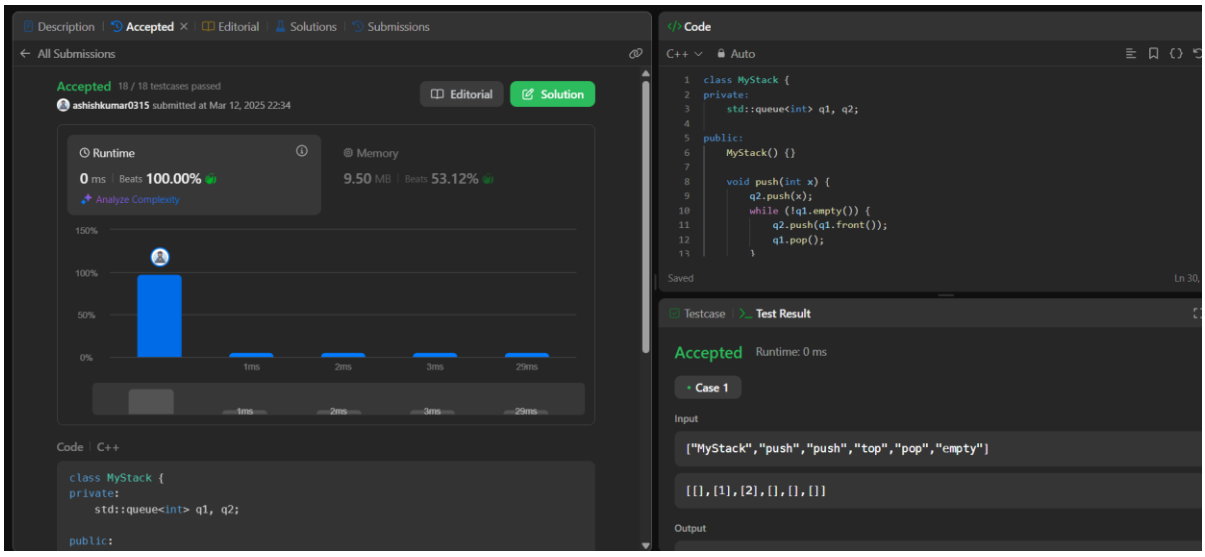
232. Implement Queue using Stacks

```
class MyStack {
private:
    std::queue<int> q1, q2;
public:
    MyStack() {}
    void push(int x) {
        q2.push(x);
        while (!q1.empty()) {
            q2.push(q1.front());
            q1.pop();
        }
        std::swap(q1, q2);
    }
    int pop() {
        int topElement = q1.front();
        q1.pop();
        return topElement;
    }
    int top() {
        return q1.front();
    }
    bool empty() {
        return q1.empty();
    }
};
```



225. Implement Stack using Queues:

```cpp
class MyStack {

private:
    std::queue<int> q1, q2;

public:
    MyStack() {}

    void push(int x) {
        q2.push(x);
        while (!q1.empty()) {
            q2.push(q1.front());
            q1.pop();
        }
        std::swap(q1, q2);
    }

    int pop() {
        int topElement = q1.front();
        q1.pop();
        return topElement;
    }

    int top() {
        return q1.front();
    }

    bool empty() {
        return q1.empty();
    }
};
```