**Assignment No:- 1.3**

**Assignment Name:- Implementation of program based on stack.**

**Name:- YOGESH PRALHAD PATIL.**

**Roll No:- 104.**

#include<iostream.h>

#include<conio.h>

class STACK\_104

{

private:

int \*A,s,top;

public:

STACK\_104(int);

void PUSH(int ele);

int POP();

void LIST\_ALL();

};

STACK\_104::STACK\_104(int par)

{

s=par;

top=0;

A=new int[s+1];

}

void STACK\_104::PUSH(int ele)

{

if(top==s)

{

cout<<"\n Stack is full"<<endl;

return;

}

top=top+1;

A[top]=ele;

}

int STACK\_104::POP()

{

if (top==0)

{

cout <<"\n Stack is empty"<<endl;

return NULL;

}

int ele=A[top];

top=top-1;

return ele;

}

void STACK\_104::LIST\_ALL()

{

if (top==0)

{

cout<<"\n Stack is empty"<<endl;

}

else

{

cout<<"\n element in stack: "<<endl;

for(int i=top;i>=1;i--)

{

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

}

}

}

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

void MENU()

{

int opt,ele,s;

cout<<"\n Enter s of stack: "<<endl;

cin>>s;

STACK\_104 obj (s);

do

{

cout<<"\n 1.PUSH" ;

cout<<"\n 2.POP" ;

cout<<"\n 3.LIST" ;

cout<<"\n 4.EXIT" ;

cout<<"\n Choose your option: "<<endl;

cin>>opt;

switch (opt)

{

case 1:

cout<<"\n Enter Element to add: "<<endl;

cin>>ele;

obj.PUSH (ele);

break;

case 2:

int ele=obj.POP();

cout<<ele<<" is deleted"<<endl;

break;

case 3:

obj.LIST\_ALL();

break;

case 4:

return;

default:

cout<<"\n Invalid Option" ;

}

}

while(1);

}

void main()

{

int ele;

clrscr();

MENU();

getch();

}

**Assignment No:- 1.4**

**Assignment Name:- Implementation of program based on simple Queue.**

**Name:- YOGESH PRALHAD PATIL**

**Roll No:- 104.**

#include<iostream.h>

#include<conio.h>

class QUEUE\_104

{

private:

int \*A,s,front,rear;

public:

QUEUE\_104(int);

void QUEUE\_ADD\_104(int ele);

int QUEUE\_DEL\_104();

void LIST\_ALL\_104();

};

QUEUE\_104::QUEUE\_104(int par)

{

front=rear=0; s=par;

A=new int[s+1];

}

void QUEUE\_104::QUEUE\_ADD\_104(int ele)

{

if(rear==s)

{

cout<<"Queue is full"<<endl;

return;

}

if(front==0)

{

front=1;

}

rear=rear+1;

A[rear]=ele;

}

int QUEUE\_104::QUEUE\_DEL\_104()

{

if(front==0)

{

cout<<"QUEUE is empty"<<endl;

return NULL;

}

int ele=A[front];

if(front==rear)

front=rear=0;

else

front=front+1;

cout<<ele<<" is deleted"<<endl;

return ele;

}

void QUEUE\_104::LIST\_ALL\_104()

{

cout<<"Elements of QUEUE are: "<<endl;

if(front==0)

{

cout<<"QUEUE is empty"<<endl;

return;

}

else

for(int i=front;i<=rear;i++)

{

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

}

}

void MENU()

{

int ele,s,opt;

cout<<"Enter s of QUEUE : "<<endl;

cin>>s;

QUEUE\_104 obj(s);

do

{

cout<<"\n 1.QUEUE\_ADD\_104";

cout<<"\n 2.QUEUE\_DEL\_104";

cout<<"\n 3.LIST\_ALL\_104";

cout<<"\n 4.EXIT"<<endl;

cout<<"Enter your option : "<<endl;

cin>>opt;

switch(opt)

{

case 1:

cout<<"Enter ele to add in QUEUE: ";

cin>>ele;

obj.QUEUE\_ADD\_104(ele);

break;

case 2:

ele=obj.QUEUE\_DEL\_104();

break;

case 3:

obj.LIST\_ALL\_104();

break;

case 4:

return;

default:

cout<<"Invalid Option";

}

}while(1);

}

void main()

{

clrscr();

MENU();

getch();

}

**Assignment No:- 1.4**

**Assignment Name:- Implementation of program based on circular Queue.**

**Name:- YOGESH PRALHAD PATIL**

**Roll No:-** 104**.**

#include<iostream.h>

#include<conio.h>

class QUEUE\_104

{

private:

int \*A,s,front,rear;

public:

QUEUE\_104 (int);

void QUEUE\_ADD\_104(int ele);

int QUEUE\_DEL\_104();

void LIST\_ALL\_104();

};

QUEUE\_104::QUEUE\_104 (int par)

{

front=rear=0; s=par;

A=new int[s+1];

}

void QUEUE\_104::QUEUE\_ADD\_104 (int ele)

{

if((front==1 && rear==s)||(rear+1==front))

{

cout<<"QUEUE is full"<<endl;

return;

}

if(front==0)

{

front=1;

}

if(rear==s)

rear=0;

rear=rear+1;

A[rear]=ele;

}

int QUEUE\_104::QUEUE\_DEL\_104()

{

if(front==0)

{

cout<<"QUEUE is empty";

return NULL;

}

int ele=A[front];

if(front==rear)

front=rear=0;

else

{

if(front==s)

front=0;

front=front+1;

}

cout<<ele<<"is deleted"<<endl;

return ele;

}

void QUEUE\_104::LIST\_ALL\_104()

{

cout<<"Elements of QUEUE are: "<<endl;

if(front==0)

{

cout<<"QUEUE is empty"<<endl;

return;

}

else

if(front<=rear)

for(int i=front;i<=rear;i++)

{

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

}

else

{

for(int i=front;i<=s;i++)

cout<<A[i]<<" ";

for(i=1;i<=rear;i++)

cout<<A[i]<<" ";

}

}

void MENU()

{

int ele,s,opt;

cout<<"Enter s of QUEUE : "<<endl;

cin>>s;

QUEUE\_104 obj(s);

do

{

cout<<"\n 1.QUEUE\_ADD\_104";

cout<<"\n 2.QUEUE\_DEL\_104";

cout<<"\n 3.LIST\_ALL\_104";

cout<<"\n 4.EXIT"<<endl;

cout<<"Enter your option : "<<endl;

cin>>opt;

switch(opt)

{

case 1:

cout<<"Enter ele to add in QUEUE: "<<endl;

cin>>ele;

obj.QUEUE\_ADD\_104(ele);

break;

case 2:

ele=obj.QUEUE\_DEL\_104();

break;

case 3:

obj.LIST\_ALL\_104();

break;

case 4:

return;

}

}while(1);

}

void main()

{

clrscr();

MENU();

getch();

}