


```

void del()
{
    node *nn;
    nn=NULL;
    nn=new node;
    if(!front)
        cout<<"Underflow";
    else
    {
        nn=front;
        front=front->next;
        delete nn;
    }
}
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
void display()
{
    node *nn;
    nn=NULL;
    nn=new node;

    if(!front)
        cout<<"\nUnderflow";
    else
    {
        nn=front;
        while(nn)
        {
            cout<<"\nroll"<<nn->roll;
            nn=nn->next;
        }
    }
}
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

//*****Void Main()*****//
void main()
{
    clrscr();
    int ch;

    do
    {
        cout<<"\n\tMain Menu:"
            <<"\n1)\tINSERT"
            <<"\n2)\tDELETE"
            <<"\n3)\tDISPLAY";
        cout<<"\n\nEnter Choice\t";      cin>>ch;
        switch(ch)
        {

```

```
case 1:
    insert();
    break;
case 2:
    del();
    break;
case 3:
    display();
    break;
default:
    cout<<"Invalid Choice";
}
cout<<"\nPress Y to continue:\t";
}while(toupper(getche())=='Y');
}
//////////////////////////////////////////////////////////////////////////////////////////////////////////////////

//*****End Of Main*****//
-.-.-.-.- Queue2/ With OOPS/ Dynamically -.-.-.-.-
#include<iostream.h>
#include<conio.h>
#include<ctype.h>
//////////////////////////////////////////////////////////////////////////////////////////////////////////////////
struct node
{
int roll;
node *next;
};
//////////////////////////////////////////////////////////////////////////////////////////////////////////////////
node *create_node()
{
node *nn;
nn=NULL;
nn=new node;
if(nn)
{
cout<<"\nEnter Roll\t";
cin>>nn->roll;
nn->next=NULL;
}
return nn;
}
//////////////////////////////////////////////////////////////////////////////////////////////////////////////////
class queue
{
node *front,*rear;
public:
queue()
{
front=rear=NULL;
}
```

```

// _____ //
~queue()
{
    while(front)
    {
        del();
    }
}
// _____ //
void insert()
{
    node *nn;
    nn=create_node();
    if(nn)
    {
        if(front)
        {
            rear->next=nn;
            rear=nn;
        }
        else
            front=rear=nn;
    }
    else
        cout<<"Overflow";
}
// _____ //
void del()
{
    node *nn;
    nn=NULL;
    nn=new node;
    if(!front)
        cout<<"Underflow";
    else
    {
        nn=front;
        front=front->next;
        delete nn;
    }
}
// _____ //
void display()
{
    node *nn;
    nn=NULL;
    nn=new node;

    if(!front)
        cout<<"\nUnderflow";
}

```



```

/////////////////////////////////////////////////////////////////
const int N=50;
/////////////////////////////////////////////////////////////////
struct student
{
    int roll;
};
/////////////////////////////////////////////////////////////////
int front=-1, rear=-1;
student Q[N];
/////////////////////////////////////////////////////////////////
void insert()
{
    if(rear==N-1)
        cout<<"overflow";
    else
    {
        if(front==-1)
            front=rear=0;
        else
            rear++;
        cout<<"\nEnter Roll:\t";
        cin>>Q[rear].roll;
    }
}
/////////////////////////////////////////////////////////////////
void del()
{
    if(front==-1)
        cout<<"Underflow";
    else
    {
        if(rear==front)
            rear=front=-1;
        else
            front++;
    }
}
/////////////////////////////////////////////////////////////////
void display()
{
    if(front==-1)
        cout<<"Underflow";
    else
    {
        int i;
        for(i=front;i<=rear;i++)
        {
            cout<<"\nRoll\t";
            cout<<Q[i].roll;
        }
    }
}

```

```
}
//////////////////////////////////////////////////////////////////////////////////////////////////////////////////
//*****Void Main()*****//
void main()
{
clrscr();
int ch;

do
{
cout<<"\n\tMain Menu:"
    <<"\n1)\tINSERT"
    <<"\n2)\tDELETE"
    <<"\n3)\tDISPLAY";
cout<<"\n\nEnter Choice\t";      cin>>ch;
switch(ch)
{
case 1:
insert();
break;
case 2:
del();
break;
case 3:
display();
break;
default:
cout<<"Invalid Choice";
}
cout<<"\nPress Y to continue:\t";
}while(toupper(getche())=='Y');
}
//*****End Of Main*****//
-.-.-.-.- Queue4/ With OOPS/ Statically -.-.-.-.-
#include<iostream.h>
#include<conio.h>
#include<ctype.h>
//////////////////////////////////////////////////////////////////////////////////////////////////////////////////
const int N=50;
//////////////////////////////////////////////////////////////////////////////////////////////////////////////////
struct student
{
int roll;
};
//////////////////////////////////////////////////////////////////////////////////////////////////////////////////
class queue
{
int front, rear;
student Q[N];

public:
```

```

// _____//
queue()
{
front=rear=-1;
}
// _____//
~queue()
{
front=rear=-1;
}
// _____//
void insert()
{
if(rear==N-1)
    cout<<"overflow";
else
{
if(front==-1)
    front=rear=0;
else
    rear++;
cout<<"\nEnter Roll:\t";
cin>>Q[rear].roll;
}
}
// _____//
void del()
{
if(front==-1)
    cout<<"Underflow";
else
{
if(rear==front)
    rear=front=-1;
else
    front++;
}
}
// _____//
void display()
{
if(front==-1)
    cout<<"Underflow";
else
{
int i;
for(i=front;i<=rear;i++)
{
cout<<"\nRoll\t";
cout<<Q[i].roll;
}
}
}

```



```

}
};
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
//*****Void Main()*****//
void main()
{
clrscr();
int ch;
queue q;
do
{
cout<<"\n\tMain Menu:"
<<"\n1)\tINSERT"
<<"\n2)\tDELETE"
<<"\n3)\tDISPLAY";
cout<<"\n\nEnter Choice\t";      cin>>ch;
switch(ch)
{
case 1:
q.insert();
break;
case 2:
q.del();
break;
case 3:
q.display();
break;
default:
cout<<"Invalid Choice";
}
cout<<"\nPress Y to continue:\t";
}while(toupper(getche())=='Y');
}
//*****End Of Main*****//
.-.-.-.-.-

```

OUTPUT
(COMMON FOR ALL)

```
      Main Menu:
1)      INSERT
2)      DELETE
3)      DISPLAY

Enter Choice    1

Enter Roll:     1

Press Y to continue:  y
      Main Menu:
1)      INSERT
2)      DELETE
3)      DISPLAY

Enter Choice    3

Roll    1
Press Y to continue:  _
```

```
      Main Menu:
1)      INSERT
2)      DELETE
3)      DISPLAY

Enter Choice    1

Enter Roll:     1

Press Y to continue:  y
      Main Menu:
1)      INSERT
2)      DELETE
3)      DISPLAY

Enter Choice    3

Roll    1
Press Y to continue:
```

```

Main Menu:
1)    INSERT
2)    DELETE
3)    DISPLAY

Enter Choice    1

Enter Roll:     1

Press Y to continue:  y
Main Menu:
1)    INSERT
2)    DELETE
3)    DISPLAY

Enter Choice    3

Roll    1
Press Y to continue:
```

```

Main Menu:
1)    INSERT
2)    DELETE
3)    DISPLAY

Enter Choice    1

Enter Roll      1

Press Y to continue:  y
Main Menu:
1)    INSERT
2)    DELETE
3)    DISPLAY

Enter Choice    2

Press Y to continue:  y
Main Menu:
1)    INSERT
2)    DELETE
3)    DISPLAY

Enter Choice    3

Underflow
Press Y to continue:
```