//quetion 6

//In this problem you have to take an array of size 10 and divide it into two equal parts.

//In half part apply Stack and in remaining half apply Queue.

#include<iostream> //header files

using namespace std;

class stack\_que //class name stack and queue

{

private: //data member

int arr[20]; //variable array 20 index(0 to 19)

int f,r,t; //f=front,delation dequeue // r=rear,back,insertion,enqueue

public:

stack\_que() //constrature

{

t=9; //set postion 9 index means arr[10]

r=-1;

f=-1;

}

//////////////////////////////////////////////

void push\_stack(int v) //push function

{

if(t==20)

cout<<"\nstack is full : ";

else

arr[++t]=v;

cout<<"\nvalue entered in a stack is : "<<v<<endl;

}

////////////////////////////////////////////////

int pop\_stack() //pop function

{

if(t==9)

cout<<"\nstack is empty : ";

else

return arr[t--];

}

///////////////////////////////////////////////

void display\_stack() //display stack

{

for(int i=10;i<=t;i++)

cout<<arr[i]<<"\t";

}

/////////////////////////////////////////////

bool full\_que() //cheak is queue full???

{

if(r==10)

return true;

else

return false;

}

////////////////////////////////////////////

bool empty\_que() //cheak is queue empty???

{

if(r==-1)

return true;

else

return false;

}

/////////////////////////////////////////////

void enque\_que(int v) //enqueue function

{

if(full\_que())

cout<<"\nThe enque not possible because que is full !";

else

arr[++r]=v;

cout<<"\nThe value is entered in the que : "<<v<<endl;

}

///////////////////////////////////////////////////

int deque\_que() //dequeue function

{

if(empty\_que())

cout<<"\nThe deque not possible : ";

else

{

int z;

z=arr[r--]; //important working of deque

return z;

}

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

{

arr[i]=arr[i+1];

}

}

////////////////////////////////////////////////////

void display\_que() //display function

{

for(int j=0;j<=r;j++)

cout<<arr[j]<<"\t";

}

};

////////////////////////////////////////////////

int main() //main function

{

stack\_que c,s; //objects

int d;

char op;

do

{

cout<<"\nFor stack operation : (s)\n"

"For Que opeartion : (q)\n"

"Enter your choice : ";

cin>>op;

switch(op)

{

case 's':

system("color b5");

cout<<"\n\t\t\tstack part : \n";

cout<<"\npush function operate in a stack : \n";

s.push\_stack(1); //0 index

s.push\_stack(2); //1 index

s.push\_stack(3); //2 index

s.push\_stack(4); //3 index

s.push\_stack(5); //4 index

s.push\_stack(6); //5 index

s.push\_stack(7); //6 index

s.push\_stack(8); //7 index

s.push\_stack(9); //8 index

s.push\_stack(11); //9 index

cout<<"\nNow pop function from stack : \n";

cout<<s.pop\_stack()<<"\t";

cout<<s.pop\_stack()<<"\t";

cout<<s.pop\_stack()<<"\t";

cout<<s.pop\_stack()<<"\t";

cout<<"\ndisplay a Stack after some pop : \n";

s.display\_stack();

cout<<"\nNow again pop : \n";

cout<<s.pop\_stack()<<"\t";

cout<<s.pop\_stack()<<"\t";

cout<<s.pop\_stack()<<"\t";

cout<<s.pop\_stack()<<"\t";

cout<<s.pop\_stack()<<"\t";

cout<<s.pop\_stack()<<"\t";

break;

case 'q':

system("color 70");

cout<<"\n\t\t\tQueue part : \n";

c.enque\_que(10);

c.enque\_que(20);

c.enque\_que(30);

c.enque\_que(40);

c.enque\_que(50);

c.enque\_que(60);

c.enque\_que(70);

c.enque\_que(80);

c.enque\_que(90);

c.enque\_que(100);

cout<<"\nNow deque from que : "<<endl;

cout<<c.deque\_que()<<"\t";

cout<<c.deque\_que()<<"\t";

cout<<c.deque\_que()<<"\t";

cout<<"\ndisplay Queue after some deque : "<<endl;

c.display\_que();

cout<<"\nagain deque : \n";

cout<<c.deque\_que()<<"\t";

cout<<c.deque\_que()<<"\t";

cout<<c.deque\_que()<<"\t";

cout<<c.deque\_que()<<"\t";

cout<<c.deque\_que()<<"\t";

cout<<c.deque\_que()<<"\t";

cout<<c.deque\_que()<<"\t";

break;

} //switch B.C

cout<<"\n\n\ndo u again use stack or Queue \n"

"Enter your option (y/n) : ";

cin>>op;

//----------------------------------------------------

} //do B.C

while(op=='y');

system("pause");

return 0;

}