Review Activity 10 Sample Solutions

Stacks and Queues Practice 2

Implement a class called **SpecialQueue** that implements a **queue** using two stacks. Implement the following methods: **int size(), void enqueue(int), int top(), void dequeue()**. If the **SpecialQueue** is empty, then return **-999**. In your code, you may only call the **<stack>** methods, such as **bool empty(), int size(), void push (int value), void pop(), and int top()**. You may use helper functions if needed, but those need to be non-recursive too.

```
    class SpecialQueue {
        const int EMPTY_STACK = -999;
        stack<int> s1, s2;

    public:
        int size() {
            return s1.size() + s2.size();
        }

        void enqueue(int value) {
            s1.push(value);
        }

        int top() {
            if (!s2.empty())
                return s2.top();

            while (!s1.empty()) {
                s2.push(s1.top());
                s1.pop();
            }

            if (!s2.empty())
                return s2.top();
            else
                return EMPTY_STACK;
        }
```

```
void dequeue() {
        if (!s2.empty()) {
            int q = s2.top();
            s2.pop();
            return;
        }

        while (!s1.empty()) {
            s2.push(s1.top());
            s1.pop();
        }

        if (!s2.empty()) {
            int q = s2.top();
            s2.pop();
        }
    }
};
```