CSCI 340 **Computer Assignment 4** **Spring 2017**

(10 points)

Stack

For this computer assignment, you are to implement the Stack class using STL queues. All relevant files are located at /home/turing/mhou/public/csci340spring2017.

`assignment4.h` contains the definition of the Stack class. It is given here to facilitate the following description:

```
class Stack {
    private:
        std::queue<int> q1, q2;
    public:
        bool empty() const;
        int size() const;
        int top();
        void push(const int& val);
        void pop();
};
```

You are required to implement this class in assignment4.cc. In this file, the main function is already provided. The driver program works with an input file `assignment4input.txt`.

In the implementation of the class, you are going to use queues `q1` and `q2` to store and manipulate data. You are suggested to keep all elements in one of the queues at anytime. More details are described below.

`empty()`:	You need to make sure both `q1` and `q2` are empty.
`size()`:	You need to count the number of elements in both `q1` and `q2`.
`top()`:	This method returns the newest element. If `q1` is not empty, simply return the end element of `q1`. Otherwise `q2` is not empty and simply return the end element of `q2`.
`push()`:	Simply add the element to a non-empty queue. If both queues are empty, the new element can be added to an arbitrary queue.
`pop()`:	This method removes the newest element. Since all elements are in one of the queues, say it is the `source`, you need to dump all elements except the newest to the other queue. And then remove the last (i.e. the newest) element in the `source`.

Programming Notes:

- Include any necessary headers.

- In the final version of your assignment, you are not supposed to change existing code, including the class definition and the `main` method, provided to you in the original files `assignment4.h` and `assginment4.cc`.

- To compile the source file, execute “ `g++ -Wall assignment4.cc –o assignment4.exe`”. This will create the executable file `assignment4.exe.` To test your program, execute “`./assignment4.exe < assignment4input.txt > assignment4.out 2>&1`”, which will put the output and error in file `assignment4.out`. `assignment4input.txt` is the input file. You can find the correct output of this program in file `assignment4.out` in the directory shown in the last page.

- Add documentation to your source file.

- Prepare your `Makefile` so that the TA only needs to invoke the command “`make`” to compile your source file and produce the executable file `assignment4.exe`. Make sure you use exactly the same file names specified here, i.e. `assignment4.cc` and `assignment4.exe`, in your `Makefile`. Otherwise your submission will get 0 points.

- When your program is ready, submit your source file assignment4.cc and Makefile to your TA by following the Assignment Submission Instructions.