

```

1  #include <iostream>
2  using namespace std;
3
4  const int MAX_SIZE = 3;
5
6  class Queue {
7  private:
8      int front, rear;
9      int arr[MAX_SIZE];
10
11 public:
12     Queue() {
13         front = -1;
14         rear = -1;
15     }
16
17     bool isFull() {
18         return (rear == MAX_SIZE - 1);
19     }
20
21     bool isEmpty() {
22         return (front == -1 || front > rear);
23     }
24
25     void enqueue(int data) {
26         if (isFull()) {
27             cout << "Queue is full. Cannot enqueue." << endl;
28             return;
29         }
30         rear++;
31         arr[rear] = data;
32         if (front == -1) {
33             front = 0;
34         }
35         cout << "Enqueued " << data << " into the queue." << endl;
36     }
37
38     int dequeue() {
39         if (isEmpty()) {
40             cout << "Queue is empty. Cannot dequeue." << endl;
41             return -1;
42         }
43         int data = arr[front];
44         front++;
45         return data;
46     }
47
48     void display() {
49         if (isEmpty()) {
50             cout << "Queue is empty." << endl;
51             return;
52         }
53         cout << "Queue: ";
54         for (int i = front; i <= rear; i++) {
55             cout << arr[i] << " ";
56         }
57         cout << endl;
58     }
59 };
60
61 class Stack {
62 private:
63     int top;
64     int arr[MAX_SIZE];
65
66 public:
67     Stack() {
68         top = -1;
69     }
70
71     bool isFull() {
72         return (top == MAX_SIZE - 1);
73     }
74
75     bool isEmpty() {
76         return (top == -1);
77     }
78
79     void push(int data) {
80         if (isFull()) {

```

```

81         cout << "Stack is full. Cannot push." << endl;
82         return;
83     }
84     top++;
85     arr[top] = data;
86     cout << "Pushed " << data << " into the stack." << endl;
87 }
88
89 int pop() {
90     if (isEmpty()) {
91         cout << "Stack is empty. Cannot pop." << endl;
92         return -1;
93     }
94     int data = arr[top];
95     top--;
96     return data;
97 }
98
99 void display() {
100     if (isEmpty()) {
101         cout << "Stack is empty." << endl;
102         return;
103     }
104     cout << "Stack: ";
105     for (int i = top; i >= 0; i--) {
106         cout << arr[i] << " ";
107     }
108     cout << endl;
109 }
110 };
111
112 int main() {
113     int nim = 22040700112;
114     int x = (nim % 100);
115
116     int arr[3] = {4, 1, 2};
117
118     Queue queue;
119     Stack stack;
120
121     cout << "Array: ";
122     for (int i = 0; i < 3; i++) {
123         cout << arr[i] << " ";
124     }
125     cout << endl;
126
127     for (int i = 0; i < 3; i++) {
128         int value;
129         cout << "Enter a value: ";
130         cin >> value;
131         queue.enqueue(value);
132         stack.push(value);
133     }
134
135     int deletedValue = queue.dequeue();
136     cout << "Deleted value from the queue: " << deletedValue << endl;
137
138     queue.display();
139     stack.display();
140
141     return 0;
142 }

```

```
C:\Users\USER\Desktop\TUBES.exe
Array: 4 1 2
Enter a value: 2
Enqueued 2 into the queue.
Pushed 2 into the stack.
Enter a value: 4
Enqueued 4 into the queue.
Pushed 4 into the stack.
Enter a value: 6
Enqueued 6 into the queue.
Pushed 6 into the stack.
Deleted value from the queue: 2
Queue: 4 6
Stack: 6 4 2

-----
Process exited after 12.94 seconds with return value 0
Press any key to continue . . .
```