CS 3358 Section 252 – Assignment 2

Due Date: March 9, 2023

This assignment has three parts pertaining to the queue implementation. The parts of code are given in the `.cpp` and `.h` files. The places you need to fill out in the code are marked by `// TODO`.

The queue implementation is supposed to be **array-based** in this assignment.

- (30 points) In `myqueue1.h`, implement the member functions of the class `MyQueue` based on the first method using a `counter` variable that keeps track of the number of elements in the queue.

- (30 points) In `myqueue2.h`, implement the member functions of the class `MyQueue` based on the second method using the `front` and `rear` indexes only (i.e., without having the `counter` variable).
    1. The number of elements in the queue must range from 0 to array size – 1 (i.e., the maximum capacity of the queue is the array size minus one).
    2. Make sure that the calculation of the current queue size in `CurrentSize()` is correct. It should be calculated based only on the `front` and `rear` indexes since there is no `counter` variable that tracks the number of elements in the queue.
    3. You may want to make use of `CurrentSize()`in `IsFull()` and `IsEmpty()`.

- (40 points) In `apptest.cpp`, complete the implementation of the function `CountStudent` for the following problem:

The CS department at Texas State offers black and white gaming mice to students, referred to by numbers `0` and `1`, respectively. All students stand in a **queue**. Each student either prefers black or white gaming mice.

The number of gaming mice provided by the department is equal to the number of students. The gaming mice are placed in a stack. At each step:

- If the student at the front of the queue **prefers** the gaming mouse on the top of the stack, they will **take it and leave** the queue.
- Otherwise, they will **leave it and go to** the queue's end.

This continues until none of the queued students want to take the top mouse and are thus unable to have.

You are given two integer arrays `mice` and `students`, where `mice[i]` is the type of the i-th mouse in the stack (`i = 0` is the top of the stack), and `students[j]` is the preference of the j-th student in the initial queue (`j = 0` is the front of the queue). **Return the number of students who are unable to have the gaming mice.**

Compilation:
You can directly compile each of `queuetest.cpp` and `apptest.cpp` to the executable separately.

Submission:
You should submit your work via Canvas. You should pack `queuetest.cpp`, `apptest.cpp`, `myqueue1.h`, and `myqueue2.h` into a single `.zip` file to upload to Canvas. The `.zip` file should be named as `a2_yourNetID.zip`, such as `a2_gwc38.zip`.

Sample tests:
Note that success in getting the following test results does not guarantee the correctness of your work and thus does not guarantee a satisfactory grade, while failure in getting the following test results probably does indicate flaws in your work and you may lose points.

Test case 1 for `queuetest`:

```
Testing the basic functions of your queue...
Please enter the max size/capacity of your queue: 4
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: e
Please enter an integer-type value you want to enqueue: 1
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: e
Please enter an integer-type value you want to enqueue: 3
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: e
Please enter an integer-type value you want to enqueue: 5
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: e
Please enter an integer-type value you want to enqueue: 7
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: e
Nothing can be enqueued since the queue is full!
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: d
1 has been popped out.
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: d
3 has been popped out.
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: d
5 has been popped out.
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: d
7 has been popped out.
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: d
Nothing has been popped out since the queue is empty!
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: s
```

Test case 2 for `queuetest`:

```
Testing the basic functions of your queue...
Please enter the max size/capacity of your queue: 3
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: e
Please enter an integer-type value you want to enqueue: 7
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: e
Please enter an integer-type value you want to enqueue: 5
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: e
Please enter an integer-type value you want to enqueue: 3
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: e
Nothing can be enqueued since the queue is full!
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: d
7 has been popped out.
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: e
Please enter an integer-type value you want to enqueue: 1
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: d
5 has been popped out.
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: d
3 has been popped out.
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: d
1 has been popped out.
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: d
Nothing has been popped out since the queue is empty!
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: e
Please enter an integer-type value you want to enqueue: 2
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: e
Please enter an integer-type value you want to enqueue: 4
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: d
2 has been popped out.
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: d
4 has been popped out.
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: d
Nothing has been popped out since the queue is empty!
Please enter 'e' for enqueue, 'd' for dequeue, and 's' for stop: s
```

Output for `apptest`:

```
Testing the CountStudent function...
Student Queue: 1 1 0 0
G-Mouse Stack: 0 1 0 1
Number of students who are unable to have the gaming mice is: 0
Testing the CountStudent function...
Student Queue: 1 1 1 0 0 1
G-Mouse Stack: 1 0 0 0 1 1
Number of students who are unable to have the gaming mice is: 3
Testing the CountStudent function...
Student Queue: 1 1 0 0 1 1 0
G-Mouse Stack: 0 1 0 1 0 0 0
Number of students who are unable to have the gaming mice is: 2
```