Submitted by Submitted To

# Lab Exam

## Muhammad Awais / 2883 Ma’am Zanaib Malik

Department of Computer Science and Engineering

# National University of modern language, Islamabad

## Question # 01

#include <iostream>

using namespace std;

int main()

{

    int row;

    int colum;

    cout << "Enter no of rows \n";

    cin >> row;

    cout << "Enter no of colum \n";

    cin >> colum;

    int Graph[row][colum];

for (int i = 0; i < row; i++)

{

    for (int j = 0; j < colum; j++)

    {

    cin>>Graph[i][j];

    }

}

for (int i = 0; i < row; i++)

{

    for (size\_t j = 0; j < colum; j++)

    {

Cout<<i<<j<<endl;

    }

}

    return 0;

}

## Question # 02

template <class T>

void PQ<T>::enqueue(int prio, T element)

{

    PNode<T> \*n = new PNode<T>(prio, element, 0);

    if (rear == 0 && front == 0) // first E

    {

        rear = front = n;

    }

    else if (n->getPrio() > rear->getPrio())

    {

        rear->setNext(n);

        rear = n;

    }

    else if (n->getPrio() < front->getPrio()) // new node has highest prio

    {

        n->setNext(front);

        front = n;

    }

    else

    {

        PNode<T> \*ptr = front;

        while (((PNode<T> \*)ptr->getNext())->getPrio() <= n->getPrio())

        {

            ptr = (PNode<T> \*)ptr->getNext();

        }

        n->setNext(ptr->getNext());

        ptr->setNext(n);

    }

} // Enqueue

## Question # 03

template <class T>

bool LinkedTree<T>::isBSTree(TreeNode<T> \*t\_root)

{

    if (t\_root == 0)

    {

        return false;

    }

    if ((t\_root->getInfo() > t\_root->getLeft()->getInfo()) && (t\_root->getInfo() < t\_root->getRight()->getInfo()))

    {

        return true;

    }

    else if (t\_root->getInfo() > t\_root->getLeft()->getInfo())

    {

        return isBSTree(t\_root->getLeft());

    }

    else

    {

        return isBSTree(t\_root->getRight());

    }

}

## End of Lab 01