Lab 4: Queues as Stacks

For this project, you need to implement a **queue ADT** using **at most** two stacks. Imagine you are writing a class that needs to behave like a queue (first in, first out), but can only use at most two stacks to store values. How would the **enqueue** and **dequeue** functions behave? What is the running time of the **enqueue** and **dequeue** functions in this case? What about functions **front** and **back**?

Using the **start project,** complete the implementation of the class **MyQueue**. The class already has member variables declared; do **NOT** declare additional member variables. One of the member variables is a **pointer** to an **STL stack**, which stores the values of the queue. The second stack that you need to manipulate the values will be created as a local object inside the functions that need a second stack to manipulate the stack pointed by the calling object.

Restrictions:

- You may **only** use the following functions from the **STL stack class**:
    - push
    - pop
    - top
    - size
    - empty
    - ❖ (**May NOT** use functions emplace and swap.)
- You may create a **temporary stack** where needed in any member function.
- You may **NOT** use the STL queue or any other container, other than the temp stack listed above and, of course, the stack to which the member variable is pointing.

The **Main.cpp** function already contains test cases.

Do **NOT** modify the **main** function.