NAME – AARYAN BAIRAGI

ROLL NO – 27002

CLASS – SE IT B

#include <iostream>

#define MAX 5

using namespace std;

class CircleQueue {

    int queue [MAX], front, rear;

    public:

        CircleQueue()

        {

            front=-1;

            rear=-1;

        }

        void EnQueue(int n) {

            if(isEmpty())

                front ++ ;

            rear=(rear+1)% MAX;

            queue[rear]=n;

        }

        void DeQueue() {

            cout<<queue[front];

            if(front==rear) {

                front=rear=-1;

            }

            else {

                front=(front+1)%MAX;

            }

        }

        bool isEmpty() {

            if(front==-1){

                return true;

            }

            else {

                return false;

            }

        }

        bool isFull() {

            if((rear+1)%MAX==front) {

                return true;

            }

            else {

                return false;

            }

        }

        void Peek() {

            cout<<"Elements at Front : "<<queue[front];

        }

        void display() {

            int i;

            if(front==-1) {

                cout<<"\n Queue is Empty !!! ";

                return;

            }

            cout<<"\nQueue : ";

            if(front <= rear) {

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

                    cout<<queue[i]<<" ";

                }

            }

            else {

                for(i=front; i<MAX; i++) {

                    cout<<queue[i]<<" ";

                }

                for(i=0;i<=rear;i++) {

                    cout<<queue[i]<<" ";

                }

            }

        }

};

int main()

{

    int op,element;

    CircleQueue obj;

    while(1) {

        cout<<"\n1. EnQueue";

        cout<<"\n2. DeQueue";

        cout<<"\n3. Display All Elements";

        cout<<"\n4. Display Front Elements ";

        cout<<"\n5. Exit";

        cout<<"\nEnter option : ";

        cin>>op;

        switch(op) {

            case 1 :

                if(obj.isFull()) {

                    cout<<"\nCan't insert....Queue Overflowe!!!";

                    break;

                }

                else {

                    cout<<"\nEnter element to be inserted : ";

                    cin>>element;

                    obj.EnQueue(element);

                    break;

                }

            case 2 :

                if(obj.isEmpty()) {

                    cout<<"\nCan't Delete....Queue Underflow!!!";

                    break;

                }

                else {

                    cout<<"\nDeleted element from queue : ", obj.DeQueue();

                    break;

                }

            case 3 :

                obj.display();

                break;

            case 4 :

                if(obj.isEmpty()) {

                    cout<<"\nNo front element....Queue Empty!!!";

                    break;

                }

                else {

                    obj.Peek();

                    break;

                }

            case 5 :

                 exit(0);

                 break;

        }

    }

    return 0;

}