/\*LAB ASSIGNMENT 11

Write C++ program for simulating job queue. Write functions to add job and delete job from queue\*/

#include<iostream>

#include <stdlib.h>

using namespace std;

class qnode

{

int jobnumber;

qnode \*next;

friend class queue;

};

class queue

{

qnode \*front, \*rear;

public:

void enque(int val);

int deque();

int isempty();

void display();

queue ()

{

front=rear=NULL;

}

};

int queue::isempty()

{

if(front==NULL && rear==NULL)

return 1;

else

return 0;

}

void queue::enque(int val)

{

qnode \*t;

t=new qnode;

t->jobnumber=val;

t->next=NULL;

// If queue is empty, open new node is front and rear both

if (isempty())

{

front=t;

rear=t;

}

else //add the new node at the end of queue and change rear

{

rear->next-t;

rear=t;

}

}

int queue::deque()

{

qnode \*t;

int val;

if (isempty())

cout<<"\nQueue is empty\n";

else

{

// Store previous front and move front one nede ahead

t=front;

front=front->next;

// if front becomes so, then nauge rear als NULL

if (front==NULL)

rear=NULL;

val=t->jobnumber;

delete t;

}

return val;

}

void queue::display()

{

qnode \*t;

if (isempty())

cout<<"\nQueue is empty\n";

else

{

cout<<"\nQueue jobnumbers: ";

for (t=front;t!=NULL; t=t->next)

{

cout<<"\t"<<t->jobnumber;

}

}

}

int main()

{

int ch, n;

queue obj;

while (1)

{

cout <<"\n1. Add job \n2.Delete job \n3.Display job \n4.exit";

cout<<"\nEnter your choice:";

cin>>ch;

switch (ch)

{

case 1:

cout<<"\nEnter jobnumbers : ";

cin>>n;

obj.enque (n);

break;

case 2:

obj.deque();

break;

case 3:

obj.display();

break;

case 4:

exit (0);

default:

cout<<"\nYou entered wrong choice : ";

}

}

return 0;

}