CSC 111 Computer Science II

Project – Stack & Queue ADTs Juggler

Due April 24, 2024

This project is to write a program to test the functionality of stackADT (textbook p500-p501) and queueADT that implemented using linked lists. You will read data from the test.dat file.
When you read *add number*, the number should be inserted to both stack (push) and queue (enqueue) named inStack and inQueue, respectively.
When you read *delete*, data should be deleted from both inStack (pop) and inQueue (dequeue). Data popped from inStack should be enqueued in outQueue. Similarly, data dequeued from inQueue should be pushed in outStack.

An example with a given test.dat:

test.dat
add 3
add 2
delete
add 5
delete
add 10
add 7
delete
add 0
delete
add 11
delete
delete
delete

Input format:
./sqjuggler test.dat

Output format:
outStack: 11 0 7 10 5 2 3
outQueue: 2 5 7 0 11 10 3

Programming Language and Environment:

- Language: C
- Platform: Sloop/Clipper

Submission:

Make the followings as "s24csc111p_lastname.tar.gz" in all lower cases and then submit to **ship.drlee@gmail.com**. The subject of your email is "s24 csc111 p lastname" in all lower cases.

- readme
  - project information
  - platform/language
  - list of source code
  - how to compile
  - how to run
  - known bugs

- documentation
  - description of the program
  - algorithm
  - algorithm analysis
  - numerical experiment
  - numerical analysis
  - conclusion

- all source and extra files (sqjuggler.c, sqjuggler.h, queue.c, stack.c, makefile, etc)

Each file must include student, course, and instructor information. Here is an example.

```
/*
 * Project: Stack & Queue ADTs Juggler
 * Programmer: Your name
 * Course: CSC111
 * Professor: Dr. Lee
 * File Created: April 10, 2024
 * File Updated: April 10, 2024
 */
```

Marking Criteria (100 marks total):

- Submission of required files only, with the information of student, course, and instructor on all submitted files (5 marks)
- Warning free compilation and linking of executable with proper name (10 marks)
- sqjuggler program (60 marks)
- Readability, suitability & maintainability of source codes (10 marks)
- Documentation (15 marks)