Cosc 4740
Program 1

Due Date: Sep 08.

Goal: A programming warm-up and learn git exercise.
Grade: 50 points.
Writing in C++ and it must be able to be compiled and run on the department linux systems.

QueueArray Class: To be used in Homework #2
Implement a variable sized generic **QueueArray** class that implements an array of queues, and exports the following functions:

1. *int* **Enqueue**(*type* item, *int* index): Enqueues *item* in the queue of array index *index*. Returns 1 if *item* is successfully enqueued; -1, if *index* is out of range; 0, otherwise
2. *type* **Dequeue**(): Dequeues an item from the first non-empty queue in the array, *i.e.*, from the non-empty queue at the lowest numbered index in the array. Returns the dequeued item, if there is at least one item in the queue array; 0 otherwise.
3. *int* **Qsize**(*int* index): Returns the number of items in the queue at array index *index*; -1, if index is out of range.
4. *int* **Asize**() : Returns the size of the array.
5. *int* **QAsize**(): Returns the total number of items stored in the array of queues.
6. *type** **Qstate**(*int* index): Copies all items stored in the queue at array index *index* in an array. Returns a pointer to this array; NULL, if index is out of range.

Implement the **QueueArray** class in such a way that an example **QueueArray** object *intqueue* consisting of an array of 10 integers queues may be constructed using the following statement:

QueueArray <int> *intqueue* (10); //c++ version.

Write or use my driver file for testing. Your file is to be called, queue_array.h When turning in the program and output, use the original driver file (no changes will be needed to the file).

Turning in the Assignment: (a hard copy and soft copy)

Hard Copy:

- A cover page
  - Your Name, cosc 4740
  - a repo name (see github and below for your repo name),
  - LAB Section #
  - Statement of help. (not okay leave blank)
- Print the output of the driver.

Soft copy:

1. Use this link to create your repo https://classroom.github.com/a/7MA7iR4d
2. Use the git_checkin_requirements document to check in the assignment to the repo. This is 10 points out of 50. Failure to read directions is not an excuse.
3. Edit the readme.md file, add the following:
    - Name
    - How to compile the code, if there is no makefile.
    - List anything that doesn't work (that you know of)
4. Lastly ensure everything has uploaded to the github website and not just the local repo.

Code will be graded on correctness, comments, and coding style.