

Queue

For this computer assignment, you are to implement the Queue class using STL stacks. All relevant files are located at /home/turing/mhou/public/csci340spring2019.

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

```
class Queue {
private:
    std::stack<int> s1, s2;
public:
    bool empty() const;
    int size() const;
    int front();
    int back();
    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 stacks s1 and s2 to store and manipulate data. You are suggested to use only s1 to save the element just “pushed” in (i.e. enqueueed), and use s2 to hold older elements. More details are described below.

empty():	You need to make sure both s1 and s2 are empty.
size():	You need to count the number of elements in both s1 and s2.
front():	This method returns the oldest element. First of all if s2 is empty, move all elements from s1 to s2. Simply return the top element in s2.
back():	This method returns the newest element. Simply return the top element in s1. However, if s1 is empty, it is possible that s2 is not empty. In that case, you need to move all elements back to s1 first.
push():	Simply add the element to s1.
pop():	This method removes the oldest element. You can reuse the method front().

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 `assignment4.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 the 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 may get 0 point.
- When your program is ready, submit your source file `assignment4.cc` and `Makefile` to your TA by following the Assignment Submission Instructions.