ROYAL UNIVERSITY OF BHUATAN
GYALPOZHING COLLEGE OF INFORMATION TECHNOLOGY
GYALPOZHING : BHUTAN

SEMESTER END EXAMINATION (AUTUMN 2020)

Class	:	**Bachelor of Science in Information Technology (Year II, Semester I)**
Module Title	:	**Algorithms and Data Structures**
Module Code	:	**ITS202**
Serial No.	:	**BSc(IT)/2020/III/F/ITS202/I**
Max. Marks	:	**50**
Max. Time	:	**3 Hours**

General Instructions:

1. Question paper has written component.
2. In no circumstances may you remove Answer Books, used or unused, from the Examination Room.
3. If the answer book is torn or folded or without Exam Cell's seal, report the matter to the Invigilator and get a new one.
4. Enter the required details such as Reg. Number, Module and other information as prescribed.
5. Do not write your name on any part of the Answer Book.
6. Number your answer according to the number assigned in the Question Paper.
7. Do not skip any pages when writing answers. Any rough sketches/calculations must be shown on the same page.
8. Do not fold or tear off any pages from the Answer Book. Any answer crossed by you will not be evaluated.
9. You may request for the supplementary Answer sheets only after the main answer Book is completely used.
10. A candidate who is found to have unauthorised materials in his /her possession, copying, talking or exchanging any material with others will be dealt with as per the Wheel of Academic Law.
11. No paper other than Admit Card will be allowed in the Examination Hall/Room unless otherwise specified in the Question Paper.

PART - I [10 Marks]

Answer all the questions

Multiple Choice Questions **[10 x 0.5 = 5]**

Q1) How many times can you tear a phonebook with 128 pages (i.e., sheets of paper) in half, each time throwing away one of the halves, before only one page remains?

A. 6 B. 7 ✓ C. 10 D. 64

Q2) Which strategy returns index of node's left child in heap where i is the position of a node.

A. 2i ✓ B. 2i + 1 C. i D. i/2

Q3) Which of the following is false about a binary search tree?

A. The left child is always lesser than its parent
B. The right child is always greater than its parent
C. The left and right sub-trees should also be binary search trees
D. In order sequence gives decreasing order of elements ✓

Q4) Why we need to a binary tree which is height balanced?

A. to avoid formation of skew trees ✓
B. to save memory
C. to attain faster memory access
D. to simplify storing

Q5) If several elements are competing for the same bucket in the hash table, what is it called?

A. Diffusion
B. Replication
C. Collision ✓
D. Duplication

Q6) The worst case time complexity of BST is

A. O(n) ✓
B. O(log n)
C. O(n log n)
D. O(n^2)

Q7) A simple method for sorting that is effective whenever the keys are small integers.

A. Shell Sort
B. Merge Sort
C. Insertion Sort
D. Key Indexed Counting ✓

Q8) Sorting algorithms which loops through till n-2 is

A. LSD Sort
B. Quick Sortt
C. Bubble Sort ✓
D. Heap Sort

Q9) .... Is the Example of Symbol Table.

A. Strings
B. BSc in IT programme
C. Book
D. DNS Lookup ✓

Q10) Consider the following operation performed on a stack of size 5. Push(1);
Pop();
Push(2);
Push(3);
Pop();
Push(4);
Pop();
Pop();
Push(5);

After the completion of all operation, the no of element present on stack are

A. 1 ✓
B. 2
C. 3
D. 4

Fill in the blanks **[5 x 0.5 = 2.5]**

Q11) ....Skewed........... Binary Tree require more memory space.

Q12) ..LSD......... sorts strings which are of same length.

Q13) A ....negative cycle... is a directed cycle whose sum of edge weights is negative.

Q14) A minimum spanning tree has ....(V – 1).... edges where V is the number of vertices in the given graph.

Q15) Worst-case time complexity is also known as ....Upper bound.....

True or False **[5 x 0.5 = 2.5]**

Q16) A stack is a first-in, first-out data structure. : False **[0.5]**

Q17) Full BT is a strictly binary tree with all leaves in the last level. : True **[0.5]**

Q18) Individual elements in linked list are stored in consecutive memory location. False **[0.5]**

Q19) The worst case time complexity of the insert operation into an AVL tree is O(log n), where n is the number of nodes in the tree. True **[0.5]**

Q20) Heap is sorted in nature. False **[0.5]**

PART - II [15 Marks]

SHORT ANSWER QUESTION

Answer all the questions

Q1) Complete the table below by specifying lower (ω) and upper (O) bounds for each algorithm. Assume that the input to each algorithm is an array of size n. **[0.5 * 5 = 2.5]**

For each row will be evaluated out of **[2* 0.25 = 0.5]**

Algorithms	ω	**O**
Binary Search	1	logn
Bubble Sort	n	n^2
Linear Search	1	n
Merge Sort	n log n	n log n
Selection Sort	n^2	n^2

Q2) What does it mean if some algorithm is in θ (n)? Give example or scenerio of such time complexity. **[2]**

Answer: Algorithms having θ (n) means both the upper bound and lower bound time complexity are equal or same. **[1]**

Example of such time complexity is when counting the number of students in a class. **[0.5]**

Q3) What are the two data structure used in finding SPT(Shortest Path Problem) **[1]**

Answer: The two data structure used in SPT are:

- distTo[v] : is length of shortest path from s to v. **[0.5]**
- edgeTo[v] : is last edge on shortest path from s to v. **[0.5]**

Q4) Explain Hibbard Deletion of BST with the help of example. **[3]**

Answer: To delete a node with key k: search for node t containing key k.

- Case 0. [0 children] Delete t by setting parent link to null.

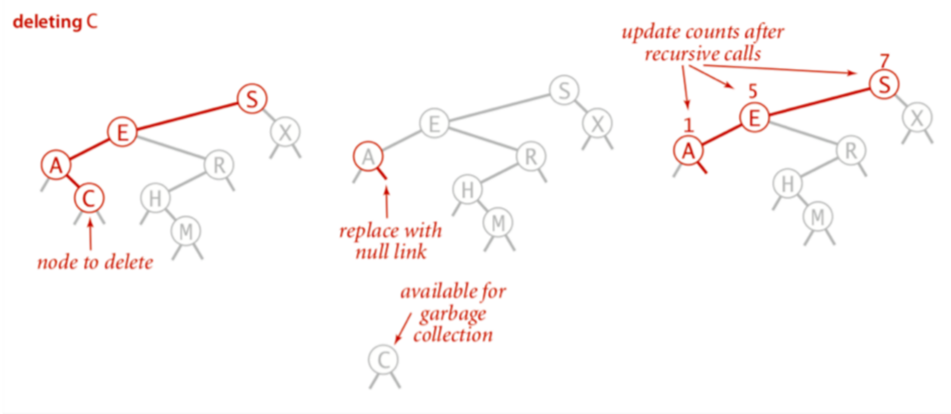


Figure 1: case 0

- Case 1. [1 child] Delete t by replacing parent link.

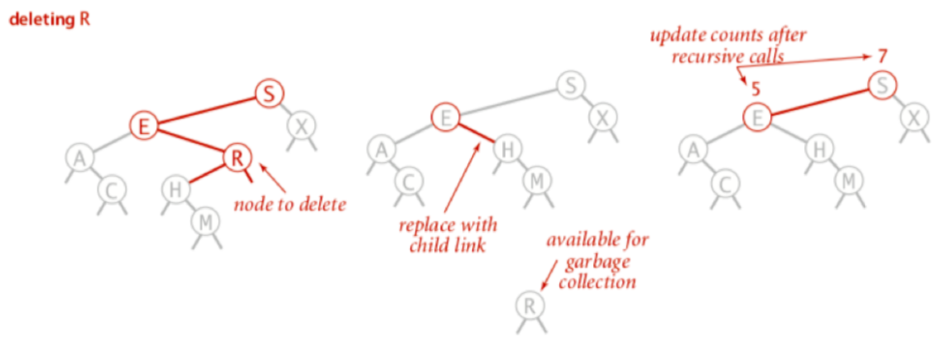


Figure 2: case 1

- Case 2. [2 children]
  - Find successor x of t. <— x has no left child
  - Delete the minimum in t's right subtree. <— but don't garbage collect x
  - Put x in t's spot. <—still a BST

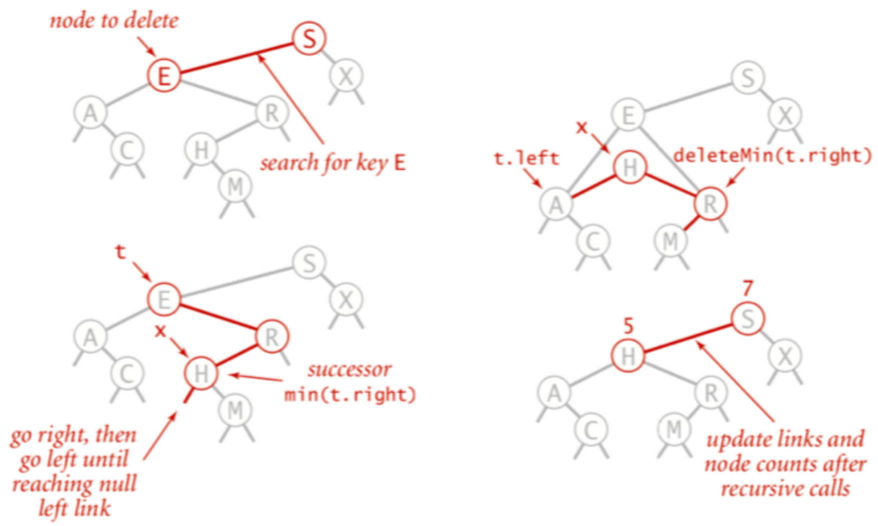


Figure 3: case 2

Q5) For the tree given below, find the preorder, inorder and postorder traversal. **[3]**

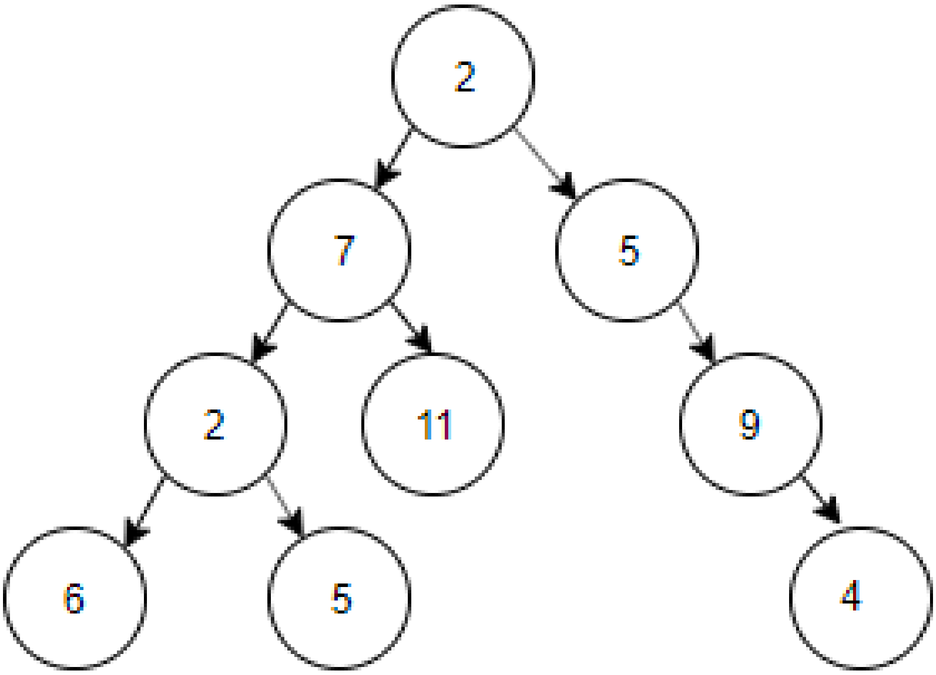


Figure 4: Tree

Answer:

- InOrder: 6, 2, 5, 7, 11, 2, 5, 9, 4 **[1]**
- PreOrder: 2, 7, 2, 6, 5, 11, 5, 9, 4 **[1]**
- PostOrder: 6, 5, 2, 11, 7, 4, 9, 5, 2 **[1]**

Q6) Explain four Operation of Linked List with the help of diagram. **[4]**
Answer: Operations on Linked List:

- Traversing the Nodes. With diagram **[1]**
- Searching for a Node. With Diagram **[1]**
- Prepending Nodes. With Diagram **[1]**
- Removing Nodes. Wiht diagram **[1]**

PART - III [25 Marks]

LONG ANSWER QUESTIONS

Answer all the questions

Q1) Apply shell sort on the given String **[5]**

E A S Y S H E L L S O R T Q U E S T I O N

Answer:

a[ ]

h	i	j	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
			E	A	S	Y	S	H	E	L	L	S	O	R	T	Q	U	E	S	T	I	O	N
13	13	13	E	A	S	Y	S	H	E	L	L	S	O	R	T	Q	U	E	S	T	I	O	N
13	14	14	E	A	S	Y	S	H	E	L	L	S	O	R	T	Q	U	E	S	T	I	O	N
13	15	2	E	A	E	Y	S	H	E	L	L	S	O	R	T	Q	U	S	S	T	I	O	N
13	16	3	E	A	E	S	S	H	E	L	L	S	O	R	T	Q	U	S	Y	T	I	O	N
13	17	17	E	A	E	S	S	H	E	L	L	S	O	R	T	Q	U	S	Y	T	I	O	N
13	18	18	E	A	E	S	S	H	E	L	L	S	O	R	T	Q	U	S	Y	T	I	O	N
13	19	19	E	A	E	S	S	H	E	L	L	S	O	R	T	Q	U	S	Y	T	I	O	N
13	20	20	E	A	E	S	S	H	E	L	L	S	O	R	T	Q	U	S	Y	T	I	O	N
			E	A	E	S	S	H	E	L	L	S	O	R	T	Q	U	S	Y	T	I	O	N

Figure 5: shellsort

4	4	4	E	A	E	S	S	H	E	L	L	S	O	R	T	Q	U	S	Y	T	I	O	N
4	5	5	E	A	E	S	S	H	E	L	L	S	O	R	T	Q	U	S	Y	T	I	O	N
4	6	6	E	A	E	S	S	H	E	L	L	S	O	R	T	Q	U	S	Y	T	I	O	N
4	7	3	E	A	E	L	S	H	E	S	L	S	O	R	T	Q	U	S	Y	T	I	O	N
4	8	4	E	A	E	L	L	H	E	S	S	S	O	R	T	Q	U	S	Y	T	I	O	N
4	9	9	E	A	E	L	L	H	E	S	S	S	O	R	T	Q	U	S	Y	T	I	O	N
4	10	10	E	A	E	L	L	H	E	S	S	S	O	R	T	Q	U	S	Y	T	I	O	N
4	11	7	E	A	E	L	L	H	E	R	S	S	O	S	T	Q	U	S	Y	T	I	O	N
4	12	12	E	A	E	L	L	H	E	R	S	S	O	S	T	Q	U	S	Y	T	I	O	N
4	13	9	E	A	E	L	L	H	E	R	S	Q	O	S	T	S	U	S	Y	T	I	O	N
4	14	14	E	A	E	L	L	H	E	R	S	Q	O	S	T	S	U	S	Y	T	I	O	N
4	15	15	E	A	E	L	L	H	E	R	S	Q	O	S	T	S	U	S	Y	T	I	O	N
4	16	16	E	A	E	L	L	H	E	R	S	Q	O	S	T	S	U	S	Y	T	I	O	N
4	17	17	E	A	E	L	L	H	E	R	S	Q	O	S	T	S	U	S	Y	T	I	O	N
4	18	10	E	A	E	L	L	H	E	R	S	Q	I	S	T	S	O	S	Y	T	U	O	N
4	19	7	E	A	E	L	L	H	E	O	S	Q	I	R	T	S	O	S	Y	T	U	S	N
4	20	8	E	A	E	L	L	H	E	O	N	Q	I	R	S	S	O	S	T	T	U	S	Y
			E	A	E	L	L	H	E	O	N	Q	I	R	S	S	O	S	T	T	U	S	Y

Figure 6: shellsort

1	1	0	A	E	E	L	L	H	E	O	N	Q	I	R	S	S	O	S	T	T	U	S	Y
1	2	2	A	E	E	L	L	H	E	O	N	Q	I	R	S	S	O	S	T	T	U	S	Y
1	3	3	A	E	E	L	L	H	E	O	N	Q	I	R	S	S	O	S	T	T	U	S	Y
1	4	4	A	E	E	L	L	H	E	O	N	Q	I	R	S	S	O	S	T	T	U	S	Y
1	5	3	A	E	E	H	L	L	E	O	N	Q	I	R	S	S	O	S	T	T	U	S	Y
1	6	3	A	E	E	E	H	L	L	O	N	Q	I	R	S	S	O	S	T	T	U	S	Y
1	7	7	A	E	E	E	H	L	L	O	N	Q	I	R	S	S	O	S	T	T	U	S	Y
1	8	7	A	E	E	E	H	L	L	N	O	Q	I	R	S	S	O	S	T	T	U	S	Y
1	9	9	A	E	E	E	H	L	L	N	O	Q	I	R	S	S	O	S	T	T	U	S	Y
1	10	5	A	E	E	E	H	I	L	L	N	O	Q	R	S	S	O	S	T	T	U	S	Y
1	11	11	A	E	E	E	H	I	L	L	N	O	Q	R	S	S	O	S	T	T	U	S	Y
1	12	12	A	E	E	E	H	I	L	L	N	O	Q	R	S	S	O	S	T	T	U	S	Y
1	13	13	A	E	E	E	H	I	L	L	N	O	Q	R	S	S	O	S	T	T	U	S	Y
1	14	10	A	E	E	E	H	I	L	L	N	O	O	Q	R	S	S	S	T	T	U	S	Y
1	15	15	A	E	E	E	H	I	L	L	N	O	O	Q	R	S	S	S	T	T	U	S	Y
1	16	16	A	E	E	E	H	I	L	L	N	O	O	Q	R	S	S	S	T	T	U	S	Y
1	17	17	A	E	E	E	H	I	L	L	N	O	O	Q	R	S	S	S	T	T	U	S	Y
1	18	18	A	E	E	E	H	I	L	L	N	O	O	Q	R	S	S	S	T	T	U	S	Y
1	19	16	A	E	E	E	H	I	L	L	N	O	O	Q	R	S	S	S	S	T	T	U	Y
1	20	20	A	E	E	E	H	I	L	L	N	O	O	Q	R	S	S	S	S	T	T	U	Y
			A	E	E	E	H	I	L	L	N	O	O	Q	R	S	S	S	S	T	T	U	Y

Figure 7: shellsort

Q2) Paro Airport runaway reservation system have only one runway available. Following details are provided for reservation.

- Reserve request: specifies landing time t.
- Add t to the set R of landing times if no other landings are scheduled within k minutes.
- k can vary: let's assume it is statically set (e.g. 3 min).
- After landing, remove request from R.
- What operations do we need in the data structure?
  - Adding requests. If they satisfy constraint!
  - Removing requests.
  - Notion of time, checks every m seconds to update the structure.
  - Nutshell: we need a data structure that allows for insertion and removal of elements.
  - Additional requirement: operations in O(lg n)

List and Explain with diagram all the data structure available comparing interms of their time complexity and finally suggest the best data structure to opt for given you are the developer of this system. **[5]**

Answer:

- Unsorted list/array: good? most operations are in O(n). Insertion can be in O(1) With diagram **[1]**

- Sorted list: Appending and sorting takes O(nlgn) time. Insertion takes O(n) time. A k minute check can be done in O(1) once the insertion point is found. With diagram **[1]**

- Sorted array: Binary search to find place to insert in O(lgn) time. Looks good? Insertion is still in O(n).We almost had it in the sorted list/array. With diagram **[1]**

- Key point: We need fast insertion into a sorted list.
- Binary Search Tree takes O(log n) time complexity. With diagram **[2]**

Q3) Show the AVL tree that results after each of the integer keys9, 27, 50, 15, 2, 21, and 36 are inserted, in that order, into an initially empty AVL tree. Clearly show the tree that results after each insertion, and make clear any rotations that must be performed. **[5]**

Answer:

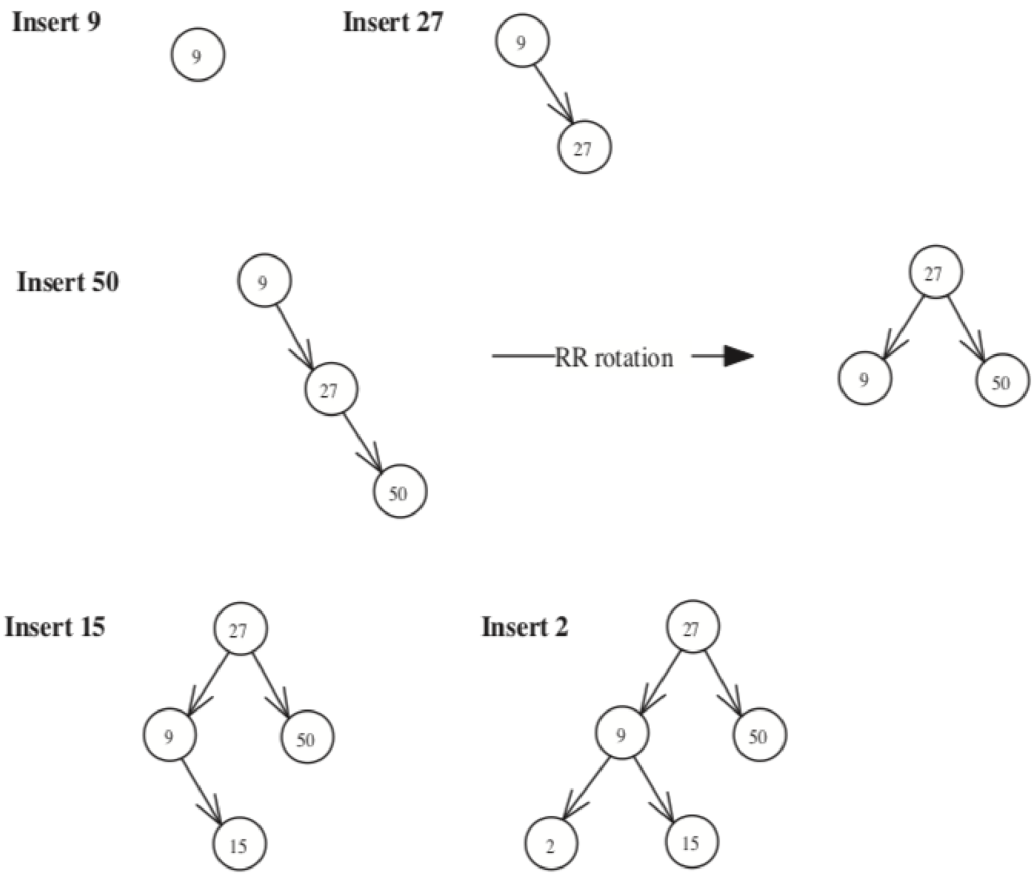


Figure 8: Insertion in AVL Tree_Part1 **[2]**

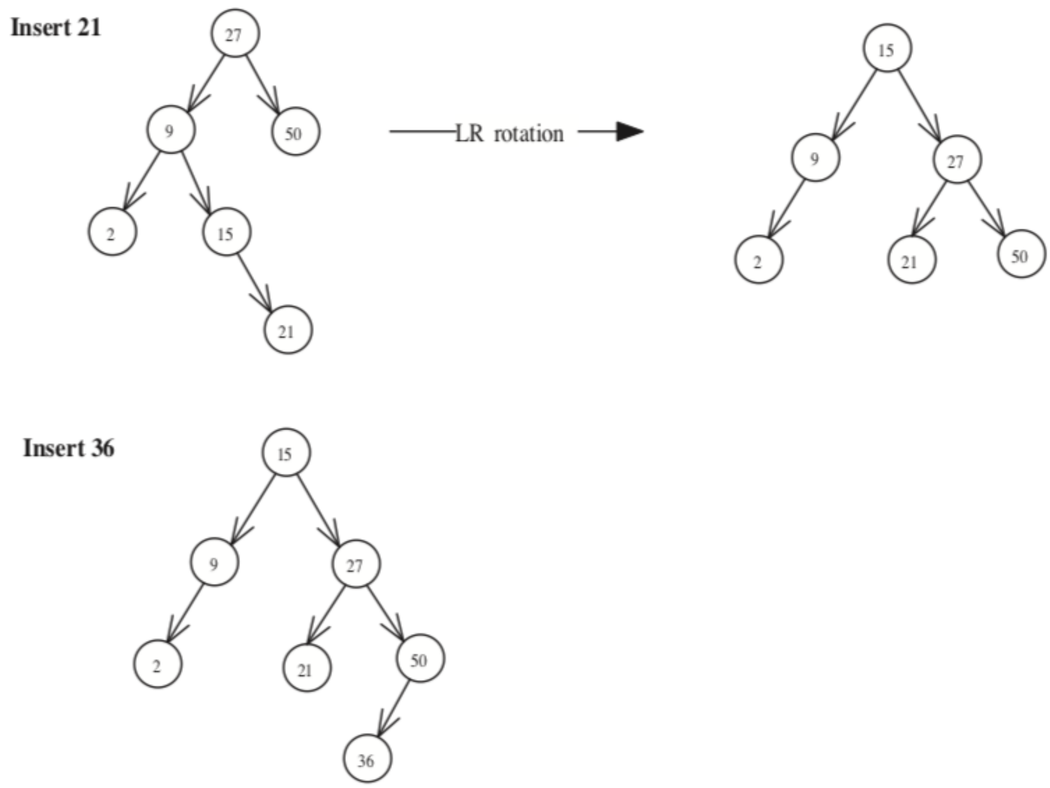


Figure 9: Insertion in AVL Tree_PartII **[3]**

Q4) Find the minimum spanning tree using Kruskal's algorithm and provide the overall weight of the MST. **[5]**

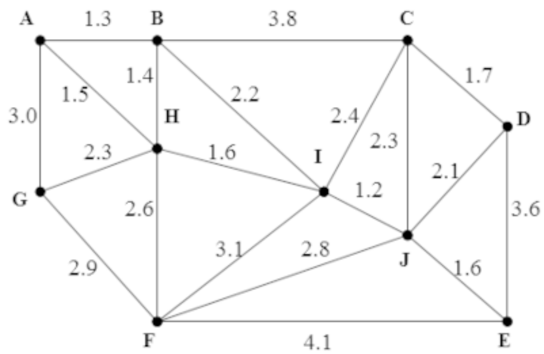


Figure 10: Graph

Answer:

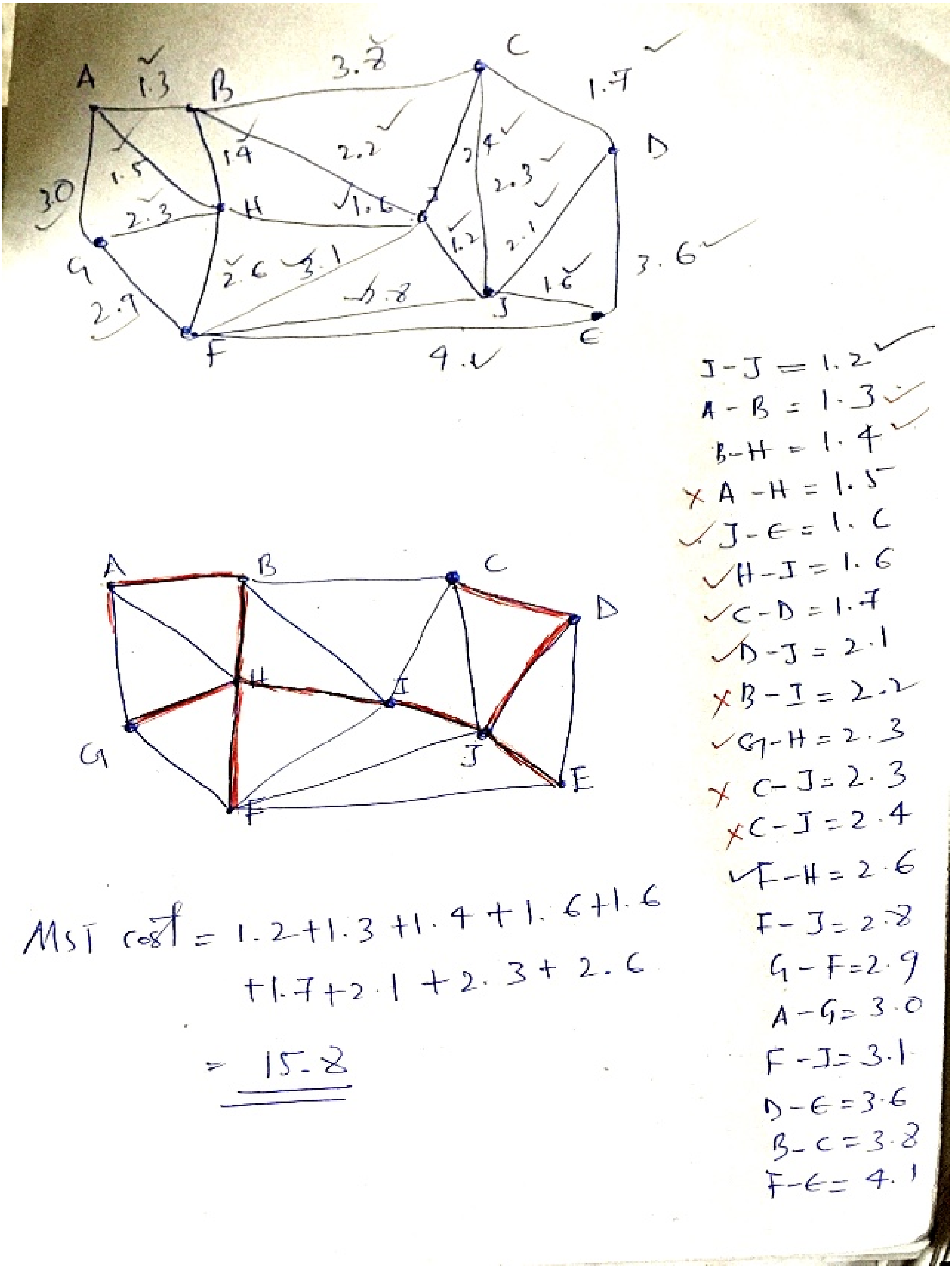


Figure 11: Kruskal MST

Q5) Perform DFS on the graph given. **[5]**

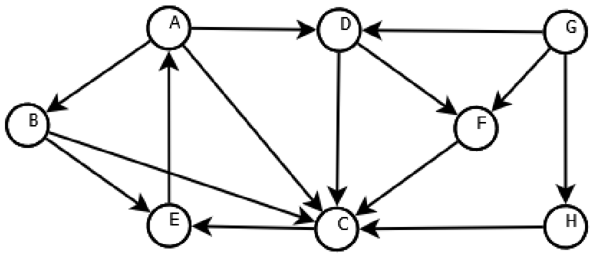
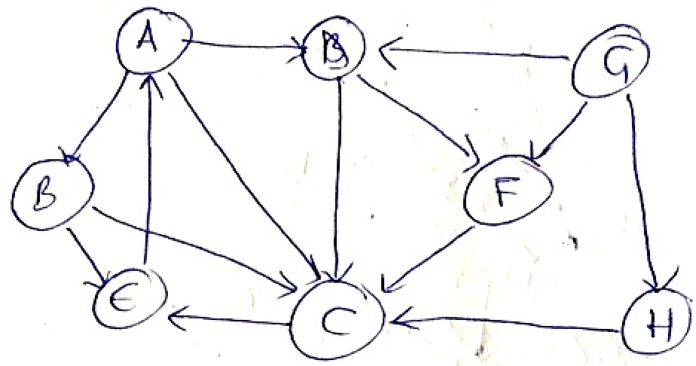


Figure 12: Graph:DFS

Answer:



V	Marked[]	edgeTo[]
A	T	—
B	T	A
C	T	B
D	T	A
E	T	B
F	T	D
G	F	—
H	F	—

Figure 13: DFS Solution