

Question No:1

name of two divide and conquer algorithm (2 marks)

Answer:- (Page 488)

- ❖ Merge Sort
- ❖ Quick Sort
- ❖ Heap Sort

These three algorithms fall under ‘divide and conquer category’.

Question No:2

difference between call by value n call by reference (2 marks)

Answer:- (Page 202)

In case of call by value, a copy of object is made and placed at the time of function calling in the activation record. By using the references, we are not making the copy but send the address of the variable that function can use the original value.

Question No:3

heap and two types of heap(3)

Answer:- (Page 333)

Heap is a data structure of big use and benefit. It is used in priority queue. “The definition of heap is that it is a complete binary tree that conforms to the heap order”. Here heap order means the min and max heap. Max heap: In max heap, each node has a value greater than the value of its left and right child nodes. Moreover, the value of the root will be largest and will become lesser at downward levels. Min heap: in a (min) heap for every node X, the key in the parent is smaller than (or equal to) the key in X. This means that the value in the parent node is less than the value on its children nodes.

Question No:4

height of a tree is 5 find sum of heights (2)

Answer:-

Height = 5

Nodes = $2^h = 2^5 = 32$

Sum of Nodes = $N - (h + 1) = 32 - (5 + 1) = 32 - 6 = 26$

Question No: 5

Here is an array with exactly 15 elements:

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15.

Suppose that we are doing a binary search for an element. Indicate any elements that will be found by examining two or fewer numbers from the array. (Marks: 5)

Answer:- [Click here for detail](#)

If there is an array of 15 elements, {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15}

it wants to know which elements can be discovered by examining two or fewer elements. I want assume the divide and conquer aspect of a binary search, so that leads me to think these elements will be found

1, 2, 14, 15, 7, 3

rather; 1, 2, 3, 4, 6, 8, 14, 15

1 and 15 for heads and tails

2, and 14 for next elements (following or preceding)

1/2 of 15 is either 6, or 8, from 7.5

and 1/2 of 6 = 3

1/2 8 = 4

Q1 Write one example of hashing?

Answer: (page 459)

Hashing can be used to store the data. Suppose we have a list of fruits. The names of Fruits are in string. The key is the name of the fruit. We will pass it to the hash Function to get the hash key.

HashCode ("apple") = 5

We pass it to the string "apple". Resultantly, it returns a number 5.

Q2 Height of a tree is 5 find sum of heights

Answer:- rep

Q3 How do we carry out degeneration of Complete Binary Tree?

Answer: [Click here for detail](#)

Degeneration of the (original) binary tree is mainly reflected by degeneration of the internal infix trees and less

by degeneration of the external height of the PPbin tree.

Q4 What is a skip List?

Answer:- (Page 446)

A skip list for a set S of distinct (key, element) items is a series of lists S_0, S_1, \dots, S_h such that

- Each list S_i contains the special keys $+\infty$ and $-\infty$
- List S_0 contains the keys of S in non-decreasing order
- Each list is a subsequence of the previous one, i.e.,
$$S_0 \supseteq S_1 \supseteq \dots \supseteq S_h$$
- List S_h contains only the two special keys

Q5 How we can generate a maze with the help of union

Answer:- (Page 424)

How can we generate maze? The algorithm is as follows:

- ❖ Randomly remove walls until the entrance and exit cells are in the same set.
- ❖ Removal of the wall is the same as doing a union operation.
- ❖ Do not remove a randomly chosen wall if the cells it separates are already in the same set.

Q7 Following is an array with exactly 15 elements:

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15.

Suppose that we are doing a binary search for an element. Indicate any elements that will be found by examining two or fewer numbers from the array. 5

Answer:- Repeat

(Marks: 10)

Write down the C++ code to implement insertion sort algorithm.

Answer:- (Page 483)

Following is the code of the insertion sort in C++.

```
void insertionSort(int *arr, int N)
{
int pos, count, val;
for(count=1; count < N; count++)
{
val = arr[count];
for(pos=count-1; pos >= 0; pos--)
if (arr[pos] > val)
arr[pos+1]=arr[pos];
else break;
arr[pos+1] = val;
}
```

}
}

Q2. What is Table ADT? Discuss any two implementations of table ADT. (Marks: 5)

Answer:- (Page 427, 428)

The table, an abstract data type, is a collection of rows and columns of information.

- Implementation of Table
- o Unsorted Sequential Array
 - o Sorted Sequential Array

Question No: 3

Suppose we have the following representation for a complete Binary Search Tree, tell the Left and Right child nodes and Parent node of the node D (Marks: 5)

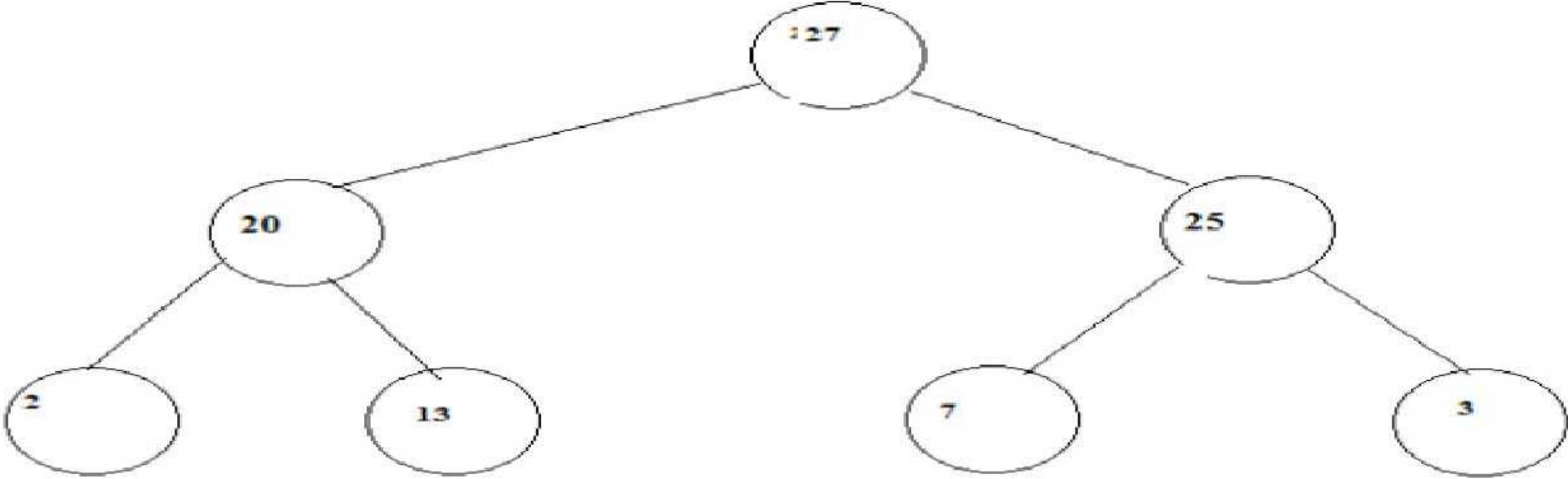
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	...
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	...

Answer:-

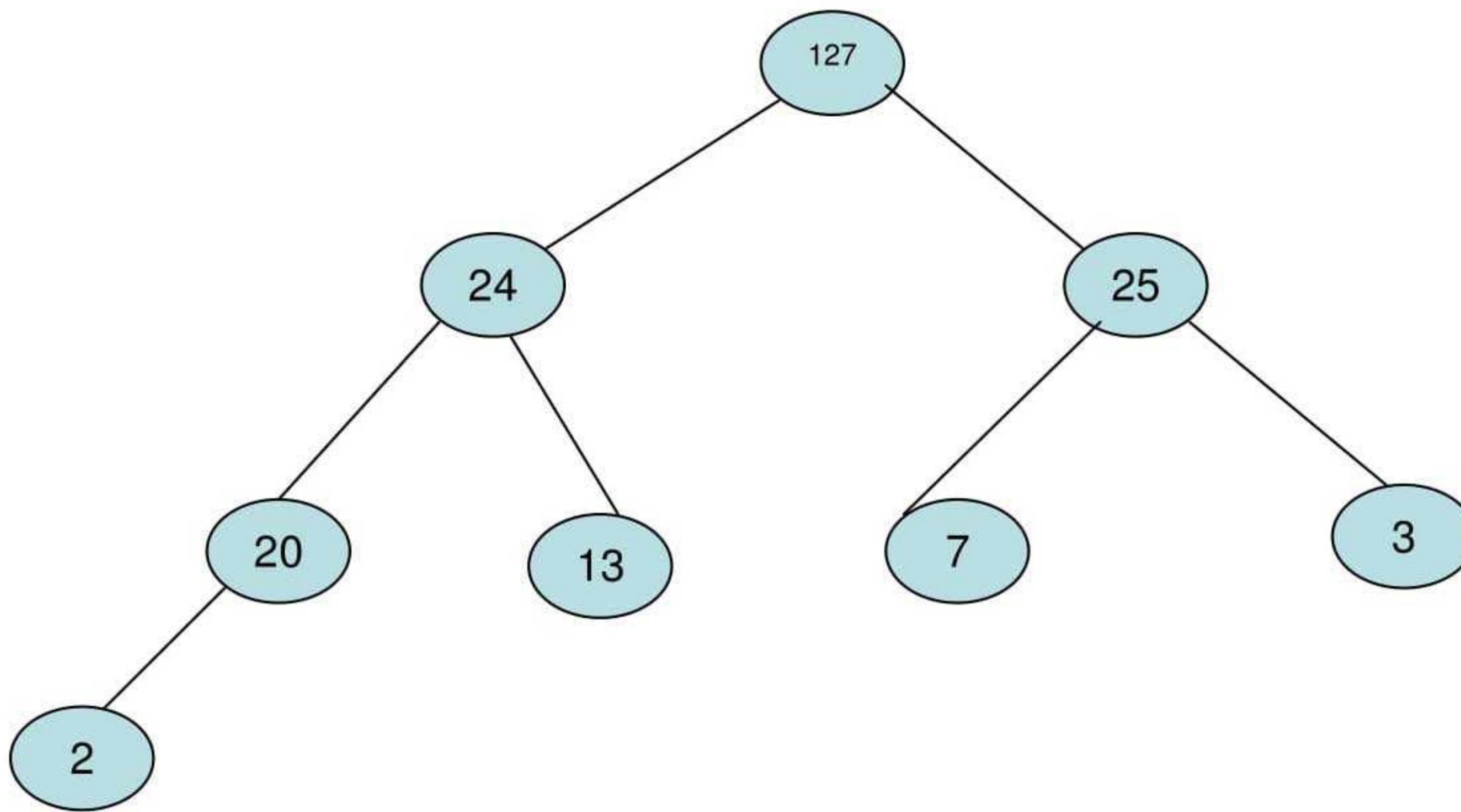
- parent: B using formula($i/2$)
- left child:-H $2i=2*4=8=H$
- Right child:-I $2i+1=8+1=9=I$

Question No: 4

.Consider the following max heap. Add node 24 in it and show the resultant Heap. (Marks: 3)



Answer:-



Question No: 5 (Marks: 3)

Here is an array of 10 integers.

5 3 8 1 7 0 2 6 4.

Draw this array after the first iteration of the large loop in a selection sort (Sorting from smallest to largest)

Answer:-

	5	3	8	9	1	7	0	2	6	4
1st step	0	3	8	9	1	7	5	2	6	4
2nd step	0	1	8	9	3	7	5	2	6	4
3rd step	0	1	2	9	3	7	5	8	6	4
4th step	0	1	2	3	9	7	5	8	6	4
5th step	0	1	2	3	4	7	5	8	6	9
6th step	0	1	2	3	4	5	7	8	6	9
7th step	0	1	2	3	4	5	6	8	7	9
8th step	0	1	2	3	4	5	6	7	8	9

Question No: 6 (Marks: 3)

What is an Equivalent relation? Give any two examples.

Answer : (page 387)

‘A binary relation R over a set S is called an *equivalence relation* if it has following properties’:

1. Reflexivity: for all element $x \in S$, $x R x$
2. Symmetry: for all elements x and y , $x R y$ if and only if $y R x$
3. Transitivity: for all elements x , y and z , if $x R y$ and $y R z$ then $x R z$

Example; sets of peoples ,electric circuit domain

Q7. What are the properties of equivalence class? 3marks

Answer:- (Page 385)

A binary relation R over a set S is called an *equivalence relation* if it has following Properties

1. Reflexivity: for all element $x \in S$, $x R x$
2. Symmetry: for all elements x and y, $x R y$ if and only if $y R x$
3. Transitivity: for all elements x, y and z, if $x R y$ and $y R z$ then $x R z$

Q8. How heap sort works to set a set of data. 2Marks

Answer:- (Page 334)

Min heap and max heap are used to sort the data. Min heap convert the data in ascending sorted order and max heap convert the in descending order.

Q9. How can we search an element in skip list? 2Marks

Answer:- (Page 447)

We search for a key x in the following fashion:

We start at the first position of the top list

- At the current position p, we compare x with $y \square \text{key}(\text{after}(p))$
- $x = y$: we return $\text{element}(\text{after}(p))$
- $x > y$: we “scan forward”
- $x < y$: we “drop down”
- If we try to drop down past the bottom list, we return NO_SUCH_KEY

Q10. What is an ADT?

Answer:- (Page 428)

The table, an abstract data type, is a collection of rows and columns of information.

Question No: 27 (Marks: 2)

Give one example of Hashing

Answer:- Repeated

Question No: 28 (Marks: 2)

How heap sort works to sort a set of data.

Answer:- Repeated

Question No: 29 (Marks: 2)

How we can implement Table ADT using Linked?

Answer:- (Page 441)

List linked list is one choice to implement table abstract data type. For unsorted elements, the insertion at *front* operation will take constant time. But if the data inside the list is kept in sorted order then to insert a new element in the list, the entire linked list is traversed through to find the appropriate position for the element.

Question No: 30 (Marks: 2)

If we allow assignment to constants what will happen?

Answer:- (Page 8)

Suppose 'a' has the value number 3. Now we assigned number 2 the number 3 i.e. all the number 2 will become number 3 and the result of $2 + 2$ will become 6. Therefore it is not allowed.

Question No: 31 (Marks: 3)

Explain the process of Deletion in a Min-Heap

Answer:- (Page 350)

We have to write a function *deleteMin()*, which will find and delete the minimum number from the tree. Finding the minimum is easy; it is at the top of the heap.

Question No: 32 (Marks: 3)

Give any three characteristics of Union by Weight method.

Answer:- (Page 408)

Following are the salient characteristics of this method:

- ❖ Maintain sizes (number of nodes) of all trees, and during *union*.
- ❖ Make smaller tree, the sub-tree of the larger one.
- ❖ Implementation: for each root node *i*, instead of setting *parent[i]* to -1, set it to -*k* if tree rooted at *i* has *k* nodes.

Question No: 33 (Marks: 3)

"For smaller lists, linear insertion sort performs well, but for larger lists, quick sort is suitable to apply." Justify why?

Answer:-

Since for smaller lists number of comparisons and interchange (if needed) decreases radically in insertion sort than quick sort. For larger lists, opposite occurs in both sorting techniques.

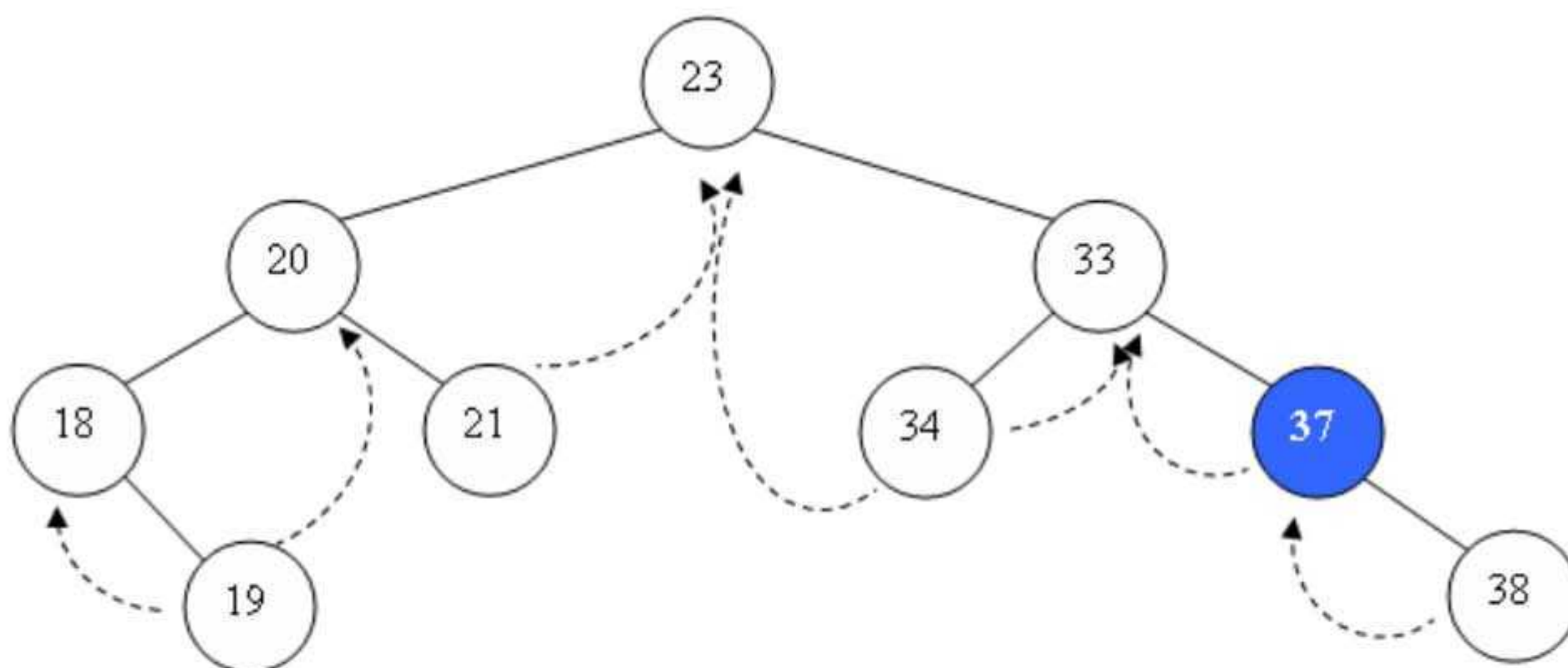
Question No: 34 (Marks: 5)

Write down the C++ code to implement Insertion Sort Algorithm.

Answer:- repeat

Question No: 35 (Marks: 5)

Consider the following Threaded Binary Tree,



You have to give the values that should be in the four variables given below, for the node 37

1. LTH (Left flag)
2. RTH (Right flag)
3. Left node pointer (->L)
4. Right node pointer (->R)

Question No: 36 (Marks: 5)

What is Disjoint Sets? Explain with an example.

Answer:- (Page 388)

Disjoint sets - Two sets are disjoint if they have no elements in common.

Example:-Suppose there are many people around you. How can we separate those who are related to each other in this gathering?

We make groups of people who are related to each other. The people in one group will say that they are related to each other. Similarly we can have many groups. So every group will say that they are related to each other with family relationship and there is no group that is related to any other group. The people in the group will say that there is not a single person in the other group who is related to them.

Question No: 27 (Marks: 2)

Give the difference between strict and complete binary tree.

Answer:- (Page 121,123)

A binary tree said to be a strictly binary tree if every non-leaf node in a binary has non-empty left and right sub trees. If there are left and right sub trees for each node in the binary tree as shown a complete binary tree.

Question No: 28 (Marks: 2)

A complete binary tree can be stored in an array. While storing the tree in an array we leave the first position (0th index) of the array empty. Why?

Answer:- (page 325)

In case of having a node with left and right children, stored at position i in the array, the left child will be at position $2i$ and the right child will be at $2i+1$ position. If the value of i is 2, the parent will be at position 2 and the left child will be at position $2i$ i.e. 4. The right child will be at position $2i+1$ i.e. 5. we have not started the 0^{th} position. It is simply due to the fact if the position is 0, $2i$ will also become 0. So we will start from the 1^{st} position, ignoring the 0^{th} .

Question No: 29 (Marks: 2)

Give the name of two Divide and Conquer algorithms.

Answer:- Repeated

Question No: 30 (Marks: 2)

Give the effect of sorted data on Binary Search.

Answer:- <http://www.cprogramming.com/tutorial/lesson18.html>

The binary tree is a fundamental data structure used in computer science. The binary tree is a useful data structure for rapidly storing sorted data and rapidly retrieving stored data.

Question No: 31 (Marks: 3)

Give any three characteristics of Union by Weight method.

Answer:- Repeated

Question No: 32 (Marks: 3)

Here is an array of ten integers:

5 3 8 9 1 7 0 2 6 4

Draw this array after the FIRST iteration of the large loop in an insertion sort (sorting from smallest to largest). This iteration has shifted at least one item in the array!

Answer:-

	5	3	8	9	1	7	0	2	6	4
1 st step	3	5	8	9	1	7	0	2	6	4
2 nd step	1	3	5	8	9	7	0	2	6	4
3 rd step	1	3	5	7	8	9	0	2	6	4
4 th step	0	1	3	5	7	8	9	2	6	4
5 th step	0	1	2	3	5	7	8	9	6	4
6 th step	0	1	2	3	5	6	7	8	9	4
7 th step	0	1	2	3	4	5	6	7	8	9

Question No: 33 (Marks: 3)

Give your comment on the statement that heap uses least memory in array representation of binary trees. Justify your answer in either case.

Answer:-

A heap is a complete binary tree, so it is easy to be implemented using an array representation

It is a binary tree with the following properties:-

- ❖ Property 1: it is a complete binary tree
- ❖ Property 2: the value stored at a node is greater or equal to the values stored at the children
- ❖ From property 2 the largest value of the heap is always stored at the root

Question No: 34 (Marks: 5)

Suppose we have the following representation for a complete Binary Search Tree, tell the Left and Right child nodes and Parent node of the node D

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	...
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	...

Answer: - Repeated

Question No: 35 (Marks: 5)

Explain the following terms:

- 1. Collision**
- 2. Linear Probing**
- 3. Quadratic Probing**

Answer:- (Page 464-470)

Collision:

When two values hash to the same array location, this is called a *collision*.

Linear Probing

When there is a collision, some other location in the array is found. This is known as linear probing. In linear probing, at the time of collisions, we add one to the index and check that location. If it is also not empty, we add 2 and check that position. Suppose we keep on incrementing the array index and reach at the end of the table. We were unable to find the space and reached the last location of the array.

Quadratic Probing

In the quadratic probing when a collision happens we try to find the empty location at index + 1^2 . If it is filled then we add 2^2 and so on.

Quadratic probing uses different formula:

- ^{1.} Use $F(i) = i^2$ (square of i) to resolve collisions
- ^{2.} If hash function resolves to H and a search in cell H is inconclusive, try $H + 1^2, H + 2^2, H + 3^2$

Question No: 36 (Marks: 5)

Here is an array with exactly 15 elements:

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15.

Suppose that we are doing a binary search for an element. Indicate any elements that will be found by examining two or fewer numbers from the array.

Answer:- Repeat

Question No: 28 (Marks: 2)

What are different applications of Hashing?

Answer (page 474)

- ❖ Compilers use hash tables to keep track of declared variables (symbol table).
- ❖ A hash table can be used for on-line spelling checkers — if misspelling detection (rather than correction) is important, an entire dictionary can be hashed and words checked in constant time.
- ❖ Game playing programs use hash tables to store seen positions, thereby saving computation time if the position is encountered again.

Question No: 29 (Marks: 2)

Give the operation names that we can perform on Table abstract data type.

Answer:- (page 429)

Insert
Find
Remove

Question No: 30 (Marks: 2)

Give your comments on the statement:

"Efficiently developed data structures decrease programming effort"

Answer:- [Click here for detail](#)

Reduces programming effort by providing useful data structures and algorithms so you don't have to write them yourself.

Increases performance by providing high-performance implementations of useful data structures and algorithms. Because the various implementations of each interface are interchangeable, programs can be easily tuned by switching implementations.

Question No: 31 (Marks: 3)

When Hashing is NOT suitable?

Answer: - (Page 475)

Hash tables are not so good if there are many insertions and deletions, or if table traversals are needed — in this case, AVL trees are better.

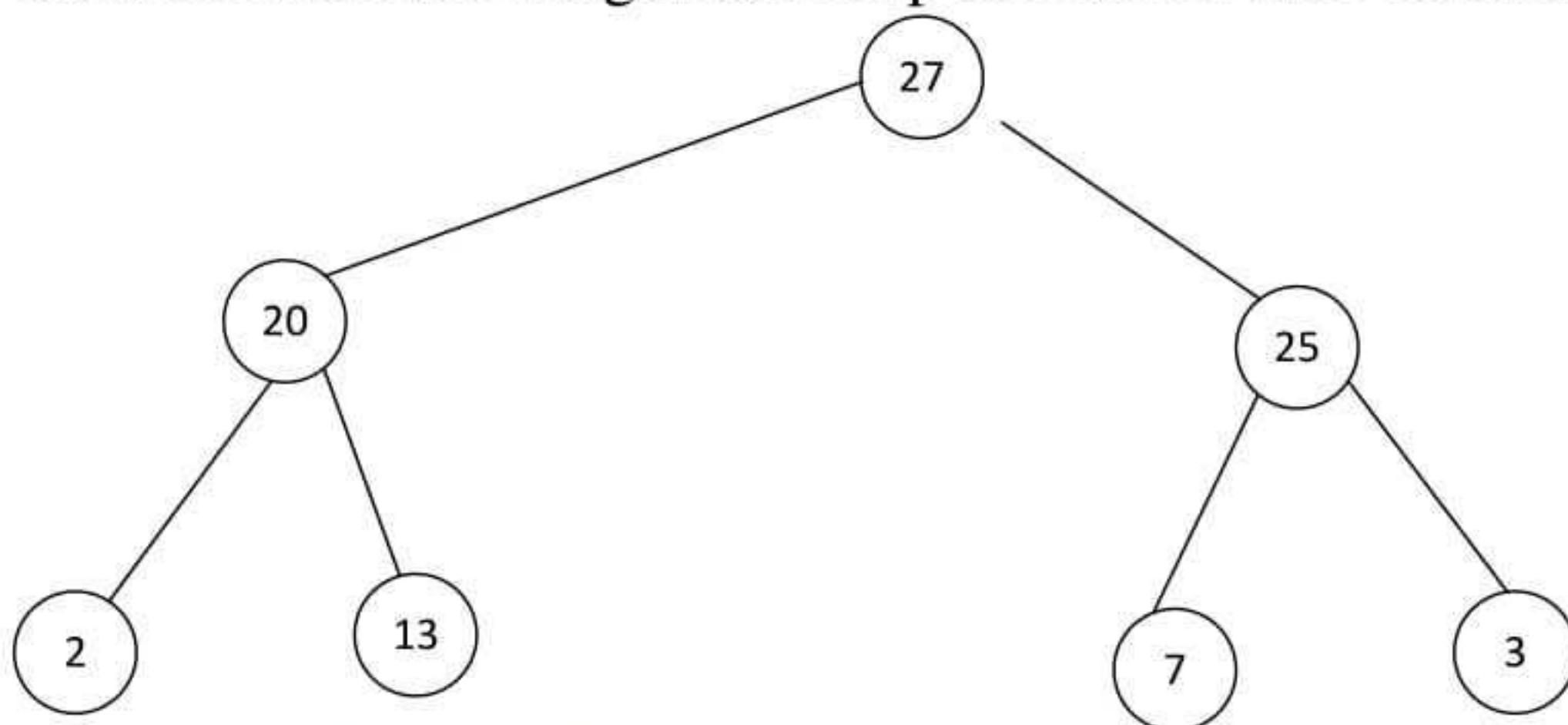
Question No: 32 (Marks: 3)

Give any three characteristics of Union by Weight method.

Answer: - Repeated

Question No: 33 (Marks: 3)

Consider the following Max Heap add node 24 in it and show the resultant Heap,



Answer: - Repeated

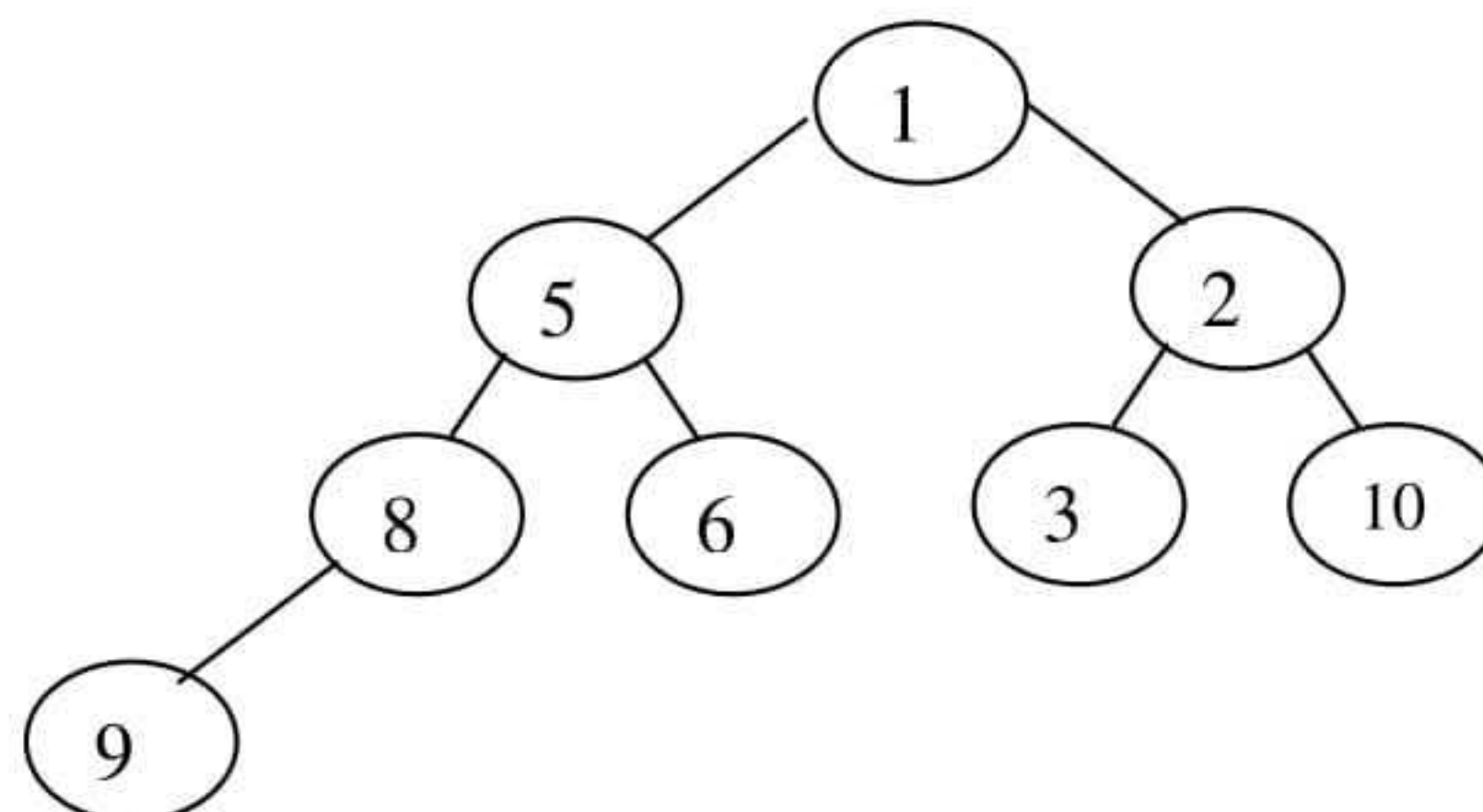
Question No: 34 (Marks: 5)

Heapify the elements of the following array (reading from left to right) into a Min Heap and show that Min Heap contents in the form of array as shown below,

original array	6	5	3	9	1	2	10	8	-
----------------	---	---	---	---	---	---	----	---	---

Answer(page 334)

Min heap



Heapified array	1	5	2	8	6	3	10	9	
------------------------	---	---	---	---	---	---	----	---	--

Question No: 35 (Marks: 5)

Here is an array of ten integers:

5 3 8 9 1 7 0 2 6 4

Show the first three merging steps for Merge sort on this array.

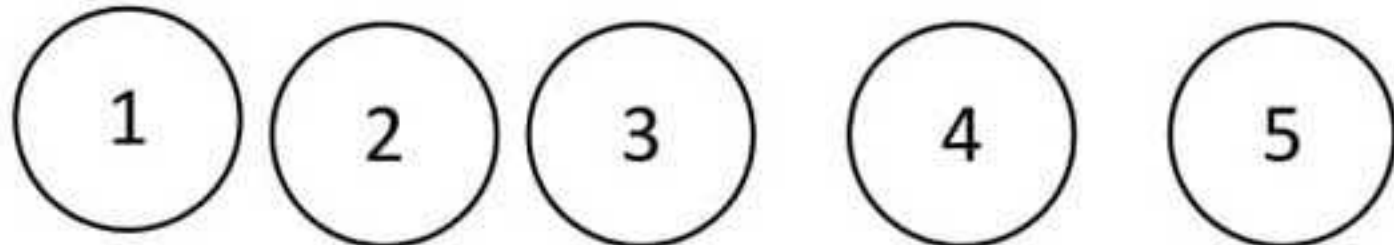
Answer:-

	5	3	8	9	1	7	0	2	6	4
1 st step	5	3	8	9	1	7	0	2	6	4
2 nd step	3	5	8	9	1	7	0	2	4	6
3 rd step	3	5	8	9	1	7	0	2	4	6
4 th step	1	3	8	5	9	2	6	7	0	4
5 th step	1	3	8	5	9	2	6	7	0	4
6 th step	1	2	3	6	7	8	0	5	4	9

Question No: 36 (Marks: 5)

Consider the following sequence of union commands on the set of elements

{1,2,3,4, 5}:



union(4,2)

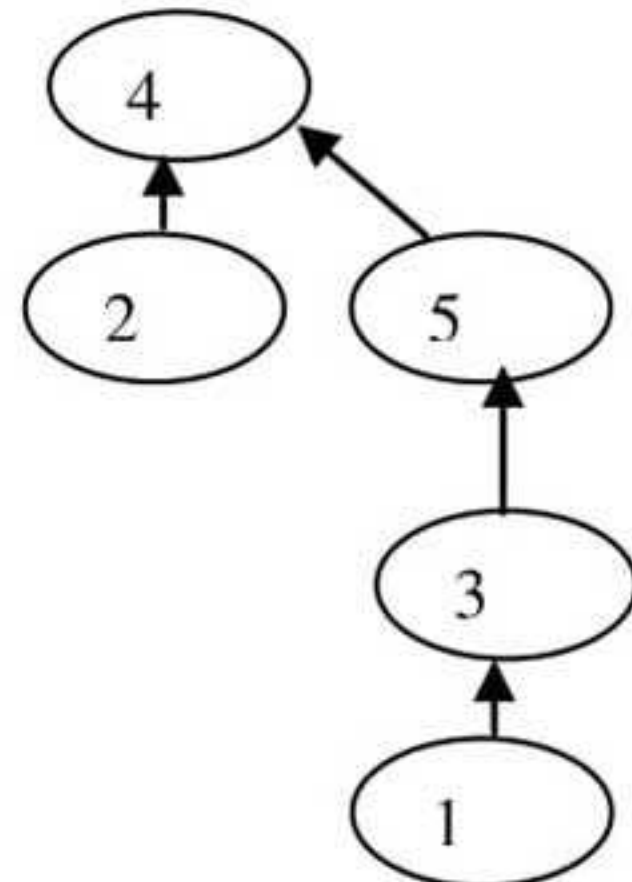
union(3,1)

union(5,4)

union(5,3)

Show the result when the unions are performed

Answer: (Page 396 to398)



Question No: 27 (Marks: 2)

Let's call the node as a that requires re-balancing. Consider two cases below:

- 1) An insertion into right subtree of the left child of a.
- 2) An insertion into left subtree of the right child of a.

Which of the following statement is correct about the above two cases.

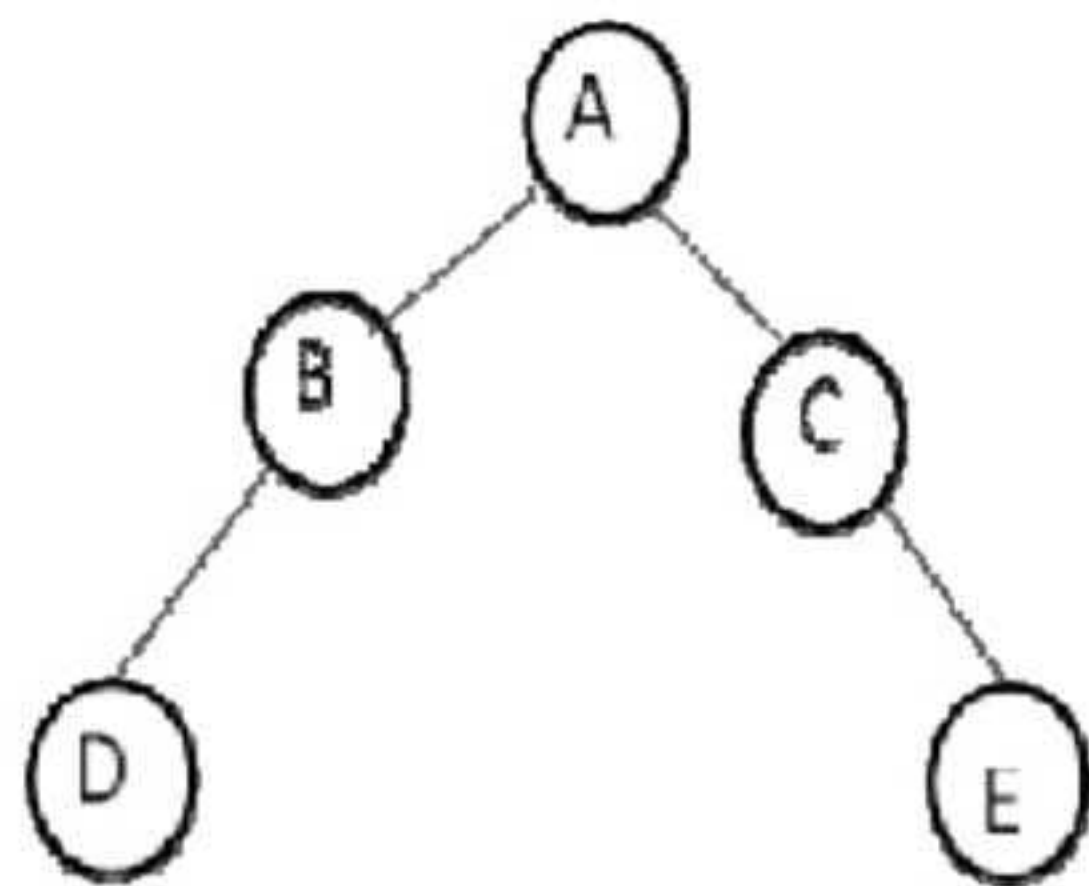
- 1) The insertion occurs outside (i.e., right to left or left to right) in the above two cases, single rotation can fix the balance in these two cases.
- 2) The insertion occurs inside ((i.e., right to left or left to right) in these two cases, single rotation cannot fix the balance in these two cases.

Answer: (page 230)

Insertion occurs on the *inside* in cases 2 and 3 which a single rotation cannot fix.

Question No: 28 (Marks: 2)

Represent the following Binary tree using array representation.



Answer:-

A	B	C	D	E
---	---	---	---	---

Question No: 29 (Marks: 2)

Give one example of Hashing

Answer:- repeat

Question No: 30 (Marks: 2)

How we can apply Find operation on elements combined through Union operation.

Answer:- (Page 392)

We give an element to the *find* method and it returns the name of the set.

Question No: 31 (Marks: 3)

Here is an array of ten integers:

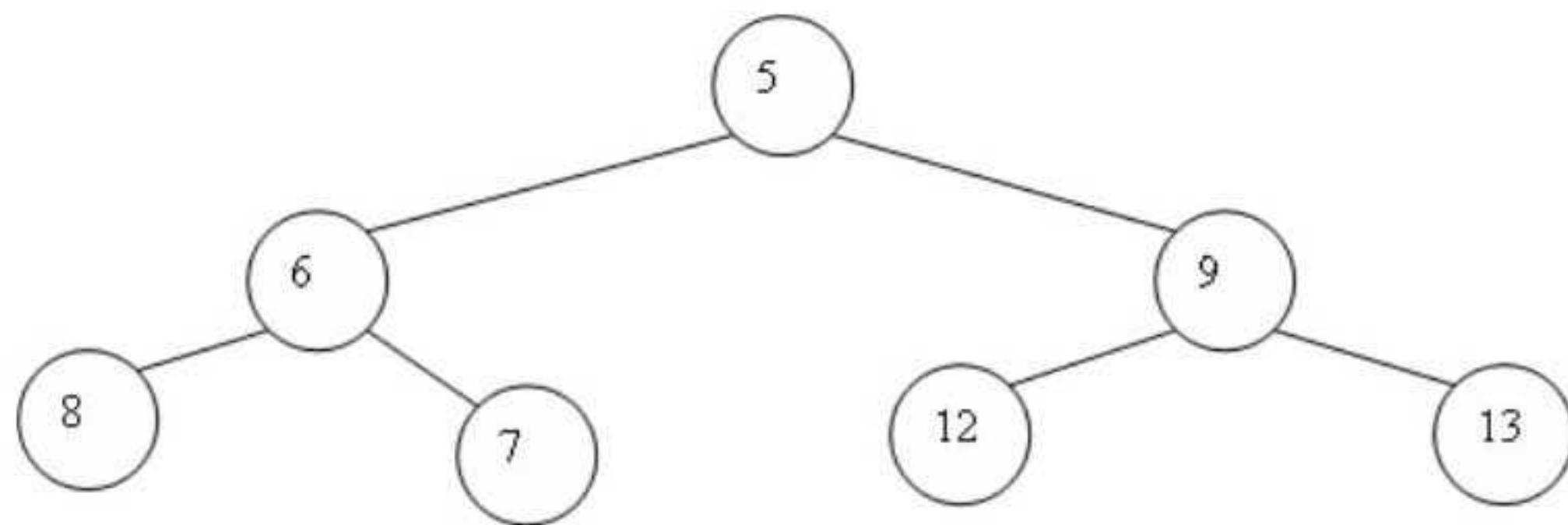
5 3 8 9 1 7 0 2 6 4

Draw this array after the FIRST iteration of the large loop in an insertion sort (sorting from smallest to largest). This iteration has shifted at least one item in the array!

Answer:- repeat

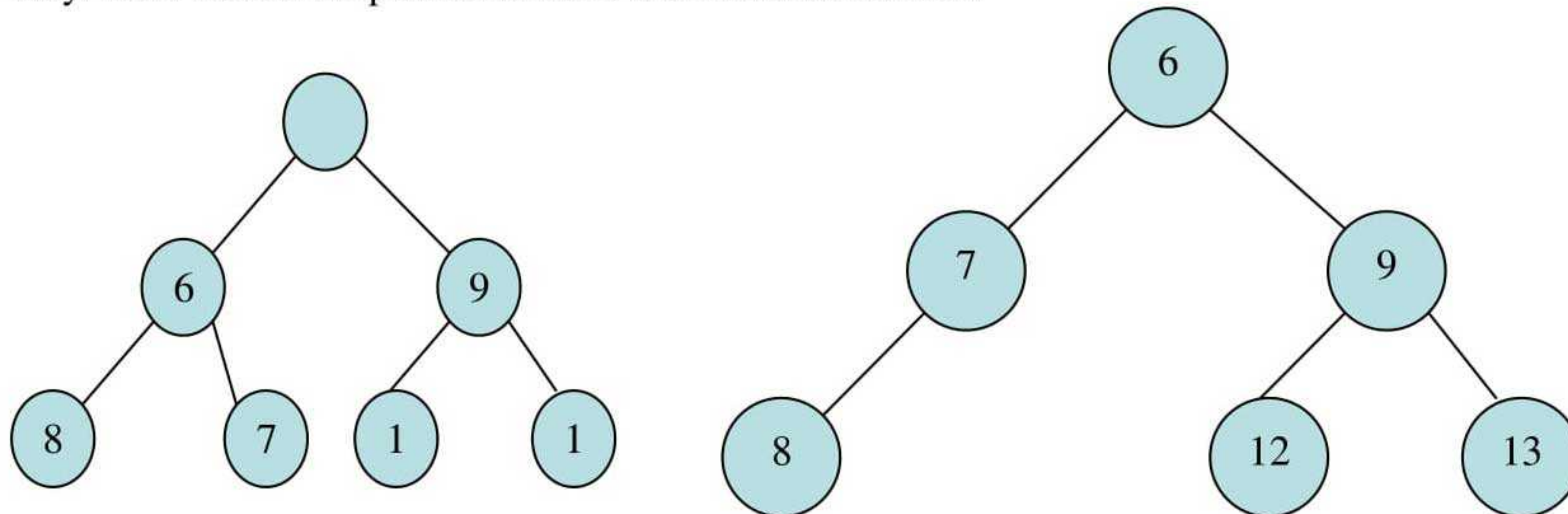
Question No: 32 (Marks: 3)

Consider the following Min Heap apply operation delMin on it and show the resultant Heap,



Answer:- (page 349)

It is clear from the definition of min-heap that it should be in the *root* of the tree. So finding the minimum is very easy. Here node 5 is operational node that has to be deleted.



Question No: 33 (Marks: 3)

Which implementation of disjoint set is better with respect space and why?

Answer:- (page 405)

An array that is a single dimensional structure and seems better with respect to space. Suppose we have 1000 set members i.e. names of people. If we make a Boolean matrix for 1000 items, its size will be 1000 x 1000. Thus we need 1000000 locations for Boolean items, its size will be 1000 x 1000. Thus we need 1000000 locations for Boolean values. in case of an array the number of location will be 1000. Thus this use of array i.e. tree like structure is better then two dimensional arrays in term of space to keep disjoint sets and doing union and find operation.

Question No: 34 (Marks: 5)

What is Table ADT? Discuss any two implementations of table ADT.

Answer (page 427,728)

The table, an abstract data type, is a collection of rows and columns of information.

Implementation of Table
 o Unsorted Sequential Array
 o Sorted Sequential Array

Question No: 35 (Marks: 5)

Suppose the following numbers are stored in an array 'BS':

32,51,27,85,66,23,13,57

We apply Bubble sort to array 'BS'. Show step by step sorting. You have to write only two passes that is passing 1 and pass 2.

Hint; while concern step by step sorting there are 6 passes involved to sort array nums. You have to show all passes

Answer:-

Pass one:-

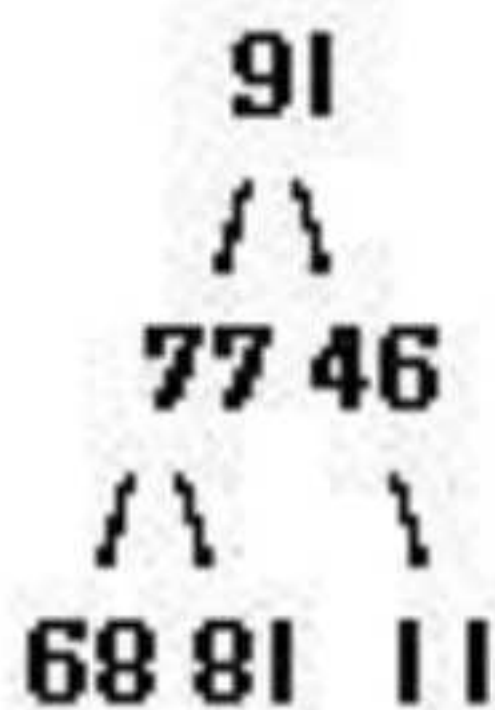
	32	51	27	85	66	23	13	57
Step1	32	51	27	85	66	23	13	57
Step2	32	27	51	85	66	23	13	57
Step3	32	27	51	66	85	23	13	57
Step4	32	27	51	66	23	85	13	57
Step5	32	27	51	66	23	13	85	57
Step6	32	27	51	66	23	13	57	85

Pass two:-

	32	27	51	66	23	13	57	85
Step1	32	27	51	66	23	13	57	85
Step2	27	32	51	66	23	13	57	85
Step3	32	27	66	23	66	13	57	85
Step4	32	27	66	23	13	66	57	85
Step5	32	27	66	23	13	57	66	85
Step6	32	27	66	23	13	57	66	85

Question No: 36 (Marks: 5)

Give two different reasons to explain why the following binary tree is not a heap:



Answer:- (Page 335)

Above array is not min heap or max heap because every parent node has not minimum or maximum value.
In max heap, each parent node has a value greater than the value of its left and right child nodes.
In min heap, each parent node has a value less than the value of its left and right child nodes.