

```

//Queue(ADT) Implementation using Array
//Author- Jyoti Lakhani
#include<iostream>
#define SIZE 5
using namespace std;
int value;

class QUEUE
{
    private:
        int q[5];
        int front;
        int rear;
    public:
        QUEUE(){ front = -1; rear = -1;    }
        int insert(int);
        int qdelete();
        void display();
};

int QUEUE::insert(int info)
{
    if(rear== SIZE-1)
    {
        cout<<"\nQueue is overflow\n";
        return 0;
    }
    else
    {
        if(front==-1 & rear == -1)
        {
            front =0;
            rear = 0;
        }
        else
        {rear++;
        }
        q[rear] = info;
    }
}

int QUEUE::qdelete()
{
    if(front>rear|| rear==-1)
    {
        cout<<"\n queue is underflow \n";
        return 0;
    }
    else
    {
        if(front==rear)
        {
            cout<<"Deleting Item:\n"<<q[front];
            value = q[front];
            front=-1;
            rear=-1;
            return value;
        }
        else

```

```

        {
        cout<<"Deleting Item:\n";
        value = q[front];
        cout<<"value:"<<value<<endl;
        front=front+1;
        return value;
        }

    }

}

void QUEUE::display()
{
cout<<"In Display:\n";
    if(front==-1 & rear==-1)
    {
        cout<<"Queue is empty\n";
        return;
    }
    for(int i=front; i<=rear; i++)
    {
        cout<<q[i]<<" -> ";
    }
    cout<<endl;
}

int main()
{
    QUEUE Q1;
    int flag=0;
    int ch;
    while(flag==0)
    {
        cout<<"Choose an option\n";
        cout<<"1. Insert \n2. Delete \n3. Display \n4. exit\n" ;
        cin>>ch;
        switch(ch)
        {
            case 1:
                cout<<("Enter value to insert\n");
                cin>>value;
                Q1.insert(value);
                break;
            case 2:
                value = Q1.qdelete();
                cout<<value<<" deleted\n";
                break;
            case 3:
                Q1.display();
                break;
            case 4:
                cout<<"Exiting...\n";
                return 0;
        }
    }
}

```