Manipal Institute of Technology, Manipal
Department of Information & Communication Technology
III- Semester BTech(IT/CCE)-FISAC-1(Take Home Assignment)
Subject: Data Structures(ICT 2153)

Max marks: 5(Each question carries 0.5 marks)

Instructions:

a. Don't copy answer from other students.
b. Answers must be hand written on paper and write Name, regno, rollnumber, semester, section and signature on top of each sheet.
c. Xerox copy is not accepted.
d. Answers should not be verbatim copy from the internet or any other sources
e. Assignment must be submitted personally to the concerned faculty between 26th October and 29th October 2022.

1. Write a user defined function to sort an array using pointer.

2. Write a user defined function to count the number of vowels and consonants in a string using a pointer.

3. Write the output for the below program. Input is : 1 2 3 4 5 6

```
#include <iostream>
using namespace std;
void test(int a[], int * p)
{
    p= a;
    p[0] = 20;
    *p = 40;
    p = &a[2];
    *p = 30;
    p--;
    *p=40;
    *(p+5) = 50;
  for(int i =0; i<6 ; i++)
    << a[i]<< endl ;
}
int main()
{
    int a[10], * p;
```

```cpp
    for(int i =0; i< 6; i++)
        cin>> a[i];
    test( a, p );
    return 0;
}
```

4. Write the output of the following program:

```cpp
int f(int x, int *py, int **ppz)
{ int y,z;
    **ppz+=1; z=**ppz;
    *py+=2; y=*py;
    x+=3;
    return x+y+z;}
int main()
{   int c, *b, **a;
    c=4; b=&c; a=&b;
    cout<<f(c,b,a);
    return 0;
}
```

5. Write a C++ function to implement a stack using 2 queues.

6. In a computer system, tasks T1, T2, T3… arrive with random priority for execution. Each task has the following attributes: username, task name, task size and arrival time. task size determines it's priority. Based on priority, next job is selected for execution. Suggest a suitable data structure and write a function to store and select jobs for execution on priority basis. Note that the data structure must store all the attributes of tasks.

7. A sparse matrix is a two-dimensional array in which a significant percentage of the array contains no data/zeros. For example, if we only store positive numbers in the matrix, this 5 x 5 integer matrix could be deemed sparse:

    0 0 0 9 0

    0 8 0 0 0

    7 0 0 6 0

    0 0 0 0 0

0 5 0 0 0

Instead of storing a sparse matrix in a two-dimensional array, we can use array of objects and only store objects for locations with non-zero values. The sparse matrix given above might be stored as indicated below:

0	<4,5,5> (data, row, col)
1	<9,0,3>
2	<8,1,1>
3	<7,2,0>
4	<5,4,1>

7A. Write the code snippet for the getByRow method, which returns the value in the matrix at the specified row and column by traversing the matrix along the list corresponding to the specified row. (Of course, if the needed object does not exist in the list, the value must be -1.)

7B. Write the code snippet for the getNumElementsInColumn method returns the number of objects in the array that correspond to the provided column in the matrix. This should equate to the number of entries in that column that are not zero. You must traverse the supplied list and count all of the objects. You cannot simply save this value in a variable/array and return it. This method verifies that your array of objects is properly configured.

8. Convert */+WZ*UR-+BA/DC to postfix using stack. Clearly show stack contents at each step of conversion. Don't convert to Infix.

9. Convert BA+ED/*ZYX+-^ to prefix using stack. Clearly show stack contents at each step of conversion. Don't convert to Infix.