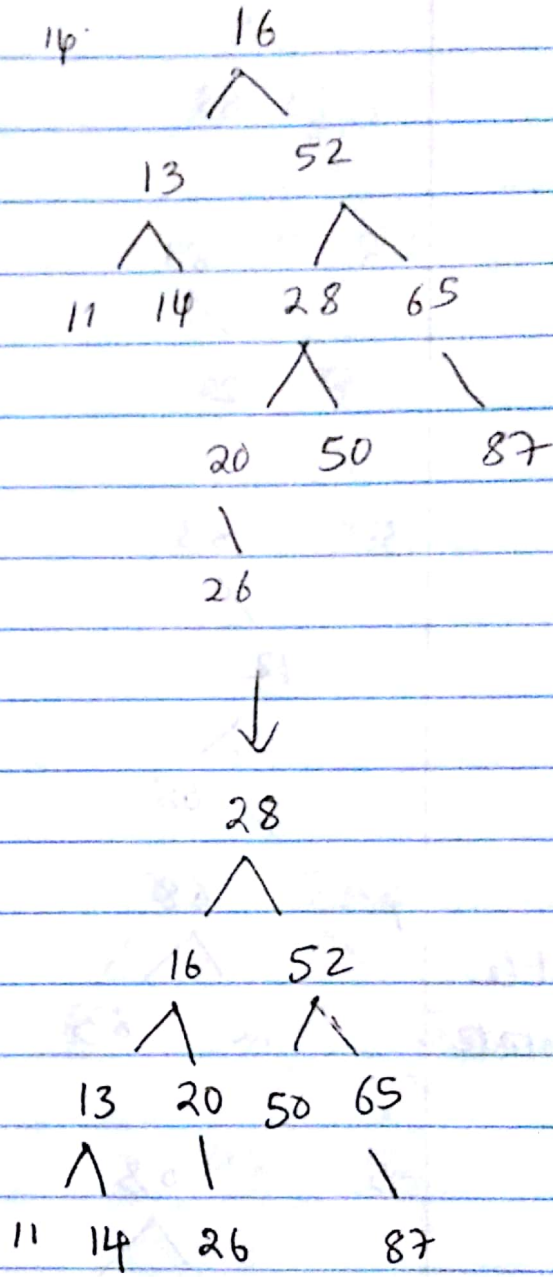


13.



double rotate

14.



(3) 83 12 68 55 32 6 46 57 62

1. 83

2. 83
/
12

3. 83
/
12
\
68

4. 68
/ \
12 83

double rotate

5. 68
/ \
12 83
\
55

6. 68
/ \
12 83
\
55
/
32

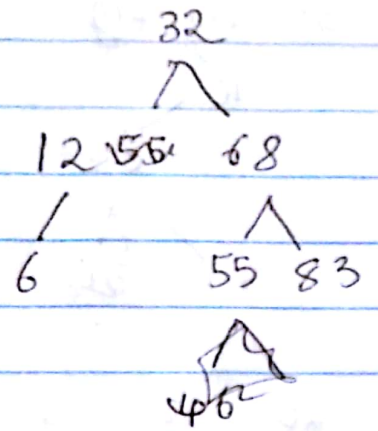
7. 68
/ \
32 83
/ \
12 55

double rotate

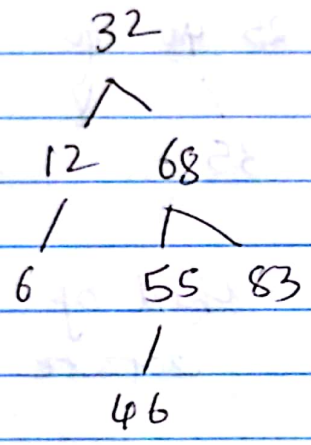
8. 68
/ \
32 83
/ \
12 55
/
6

9.

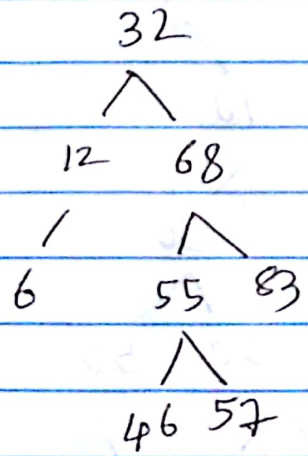
single rotate



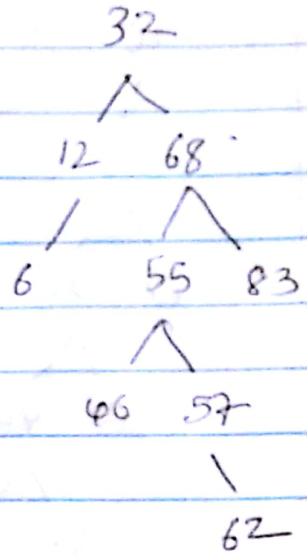
10.



11.

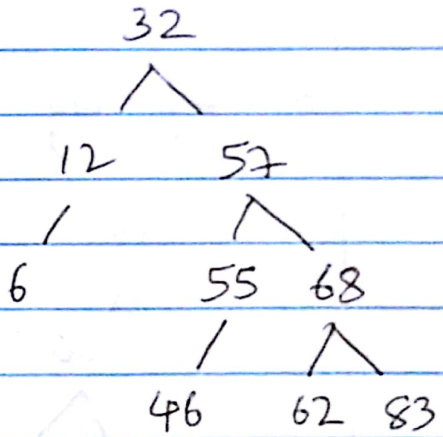


12.



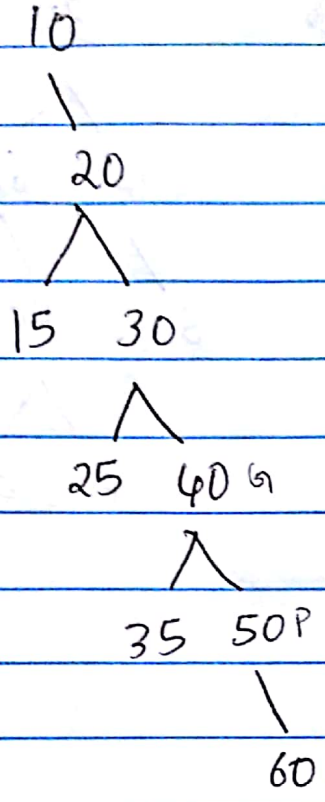
13.

double rotate

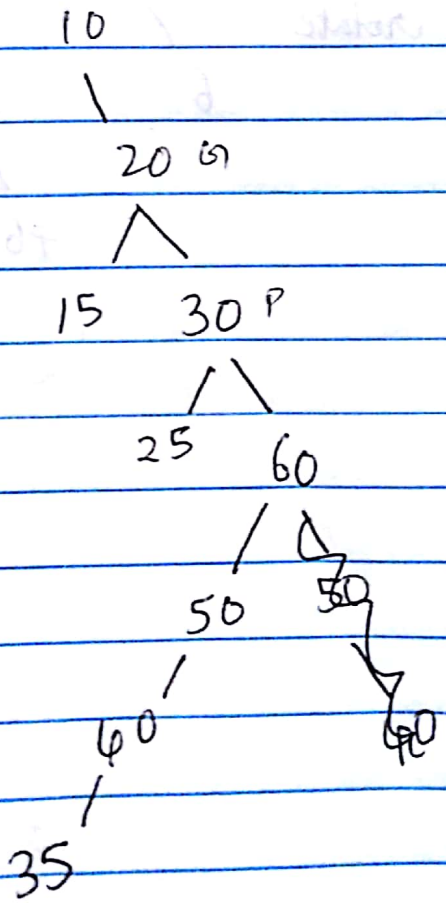


(4)

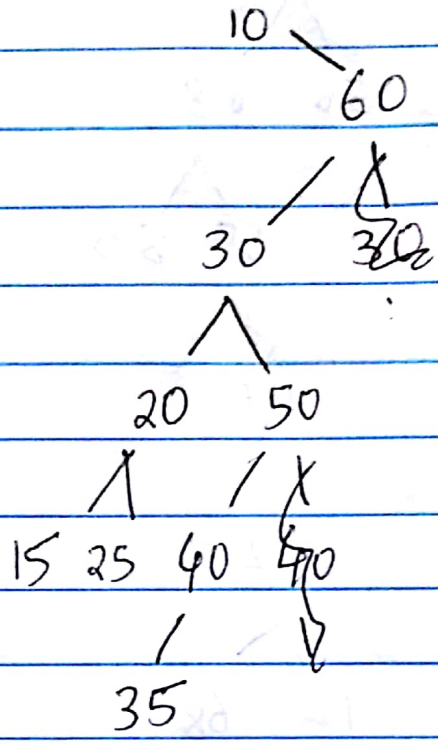
1.



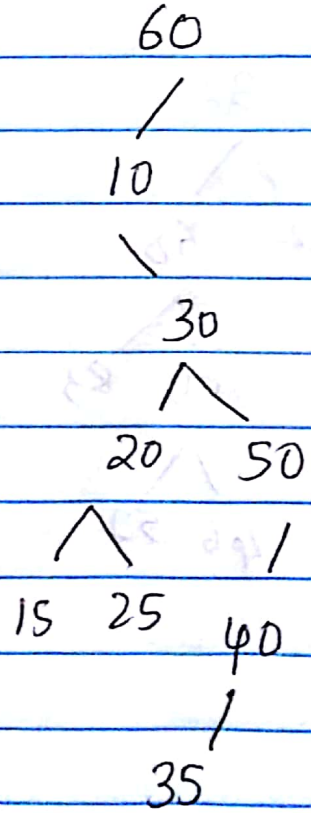
2. zig-zig



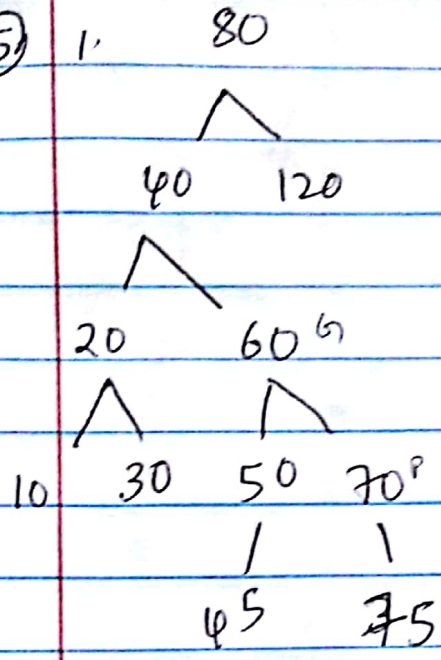
3. zig-zig



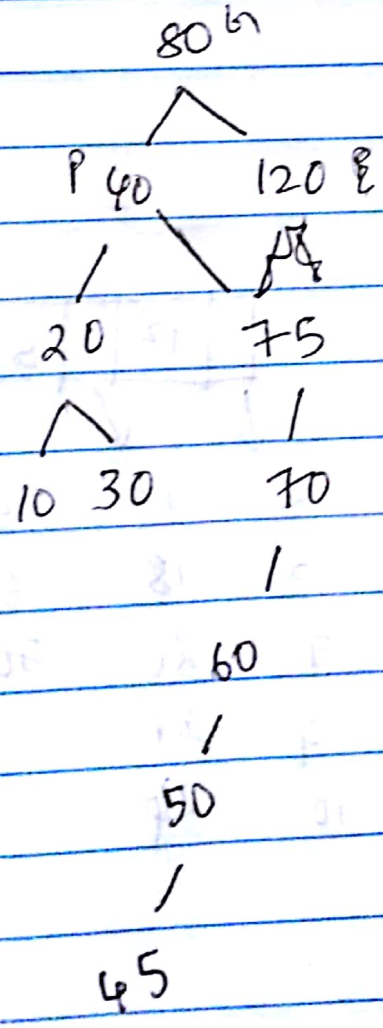
4. Child of root, rotate



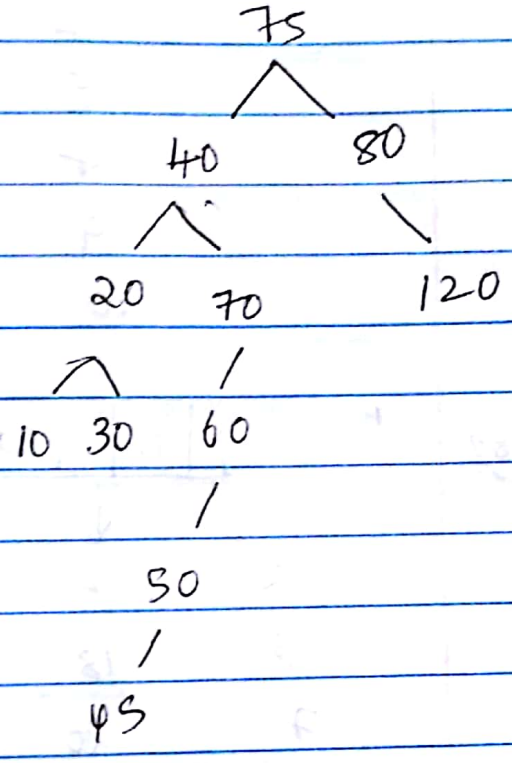
(5) 1.



2. zig-zig

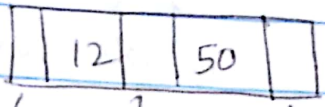


3. zig-zag



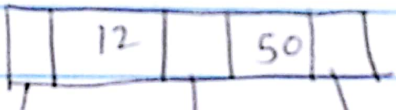
6)

1.



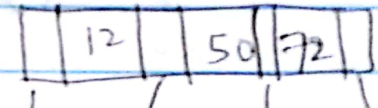
2	12	50
5	18	65
7	20	70
9	21	72
10	24	78

2.

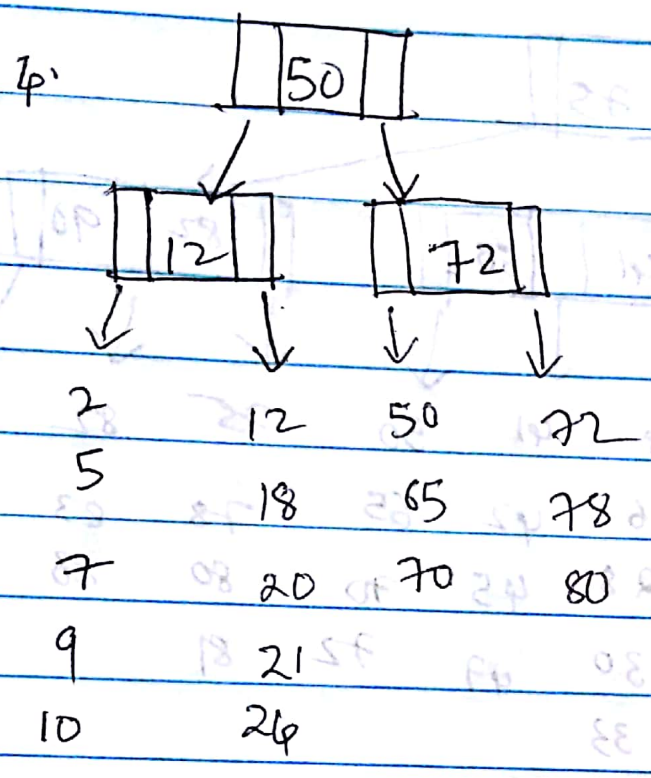


2	12	50	
5	18	65	
7	20	70	
9	21	72	
10	24	78	
		80	

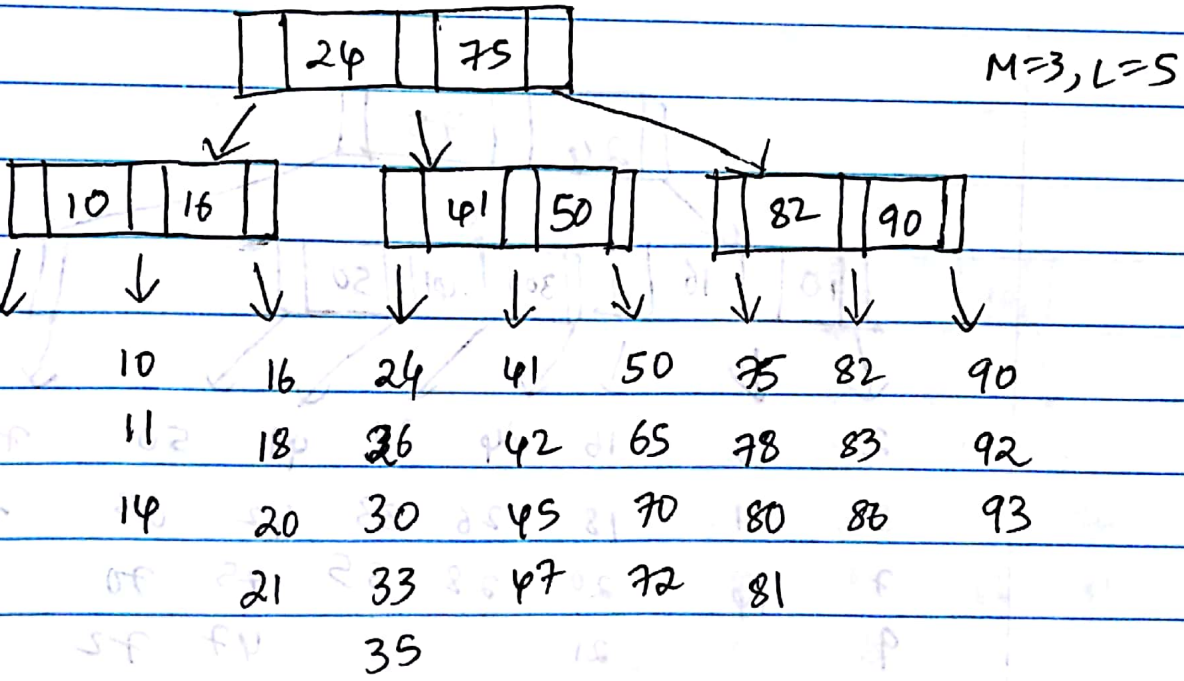
3.



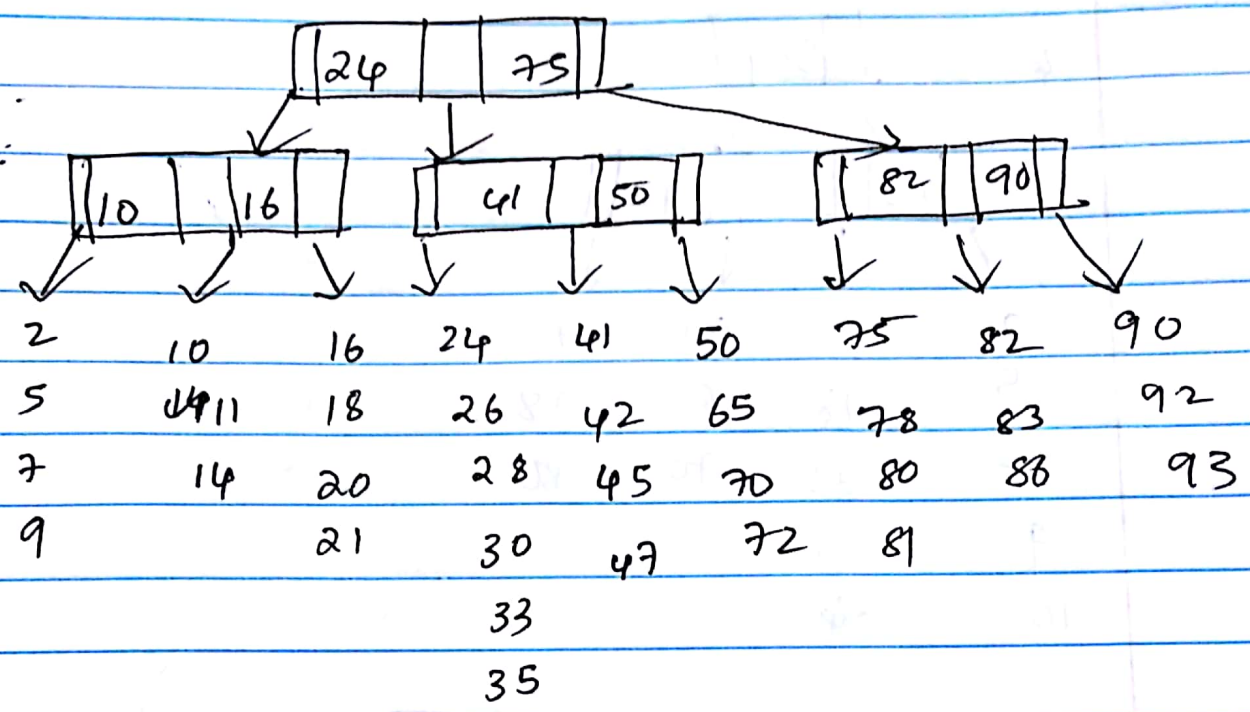
2	12	50	72
5	18	65	78
7	20	70	80
9	21		
10	24		

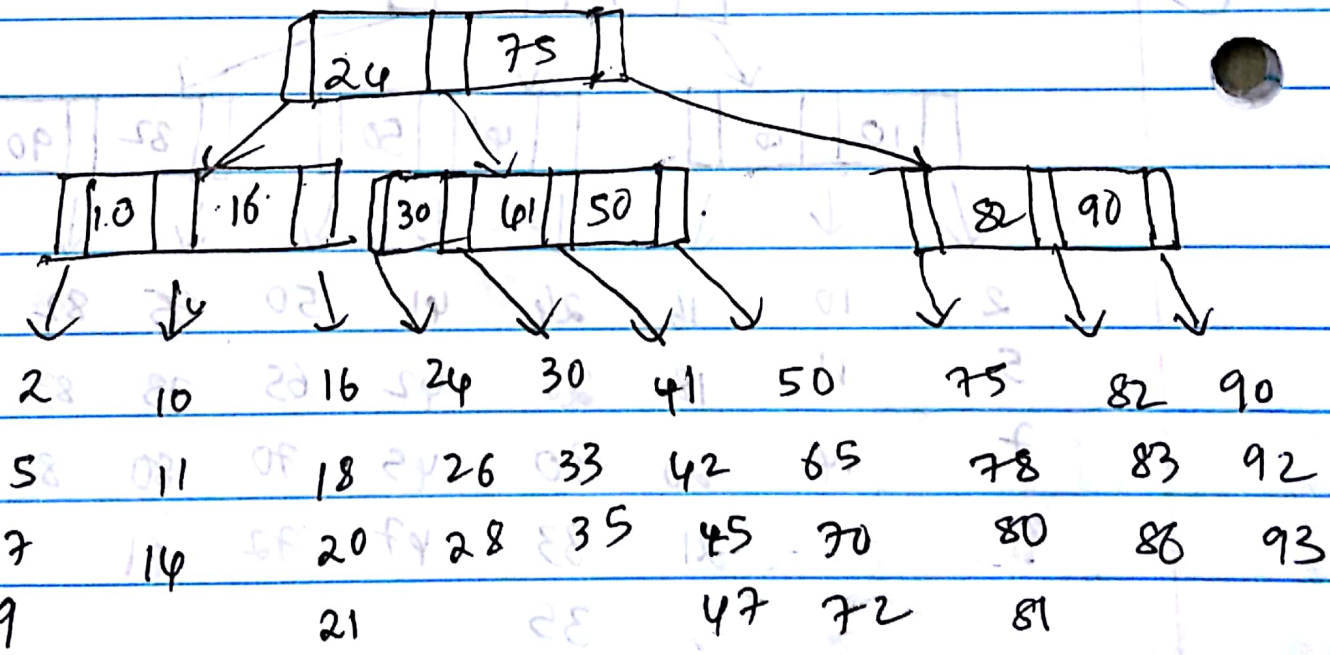


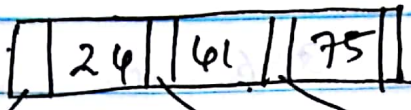
(7)

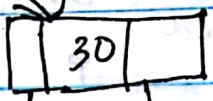


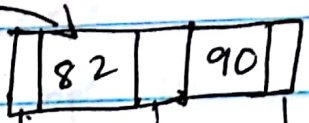
M=3, L=5









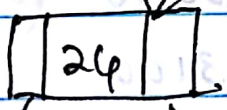


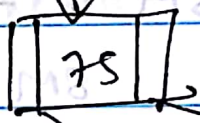
2 10 16 24 30 41 50 75 82 90

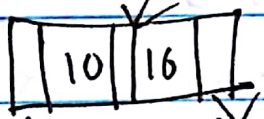
5 11 18 26 33 42 65 78 83 92

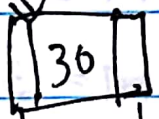
7 14 20 28 35 45 70 80 86 93

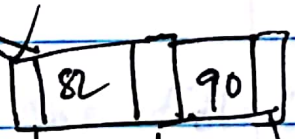
9 21 47 72 81











2 10 16 18 24 30 41 50 75 82 90

5 11 20 26 33 42 65 78 83 92

7 14 21 28 35 45 70 80 86 93

9 47 72 81

(8) block size : 3096

records to be stored : 36

Key : 4

M = ? L = ? Pointers = 4 bytes each

⇒

Keys to be stored = M-1

$4(M-1) + 4M = 8M - 4$

$8M - 4 = 3096$

$8M = 3100$

$M = 387.5$

$M = 387$

$L = \frac{3096}{38} = 86$

9. 8,600,000

$L = 86$

Assuming the worst case that all the records are half-full,

$$\frac{8,600,000}{\frac{86}{2}} = 200,000 \text{ leaves}$$

Levels: There are atmost $\frac{200,000}{(387/2)}$

$= 258.39$ nodes at the level before the leaves.

$$\frac{\log N}{\log M/2} = \frac{\log 200,000}{\log \frac{387}{2}} = \frac{5.301}{2.28} = 2.19$$

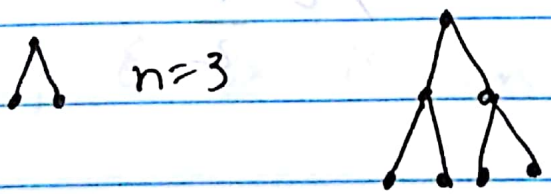
$2.19 \simeq 3 \Rightarrow$ Levels are $3+1 = 4$

(10) Consider a binary tree of n nodes, each node will have 2 pointers, each of which may or may not be null, so the tree will have 2n pointers.

We have n nodes, excluding the null node, every node must have a pointer pointing to it, hence, we have n-1 pointers that are not null.

No. of null pointers = $2n-(n-1) = 2n+1-n$
$= n+1$.

(11) In a perfect binary tree (one filled at every level), adding another level will make the total no. of nodes in the tree to be $2N+1$, N being the no. of nodes.



$2n+1 = 2(3)+1$
$= 7$ → n=7