

CSCI-2270

Assignment 3

Instructor: Boese

Stacks and Queues

Objectives

- Create, add, and delete, and work with a stack using a linked-list
 - Create, add, and delete, and work with a stack using an array
 - Create, add, and delete, and work with a queue using a linked-list
 - Create, add, and delete, and work with a queue using an array
-

Background

Stacks and Queues are both data structures that can be implemented using either an array or a linked list. Therefore, to fully understand how they work you will complete the implementation for both a Stack and a Queue using both an array and a linked-list.

Assignment

You will be working with a todo list of items that will either go on to a stack or on a queue. A todo item for all implementations will include the following struct:

```
struct TodoItem
{
    std::string todo;
};
```

You will also be working with classes in this assignment. The header files for all four implementations are provided on the course website.

Program Specifications

- Use the starter code on the website and do not modify what is provided. These are the header files; you will write the implementation files and submit those only.

Provided Files (do not submit):

- HW3-Todo-StackArray.hpp
- HW3-Todo-StackLinkedList.hpp
- HW3-Todo-QueueArray.hpp
- HW3-Todo-QueueLinkedList.hpp

Files You Create (do submit):

- HW3-Todo-StackArray.cpp
- HW3-Todo-StackLinkedList.cpp
- HW3-Todo-QueueArray.cpp
- HW3-Todo-QueueLinkedList.cpp
- Do NOT add a main method to any of your submitted files.
- DO write your own test drivers to test your code, but do not submit them.
- Your code needs to be readable, efficient, and accomplish the task provided.

- Make sure you delete your dynamically allocated memory in the appropriate methods!
- When working with array-based implementations, there is a max size available (set to 5 for this assignment in the header files). Display an error message if it is full:
 “Stack full, cannot add new todo item.”
 Or “Queue full, cannot add new todo item.”
 Note – this does not apply to linked-list implementations.
- If the stack or queue is empty when you try to pop or peek it, display an error message:
 “Stack empty, cannot pop an item.”
 “Stack empty, cannot peek.”
 “Queue empty, cannot dequeue an item.”
 “Queue empty, cannot peek.”
- Make sure your code is commented enough to describe what it is doing. Include a comment block at the top of the .cpp file with your name, assignment number, and course instructor, and anyone you worked with.
- You must fill in the functions as specified. You do not need any additional functions.
- To untar a tarball file, most systems you can double-click and it will uncompress or from the command-line use: `tar -xvf filename.tar`

Example to-do items could include:

Take out the garbage
 Clean the dishes
 Make bed
 Do laundry
 Buy detergent

Submitting Your Code:

Zip your **HW3-StackArray.cpp**, **HW3-StackLinkedList.cpp**, **HW3-QueueArray.cpp**, and **HW3-QueueLinkedList.cpp** file into a zipped file named **HW3_[yourFirstName]_[yourLastName].zip** and submit to Moodle.