COLLEGE OF ENGINEERING, PUNE

(An Autonomous Institute of Government of Maharashtra.)

END Semester Examination

Programme: B.Tech | **Semester: III**

Course Code: CT 16004 | **Course Name: DSA**

Branch: CS | **Academic Year: 2019 - 2020**

Duration: 3 Hours | **Max Marks: 60**

Student PRN No. □□□□□□□□□

Instructions:

1. Figures to the right indicate the full marks.
2. Mobile phones and programmable calculators are strictly prohibited.
3. Writing anything on question paper is not allowed.
4. Exchange/Sharing of stationery, calculator etc. not allowed.
5. Write your PRN Number on Question Paper.
6. Write answers in clear legitimate hand writing.
7. All answers are expected in C Programming language.
8. Ignore minor syntax errors or typing errors and make suitable assumptions.
9. Support you answers with diagrams wherever necessary.

			Marks	CO	PO
Q 1	a.	A binary tree T has 9 nodes. The inorder and preorder traversal of T yield the following sequence of nodes: Inorder : E A C K F H D B G Preorder : F A E K C D H G B Draw the binary tree T.	1		1
	b.	Take an initially empty hash table with five slots, with hash function h(x) = x mod 5, and with collisions resolved by chaining. Draw a sketch of what happens when inserting the following sequence of keys into it: 35, 2, 18, 6, 3, 10, 8, 5.	1	1	1
	c.	Derive the upper bound (ie Big -Oh notation), stating values of 'c' and 'n' for the following function: $f(n) = n^2 + n + 1$	2		1
	d.	An array of size 9 is given. Three stacks are implemented in this array in order 1,2 and 3. Size of each stack is :	2	1	1

Stack number	Stack size
1	2
2	3
3	4

Initially array contains no elements.

Operation *push(1,2)* means push value 2 in stack number 1.

Assume all push operations happen from *right* of stack.

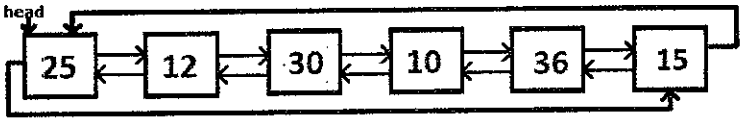
After performing the following operations, what are the contents of the array?

```
push(1,2)
push(2,3)
push(3,4)
push(1,5)
push(2, pop(1))
push(2,6)
push(3,8)
push(3,7)
push(3,pop(1))
push(1,pop(2))
push(1,5)
push(2,1)
```

e. Suggest the most suitable data structure for each of the following with brief reason: 2 1 1

i. Level order traversal of a tree.
ii. Syntax checking by compiler.
iii. Round robin scheduling of jobs by OS.
iv. Phone directory.

f. The following figure depicts the initial configuration of a doubly linked list: 2 2 1



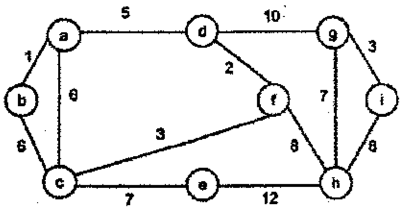
What would be the output after the following sequence of steps:

```
head->next->next ->next = head ->prev;
head->prev->prev->prev − head ->next->next->prev;
head->next->next->next->prev = head->prev->prev->prev;
head->next –head->next->next;
head->next->prev->next = head->next->next->next;
printf("%d", head->next->prev->next->data); // Write output at this stage.
```

g. Identify valid search sequence(s) for searching element 88 in a binary search tree: 2 2 1

i. 1,100, 200, 150, 99, 88
ii. 100, 90, 10, 30, 80, 95, 65, 88
iii. 40, 90, 45, 65, 89, 70,75, 88
iv. 120, 10, 200, 100, 20, 90, 60, 80, 88

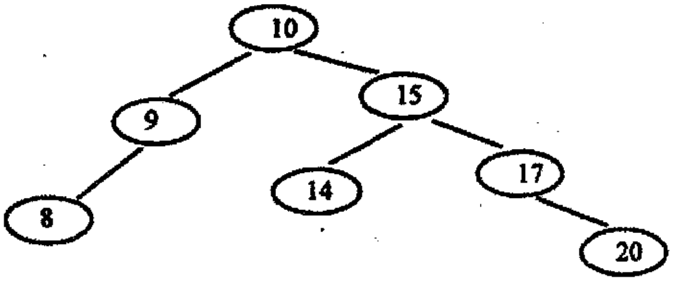
h. 2 2 1



For the undirected, weighted graph given above, which of the following sequences of edges represents a correct execution of Prim's algorithm to construct a Minimum Spanning Tree?

i. (a, b), (d, f), (f, c), (g, i), (d, a), (g, h), (c, e), (f, h)
ii. (c, e), (c, f), (f, d), (d, a), (a, b), (g, h), (h, f), (g, i)
iii. (d, f), (f, c), (d, a), (a, b), (c, e), (f, h), (g, h), (g, i)
iv. (h, g), (g, i), (h, f), (f, c), (f, d), (d, a), (a, b), (c, e)

i. A BST is as follows: 2 3 1



Draw possible configurations of the tree if 10 is deleted from it?

j. Match the following pairs. 2 1

A	O(log n)	i	Bubble sort
B	O(n)	ii	Insertion sort
C	O(n log n)	iii	Binary search
D	O(n^2)	iv	Merge sort

k. 2 1

```
int C(int n,int k){
     if(k = = 0 || n = =k)
         return 0;
     if(n>k )
         return C(n-1, k) + C(n-1, k-1);
}
```

Draw a recursion tree for the above function when called with C(5, 3). What is the value returned by the function?

Q2 **a.** 3 3 1

```
typedef struct node{
               char *data;
               struct node *next;
}node;
typedef node* list;
list* getlist(list *l, int n);
```

Write function *getlist()* which does the following: returns a new list which has all nodes from the given list (represented by ‘l’) such that the length of the string in that node is more than ‘n’. The function also returns the list ‘l’ of all nodes which have string length less than or equal to ‘n’.

b. Write a program which counts the number of occurrences of all lower-case vowels [a, e, i, o, u] in a given file. You should use open( ) function to open the file. The program will run as follows: 3 3 1

```
./program  <filename>
```

c. Write a *recursive function* to swap left and right subtrees all nodes of a binary tree. 3 2 1

```
tree* swap_subtrees(tree *t);
```

d. Write a non-recursive function to search for a string in a binary search tree stored as an array with root at index 0. 3 3 1

```
int search( tree *t, char*);
```

If the string is found function should return index of the string in the array. Analyse the code for best case and worst-case time complexity.

e. Draw diagram of data structure created by following code: 3 2 1

```
int main() {
       typedef struct data{
              struct data *right;
              char *val;
       }data;
        typedef struct test {
                struct test *vp, *vq;
                char *str;
        }test;
        test *p = (test *)malloc(sizeof(test)), *q;
        int i; q = p;
        for(i = 0; i < 3; i++) {
                 p->vq = (test *)malloc(sizeof(test));
                 p->vp = p;
                 p = p->vq;
        }
        p->vq = NULL;
        q→vp→vq→vq→vq→vp = q;
        p = q->vp; /* Draw diagram at this stage */
        return 0;
}
```

f. Using the given type definition for a polynomial, write a function to evaluate the polynomial for a given value of the variable. 3 3 1

```
typedef struct entry {
        double coeff;
        int power;
        struct entry *next;
}entry;
typedef entry *poly;
double evalpoly(poly a, double x);
```

Q 3 **a.** Working of enque() and dequeue() functions of ADT queue is demonstrated by the example below: 4 3 4

queue q

enqueue(q,56), enqueue(q,32), enqueue(q,18), enqueue(q,12)

56	32	18	12	

dequeue(q) = 56

32	18	12		

dequeue(q) =32

18	12			

Write queue.h and queue.c files for ADT queue with all functionalities.

b. Write a *non-recursive* DFS function for a graph using adjacency list. Assume stack of required datatype is available. 4 1 1

c. Write a generic function to sort an array of integers using insertion sort. You can select appropriate function prototype. 4 3 1

Array A has 12,8,6,14,7 elements

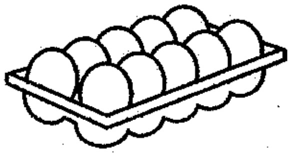
Show the contents of array A after every pass of the function.

d. Connection.txt file contains following segment of data. 4 3 4

Mumbai	Pune	149
Mumbai	Nashik	207
Kolhapur	Mumbai	376
Nashik	Pune	210
Kolhapur	Nashik	450

Draw Adjacency Matrix and Adjacency List representation for it.
What is the time complexity of determining connection between the cities in both the representations?
Write function using any representation to find degree of all the vertices of a graph.

Q 4 'Crate' is a tray used for storage and transport of items like eggs/ apples/ soft drink cans etc. 6 5 4



Take a case of "Crate of eggs" shown in the picture. When you buy a crate, it is filled with eggs. When all egges are consumed you discard it. When ever you want an egg from the crate, you pick a random egg. You can also keep back the unused egg in an empty slot of the crate. Crate can come varying sizes 2 * 5 or 3 * 4 etc. Design ADT 'Crate' using suitable storage representation with following functionalities:

Buy();
DiscardCrate ();
GetOne();
KeepBack();

Use your own prototypes. You can add more functions if necessary.

Write crate.h and crate.c files for the ADT Crate.