/\*

Name: Swami Amit Bandu

Roll no. 2221018

Assigenment no. : 12

Problem statement : write c++ program for simulating double ended queue.

\*/

#include<iostream> using namespace std;

class queue

{

int q[5],f,r;

public:

queue()

{

f = -1;

r = -1;

}

int isfull();

int isempty();

void enq\_r();

void dq\_f();

void display();

void enq\_f();

void dq\_r();

};

int queue::isfull()

{

if(f==0 && r==4 || f==r+1)

{

return 1;

}

else

{

return 0;

}

}

int queue :: isempty()

{

if(f==-1 && r==-1)

{

return 1;

}

else

{

return 0;

}

}

void queue :: enq\_r()

{

int x;

cout << " Enter value to be inserted : " << endl;

cin >> x;

if(f==-1 && r==-1)

{

f = 0,r = 0;

q[r] = x;

}

else

{

r=(r+1)%5;

q[r] = x;

}

}

void queue :: dq\_f()

{

cout << "Deleted Element = " << q[f];

cout << endl;

if(f == r)

{

f=-1;

r=-1;

}

else

{

f=(f+1)%5;

}

}

void queue :: display()

{

int i;

for(i=f; i!=r; i=(i+1)%5)

{

cout << q[i];

cout << endl;

}

cout << q[i];

}

void queue :: enq\_f()

{

int x;

cout<<"Enter value to be inserted"<<endl;

cin>>x;

if(f==-1 && r==-1)

{

f=0,r=0;

q[f] = x;

}

else if (f==0)

{

f = 4;

q[f] = x;

}

else

{

f=(f-1)%5;

q[f]=x;

}

}

void queue :: dq\_r()

{

cout << "Deleted Element = " << q[r];

cout << endl;

if(f == r)

{

f=-1;

r=-1;

}

else if(r==0)

{

r=4;

}

else

{

r=(r-1)%5;

}

}

int main()

{

queue ob;

int x;

int ch;

while(1)

{

cout << " Enter 1 for Enque from Rear " << endl;

cout << " Enter 2 for Deque from Front " << endl;

cout << " Enter 3 for Display " << endl;

cout << " Enter 4 for Enque from Front " << endl;

cout << " Enter 5 for Deque from Rear " << endl;

cout << " Enter your choice = " << endl;

cin >> ch;

switch(ch)

{

case 1 :

if (ob.isfull())

{

cout << "Queue is Full" << endl;

}

else

{

ob.enq\_r();

}

break;

case 2 :

if (ob.isempty())

{

cout << "Queue is Empty" << endl;

}

else

{

ob.dq\_f();

}

break;

case 3 :

if (ob.isempty())

{

cout << "Queue is Empty" << endl;

}

else

{

ob.display();

}

break;

case 4 :

if (ob.isfull())

{

cout << "Queue is Full" << endl;

}

else

{

ob.enq\_f();

}

break;

case 5 :

if (ob.isempty())

{

cout << "Queue is Empty" << endl;

}

else

{

ob.dq\_r();

}

break;

default :

cout << "Wrong ch";

break;

}

}

}