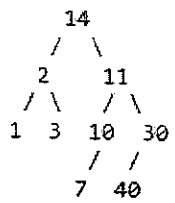
Ref. No. : Ex/IEBE/IT/T/314/2019

Name of the Examinations: B.E. INSTRUMENTATION AND ELECTRONICS ENGINEERING THIRD YEAR FIRST SEMESTER - 2019

Subject : DATA STRUCTURES & OOP **Time : 3 hours** **Full Marks : 100**

Section 1 2*10=20 Marks

1. What are splay trees?
2. What is recurrence for worst case of Quicksort and what is the time complexity in Worst case?
3. What is the best time complexity of bubble sort?
4. Consider the following sorting algorithms. I. Quicksort II. Heapsort III. Merge sort which of them perform in least time in the worst case?
5. Which one of the in place sorting algorithms needs the minimum number of swaps?
6. What algorithm design technique is used in merge sort?
7. The height of a binary tree is the maximum number of edges in any root to leaf path. The maximum number of nodes in a binary tree of height h is?
8. The maximum number of binary trees that can be formed with three unlabeled nodes is?
9. Here is a small binary tree:



Write the order of the nodes visited in:
B. A pre-order traversal:
C. A post-order traversal:

10. The keys 12, 18, 13, 2, 3, 23, 5 and 15 are inserted into an initially empty hash table of length 10 using open addressing with hash function h(k) = k mod 10 and linear probing. What is the resultant hash table?

Section 2 3*5=15 Marks

1. Given an unsorted array. The array has this property that every element in array is at most k distance from its position in sorted array where k is a positive integer smaller than size

of array. Which sorting algorithm can be easily modified for sorting this array and what is the obtainable time complexity?

2. Suppose we are sorting an array of eight integers using heapsort, and we have just finished some heapify (either maxheapify or minheapify) operations. The array now looks like this: 16 14 15 10 12 27 28 How many heapify operations have been performed on root of heap?
3. Mention what are the applications of Linked Lists?
4. Mention how to delete first node from singly linked list?
5. Suppose that a binary search tree contains the number 42 at a node with two children. Write two or three clear sentences to describe the process required to delete the 42 from the tree.

Section 3

2*10=20 Marks

1. Given the following sequence of letters and asterisks:
EAS*Y*QUE***ST***IO*N***

(a) Consider the stack data structure, supporting two operations push and pop. Suppose that for the above sequence, each letter (such as E) corresponds to a push of that letter onto the stack and each asterisk (*) corresponds a pop operation on the stack. Show the sequence of values returned by the pop operations. (5 Marks)

(b) Consider the queue data structure, supporting two operations insert and remove. Suppose that for the above sequence, each letter (such as E) corresponds to an insert of that letter into the queue and each asterisk (*) corresponds a remove operation on the queue. Show the sequence of values returned by the remove operations. (5 Marks).

2. (a) Determine and explain the functionality of the following code fragment: (2 Marks)

```
int function1 (int a[], int n, int x)
{
int i;
for (i = 0; i < n && a[i] != x; i++);
if (i == n)
return -1;
else
return i;
}
```

(b) Compute the best case, worst case and average case time complexity of the above function1.Explain your answers. (3 Marks)

(c) Determine and explain the functionality of the following code fragment: (2 Marks)

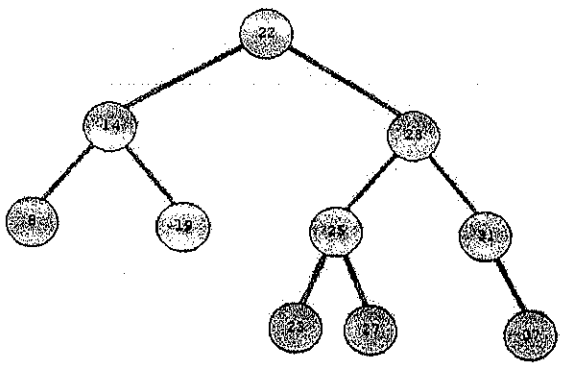
```
int function2 (int a[], int n, int x) {
int i, j, k; i = 0; j = n -1;
while (i <= j)
{ k = (i + j)/2;
if (x == a[k]), return k;
if (x > a[k]), i = k + 1;
else j = k - 1;
}
return -1;
}
```

(d) Compute the best case, worst case and average case time complexity of the above function2.Explain your answers. (3 Marks)

Section 4

15*3=45 Marks

1. Starting with the AVL tree below, insert the following values: 26,34,40,29,33,32. Show the resulting tree after each insertion.



2. Show the result of inserting 28, 12, 17, 4, 31, 34, 8, 14 &16 in that order into an initially empty B tree with M=3 and L=2(M is pointers in internal node and L is data items in leafs). Show the tree after each insertion, clearly labeling which tree is which. In an actual implementation, there is flexibility in how insertion overflow is handled. However, in this problem, follow these guidelines:

- Always use splitting and not adoption.
- Split leaf nodes by keeping the smallest 2 elements in the original node and putting the 1 largest element in the new node.

- Split internal nodes by keeping the 2 children with the smaller values attached to the original node and attach the 2 children with the larger values to the new node.

3. (A) Explain how you can implement a queue using a circular singly linked list. Show how you can do enqueue and dequeue operations in O (1) time. Note: A circular singly linked list is a singly linked list where the tail element points back to the head of the linked list (instead of pointing to a null element). The linked list is accessed by an external pointer pointing to one of the elements. [10 Marks]

(B) Consider a hash table with m slots. Under the simple uniform hashing assumption, what is the probability that after the first 5 insertions, the first 10 slots are not filled? [5 Marks]