

Implement Queue using an Array

// Function to push an element x in the queue.

```
void MyQueue::push(int x) {  
    if (rear < 100005) {  
        arr[rear] = x;  
        rear++;  
    }  
}
```

// Function to pop an element from the queue and return it.

```
int MyQueue::pop() {  
    if (front == rear) return -1;  
    int res = arr[front];  
    front++;  
    return res;  
}
```

The screenshot displays a C++ IDE interface. On the left, the 'Output Window' shows 'Compilation Results' for a problem solved successfully. It reports 170/170 test cases passed, 1/1 attempts correct, 100% accuracy, 1/1 points scored, and a time taken of 0.7 seconds. The 'Solve Next' section suggests related problems: 'C++ STL | Set 5 (queue)', 'Queue Reversal', and 'Professor and Parties'. The main editor on the right shows the C++ code for the 'MyQueue' class, including the 'push' and 'pop' functions, with line numbers 53 to 88. The code is in C++ (g++ 5.4) and includes a 'Start Timer' button. The bottom status bar shows the temperature as 28°C, the language as ENG, and the time as 16:21.

Problem Solved Successfully ✓

Test Cases Passed: **170 / 170**

Attempts : Correct / Total: **1 / 1**

Accuracy : 100%

Points Scored: **1 / 1**

Your Total Score: 35 ↑

Time Taken: **0.7**

Solve Next

- C++ STL | Set 5 (queue)
- Queue Reversal
- Professor and Parties

```
1 // Driver Code Ends  
53  
54  
55 /*  
56  
57 The structure of the class is  
58 class MyQueue {  
59 private:  
60     int arr[100005];  
61     int front;  
62     int rear;  
63  
64 public :  
65     MyQueue(){front=0;rear=0;}  
66     void push(int);  
67     int pop();  
68 };  
69 */  
70  
71  
72 // Function to push an element x in the queue.  
73 void MyQueue::push(int x) {  
74     if (rear < 100005) {  
75         arr[rear] = x;  
76         rear++;  
77     }  
78 }  
79  
80 // Function to pop an element from the queue and return it.  
81 int MyQueue::pop() {  
82     if (front == rear) return -1;  
83     int res = arr[front];  
84     front++;  
85     return res;  
86 }  
87  
88
```