

7) Queue Operation

```
#include <iostream>
using namespace std;
class Queue
{
public:
    int front, rear, n, q[50];
    void init()
    {
        cout << "\n\tQueue Operation:";
        cout << "\n Enter the size of Queue:";
        cin >> n;
        front = rear = 0;
    }
    void insert()
    {
        if (rear == n)
        {
            cout << "\n Queue is full";
        }
        else
        {
            front = 0;
            cout << "\n Enter the element:";
            cin >> q[rear];
            rear = rear + 1;
        }
    }
    void del()
    {
        if (front > rear)
        {
            cout << "\n Q is empty";
        }
        else
        {
            cout << "\n Deleted element is " << q[front];
            front = front + 1;
            rear = rear - 1;
        }
    }
    void display()
    {
        for (int i = front; i < rear; i++)
        {
            cout << "\n The Element " << i << " is "
```

```

        << q[i];
    }
}
};
int main()
{
    int ch;
    char ans = 'y';
    Queue q;
    q.init();
    while (ans == 'y')
    {
        cout << "\n 1. Insert";
        cout << "\n 2. Delete";
        cout << "\n 3. Display";
        cout << "\n Enter your choice:";
        cin >> ch;
        switch (ch)
        {
        case 1:
        {
            q.insert();
            break;
        }
        case 2:
        {
            q.del();
            break;
        }
        case 3:
        {
            q.display();
            break;
        }
        default:
            cout << "Enter the valid choice";
        }
        cout << "Do you want to continue (y/n):";
        cin >> ans;
    }
    return 0;
}

```