Type: MCQ

Q1. What is the maximum number of divisions made by Euclid's algorithm for the inputs *m* & *n* for any values in the range $1 < m < n \leq 10$? (0.5)

1. ** 5
2. 2
3. 6
4. 3

Q2. Identify the pair of functions in which the first function's order of growth is higher than the second function: (0.5)

1. $n(n + 1)$ and $2000n^2$
2. $(n - 1)!$ and $n!$
3. 2^{n-1} and 2^n
4. ** $log^2{}_2(n)$ and $log_2(n^2)$

Q3. What will be the output list after completing the first pass of bubble sort on the input array 32, 51, 27, 85, 66, 23, 13, 57 ? (0.5)

1. 23, 13, 27, 33, 51, 57, 66, 85
2. 32, 51, 27, 66, 23, 13, 57, 85
3. 27, 33, 51, 23, 13, 57, 66, 85
4. ** 32, 27, 51, 66, 23, 13, 57, 85

Q4. Which of the following can be stated as the problem of finding the shortest Hamiltonian Circuit of the graph? (0.5)

1. Assignment problem
2. **Traveling Salesman problem
3. Breadth first search
4. Depth first search

Q5. How many character comparisons are made by the brute force algorithm in searching for the pattern **RRRRS** in the text

RRRRRRRRRRRRRRRRRRRRRRRRS

(0.5)

1. ** 105
2. 100
3. 21
4. 25

Q6. In what order should we insert the elements {1, 2, 3, 4, 5, 6, 7} into an empty AVL tree so that we don't have to perform any rotations on it?

(0.5)

1. 4,2,1,6,3,5,7
2. 4,2,6,1,3,5,7
3. **4,2,1,6,3,7,5
4. 4,1,2,6,3,7,5

Q7. Which of the following stable sorting algorithm takes the least time when applied to an almost sorted array? (0.5)

1. Quick sort
2. Insertion sort

3. Selection sort
4. ** Merge sort

Q8. Consider the problem of searching an element (x) in an array arr[] of size (n). The problem can be solved in (O(log n)) time if

I.Array is sorted
II.Array is sorted and rotated by (k), where (k) is given and k <= n
III. Array is not sorted (0.5)

1. I only
2. **I and II only
3. I ,II and III only
4. None of the mentioned

Q9. In the context of a extended Binary tree (0.5)

I. The extension of the empty binary tree is a single external node.
II. The number of internal nodes is always 1 more than the number of external nodes.

1. Both are True
2. Both are False
3. **I – True, II – False
4. I – False,II – True

Q10. When a graph is traversed using BFS method, if a new unvisited vertex v is reached for the first time from a current vertex say u, then the edge (u,v) is called as _____ (0.5)

1. ** Tree Edge
2. Cross Edge
3. Back Edge
4. Solid Edge

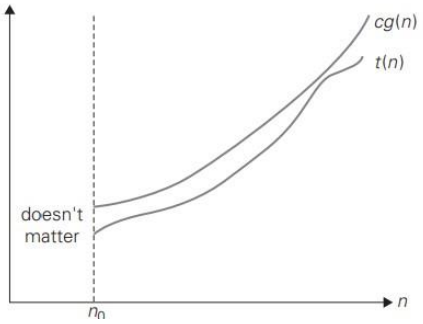
11)

There are mainly three asymptotic notations:

1. Big-O notation
2. Omega notation
3. Theta notation

Big-O notation: A function t (n) is said to be in O(g(n)), denoted t (n) ∈ O(g(n)),if t (n) is bounded above by some constant multiple of g(n) for all large n, i.e., if there exist some positive constant c and some nonnegative integer n0 such that **0.5 Marks**

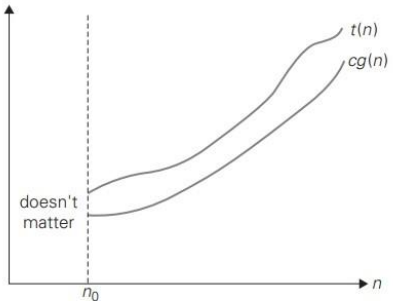
$$t(n) \leq cg(n) \text{ for all } n \geq n0.$$



Example: 100n + 5 ≤ 100n + n (for all n ≥ 5) = 101n ≤ 101n2 **0.5 Marks**

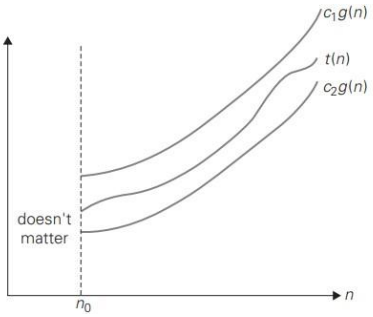
Ω (Omega) notation : A function t (n) is said to be in Ω (g(n)), denoted t (n) ∈ Ω (g(n)), if t (n) is bounded below by some positive constant multiple of g(n) for all large n, i.e., if there exist some positive constant c and some nonnegative integer n0 such that **0.5 Marks**

t (n) ≥ cg(n) for all n ≥ n0.



Example :n3 ∈ (n2): n3 ≥ n2 for all n ≥ 0 **0.5 Marks**

Θ (Theta) notation : A function t (n) is said to be in Θ (g(n)), denoted t (n) ∈ Θ (g(n)), if t (n) is bounded both above and below by some positive constant multiples of g(n) for all large n, i.e., if there exist some positive constants c1 and c2 and some nonnegative integer n0 such that



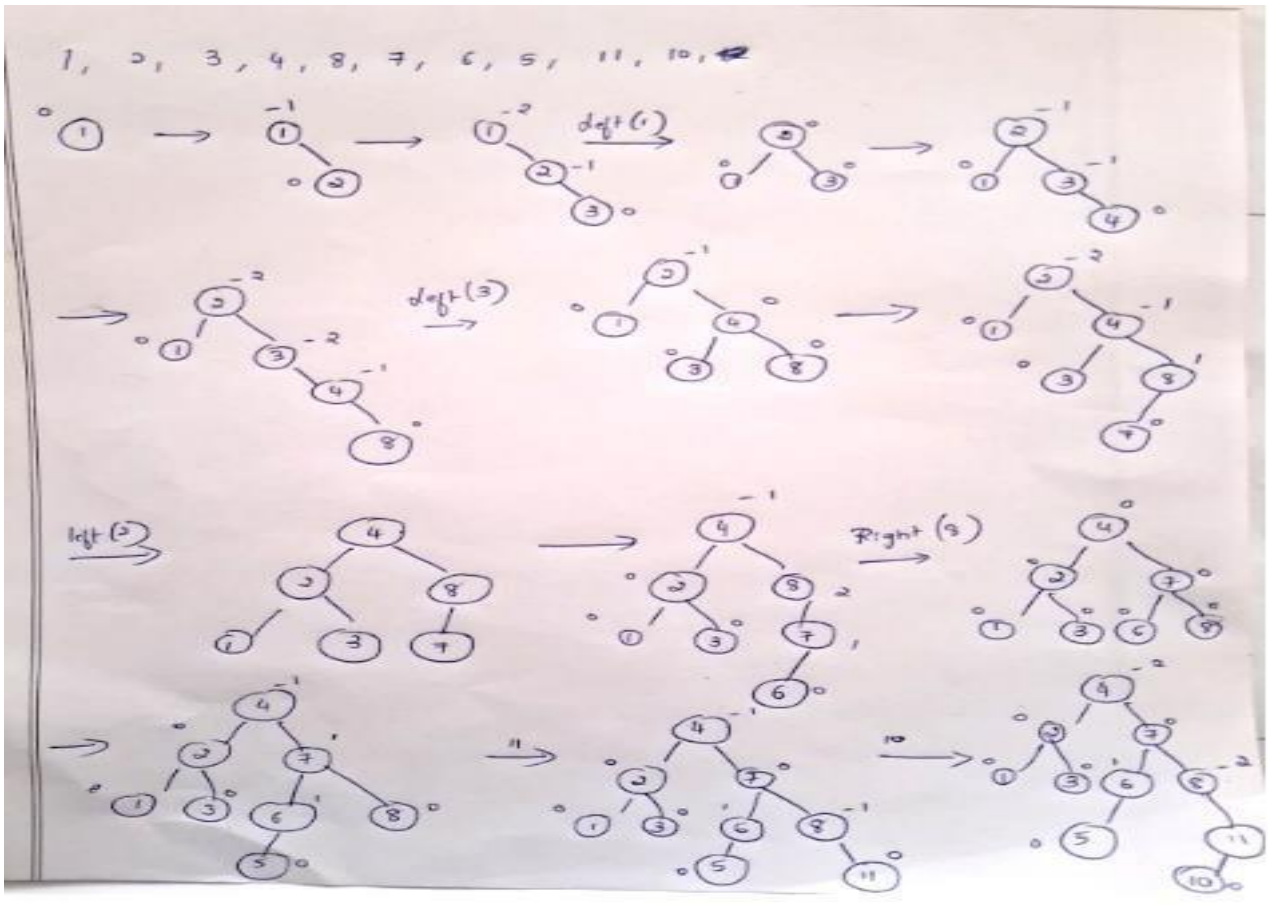
c2g(n) ≤ t (n) ≤ c1g(n) for all n ≥ n0. **0.5 Marks**

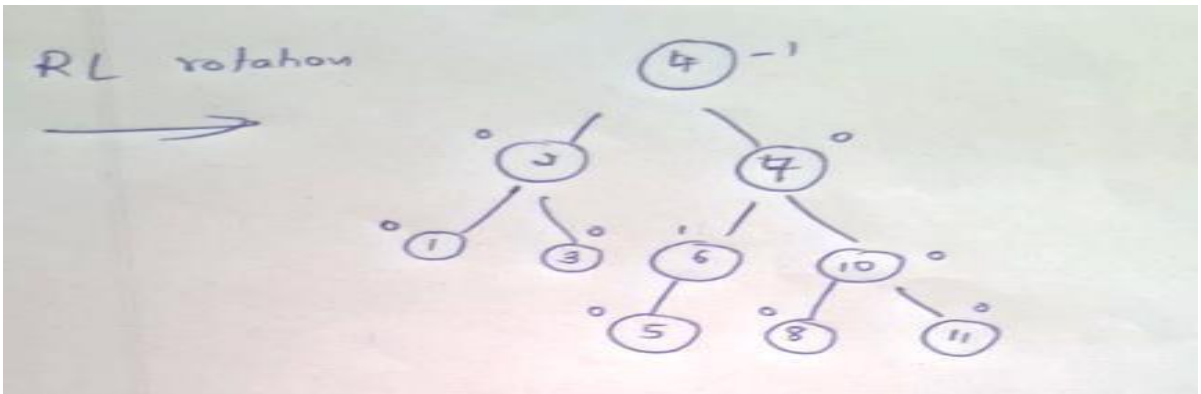
Example : **0.5 Marks**

$$\frac{1}{2}n(n-1) = \frac{1}{2}n^2 - \frac{1}{2}n \le \frac{1}{2}n^2 \quad \text{for all } n \ge 0.$$

Note: marks have giving which has closer meaning to the definition and one who not written the map.

12)





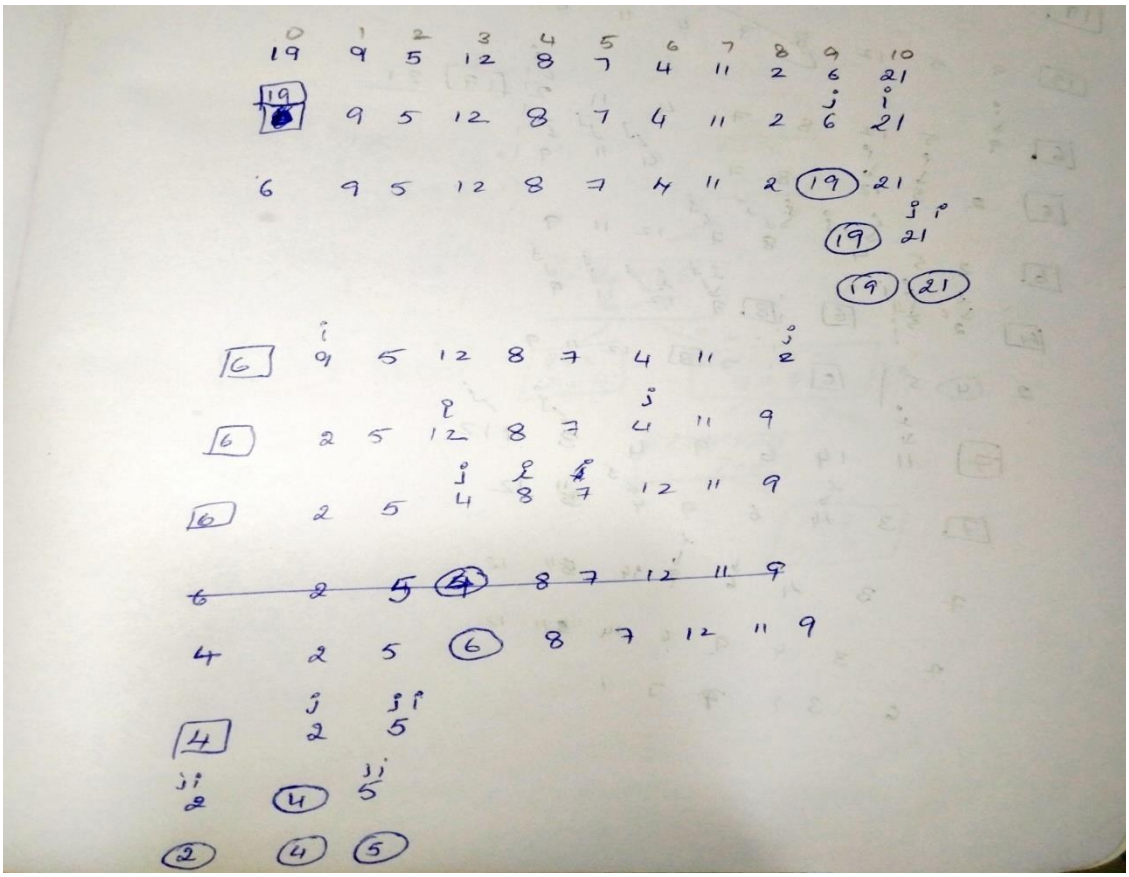
13) Apply quicksort to sort the list: 19, 9, 5, 12, 8, 7, 4, 11, 2, 6, 21 in ascending order. Draw the tree of the recursive calls made.

If any other algorithm other than Hoare is used for partition, then the pseudocode of the algorithm must be written.

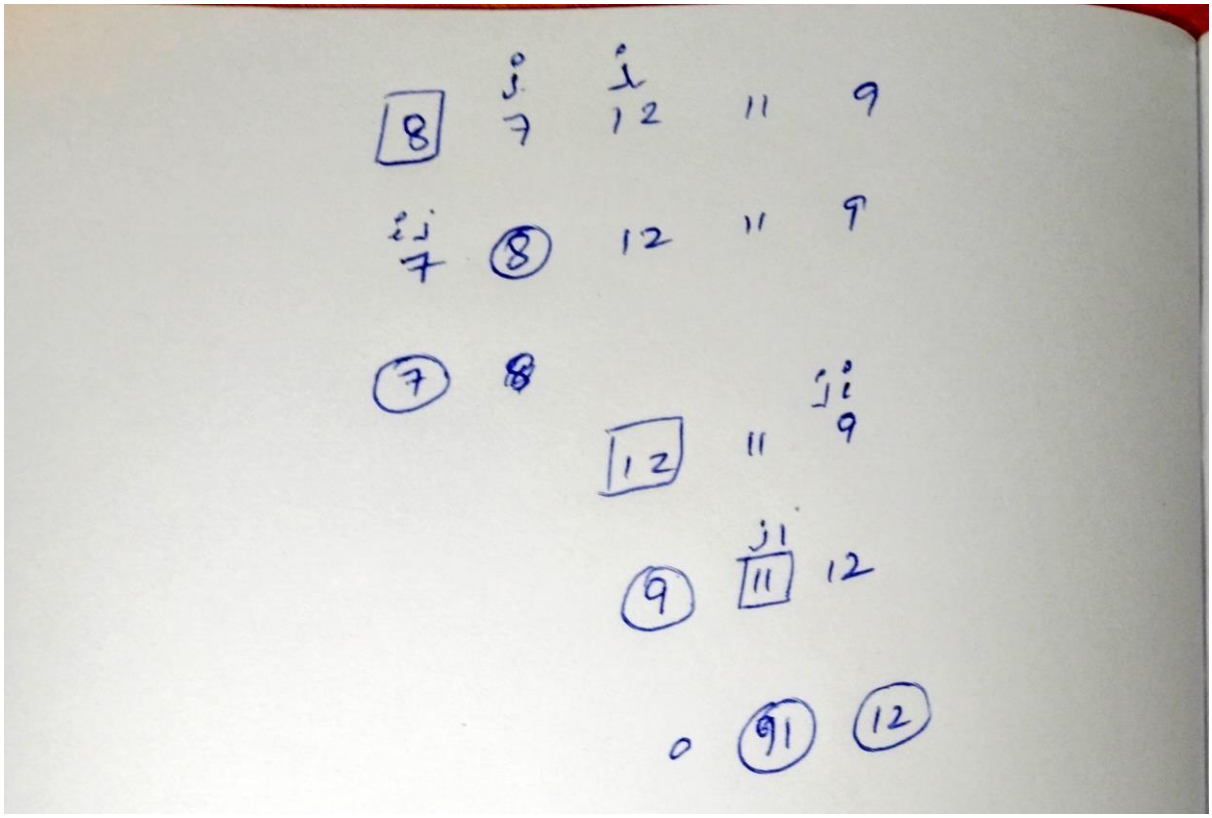
All the steps need to be written clearly.

Refer to the textbook for the format of the tree of the recursive calls.

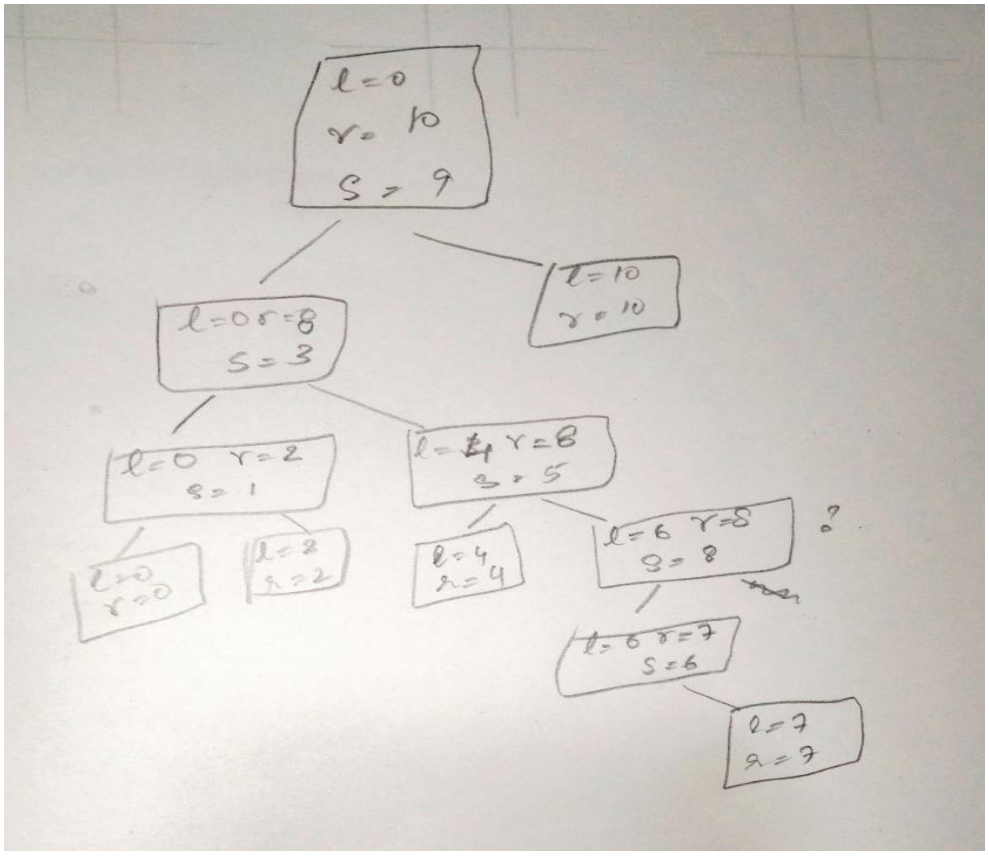
✞ All the split points needs to be clearly written



1.5



1.5M



1M

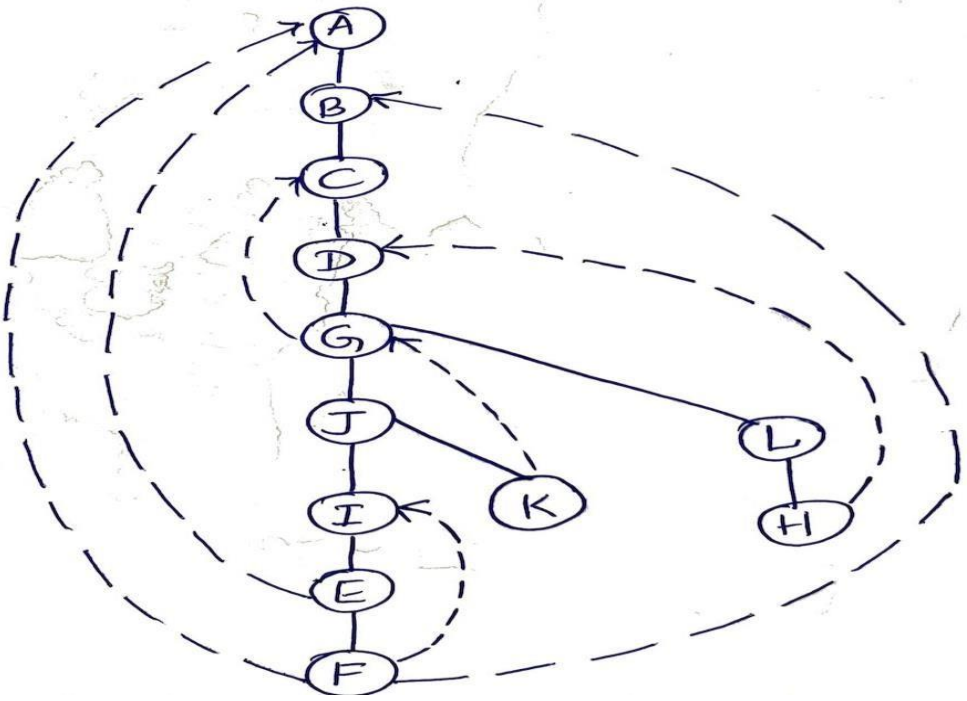
Q14) Push order of vertices - 1 M

A B C D G J I E F K L H

Pop order of vertices - 1 M

F E I K J H L G D C B A

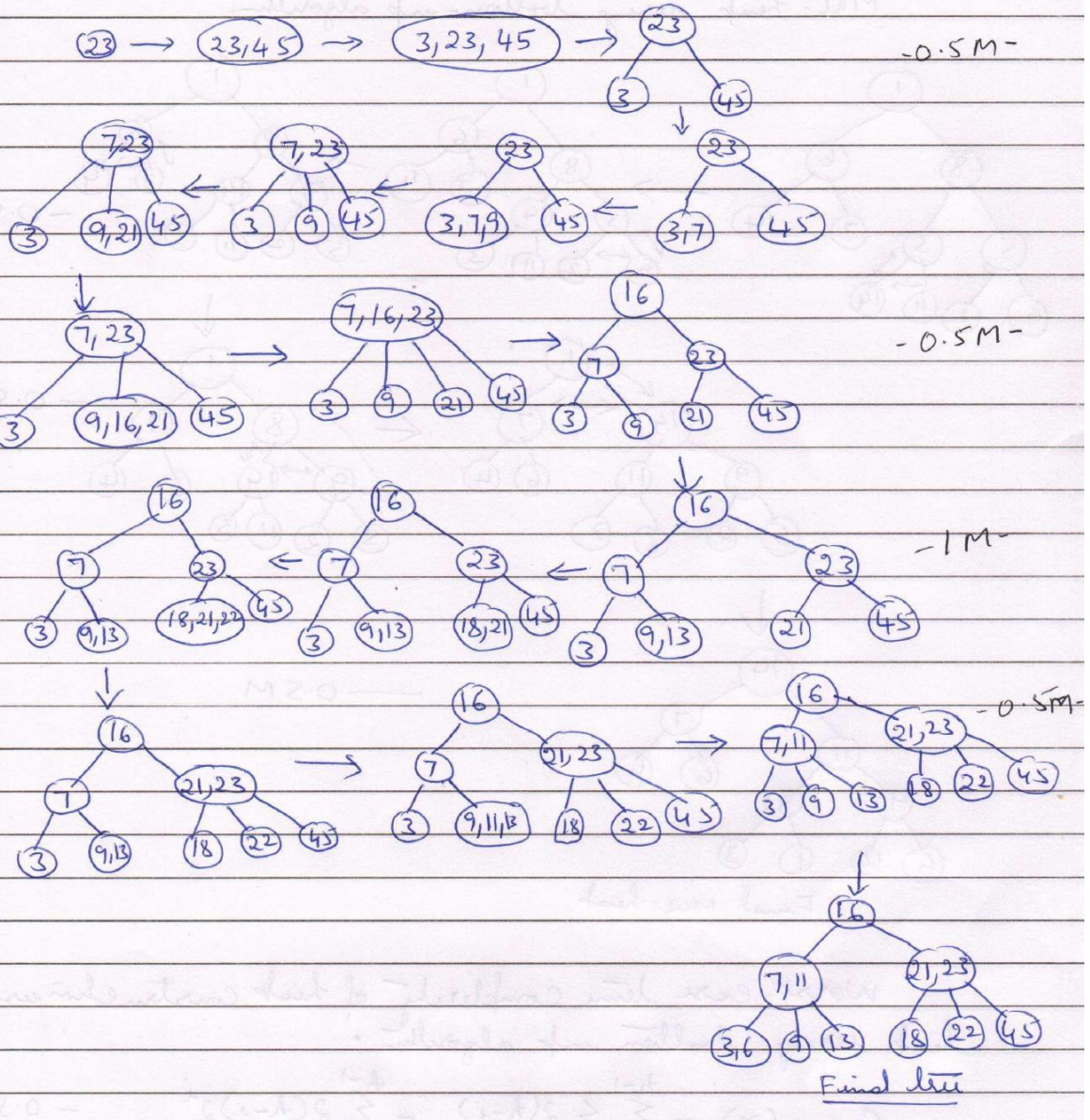
DFS TREE - 1 M



If alphabetical order is not followed marks have been deducted. Whichever part of answer is missing marks are deducted accordingly.

15) 2-3 Tree

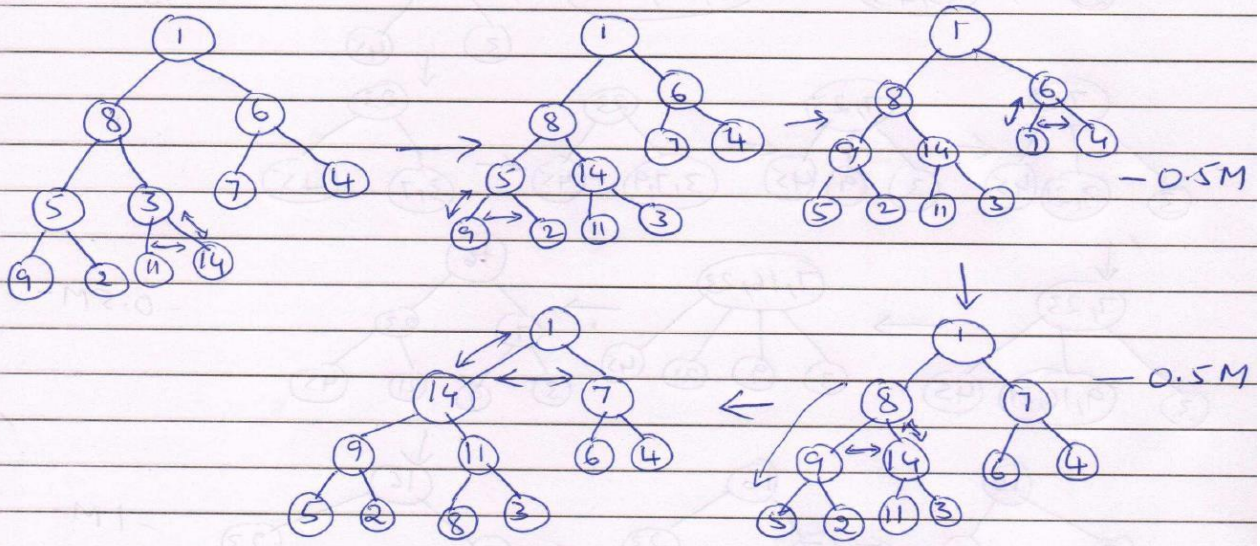
23, 45, 3, 7, 9, 21, 16, 13, 18, 22, 11, 6



The maximum number of comparisons in the full tree of height 3 is = 8 [Every level 2 comparisons × 4 = 8] -0.5M-

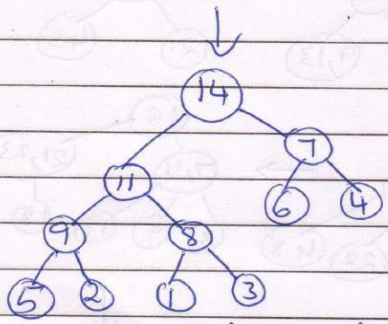
16] 1, 8, 6, 5, 3, 7, 4, 9, 2, 11, 14

Max-heap using bottom-up algorithm



— 0.5M

— 0.5M



— 0.5M

Final max-heap

Worst-case time complexity of heap construction using heap using bottom up algorithm :

$$C_{worst}(n) = \sum_{i=0}^{h-1} \sum_{\text{level } i \text{ keys}} 2(h-i) = \sum_{i=0}^{h-1} 2(h-i)2^i \quad - 0.5M$$

$$= 2\left[\sum_{i=0}^{h-1} h2^i - \sum_{i=0}^{h-1} i \cdot 2^i\right]$$

$$= 2\left[h \sum_{i=0}^{h-1} 2^i - \sum_{i=0}^{h-1} i \cdot 2^i \right]$$

$$= 2\left[h(2^h - 1) - [(h-2)2^h + 2] \right]$$

$$= 2\left[h(2^h - 1) - h2^h + 2 \cdot 2^h - 2 \right]$$

$$= 2\left[h2^h - h - h2^h + 2 \cdot 2^h - 2 \right]$$

$$= 2\left[-h + 2 \cdot 2^h - 2 \right]$$

$$= 2\left[-h + 2^{h+1} - 2 \right]$$

$$= 2\left[-h - 1 + 2^{h+1} - 1 \right]$$

$$\left[\because n = 2^{h+1} - 1 \right] \rightarrow n$$

$$= 2\left[-\log_2(n+1) + n \right]$$

$$= 2\left[n - \log_2(n+1) \right]$$

$$C_{worst}(n) \in O(n)$$

0.5 M

0.5 M

17) Bad symbol shift table.

R	I	M	J	others
5	1	3	2	6

(0.5 M)

Good Suffix shift table

K		
1	RIMJIM	6
2	RIMJIM	3
3	RIMJIM	6
4	RIMJIM	6
5	RIMJIM	6

(0.5 M)

THE-MUSIC-SOUNDS-RIMJIMSE

RIMJIM
(K=0, d_1=6)

RIMJIM
(K=0, d_1=6)

RIMJIM
(K=0 d_1=5)

RIMJIM

[0.5 M]

No. of comparisons = 1+1+1+6 = 09 [0.5M]

Q18)

Input: b, c, d, c, b, a, a, b, c, a, b

A[]

Frequencies	a	b	c	d
	3	4	3	1

Distribution values	a	b	c	d
	3	7	10	11

} 0.5

A[10] = b

D[a..d]

	a	b	c	d
b	3	7	10	11
a	3	6	10	11
c	2	6	10	11
b	2	6	9	11
a	2	5	9	11
a	1	5	9	11
b	0	5	9	11
c	0	4	9	11
d	0	4	8	11
c	0	4	8	10
b	0	4	7	10

S[0..10]

0	1	2	3	4	5	6	7	8	9	10
a	a	a	b	b	b	b	c	c	c	d

1-5 M