Programming Assignment 5

Stacks – Queues

Due Date : Monday March 28th – 2016 @

3:30 pm for section 2 & 5:00 pm for section 3

For this assignment you will implement stacks and queues ADT. A stack and a queue are the same if they have the same number of elements and the elements at the corresponding positions are the same. The program displays a menu on the screen allowing the user to enter the elements of stack and queue. The program terminates when the user enters 9.

Note :

You are not allowed to use stack / queue libraries. You must implement all stack / queue functions and operations .

Must use templates to implement this program.

Validations:

The program accepts only integer data type when selecting from the menu (1 or 9). Everything else should be rejected with an invalid option message.

Style Guidelines:

At the beginning of your program (and **before** the #include statement), include the following :

Header comments (file documentation block) should be at the top of each file and should contain: Author / s, Due Date, Assignment Number, Course number and section, Instructor, and a brief description of the purpose of the code in the file. For example :

```
//      Roster Number / s :      xxxxxxxxx
//
//      Author / s : (Your name  here!!)
//      Due Date :
//      Programming Assignment Number  5
//
//      Spring  2016  -  CS 3358  -  Your Section  Number
//
//      Instructor:  Husain Gholoom.
//
//       <Brief description of the  purpose of the program>
```

Variable names :

- Must be meaningful.
- The initial letter should be lowercase, following words should be capitalized, no other caps or punctuation (i.e. `weightInPounds`).
- Each variable must be declared on a separate line with a descriptive comment.

Named constants :

- Use for most numeric literals.
- All capitals with underscores (i.e. `TX_STATE_SALES_TAX`)
- Should occur at top of function, or global (only if necessary)

Line length of source code should be no longer than 80 characters (no wrapping of lines).

Indentation :

- Use 2-4 spaces (but be consistent throughout your program).
- Indent blocks, within blocks, etc.
- Use blank lines to separate sections.

Comments for variables :

All variable definitions should be commented as follows:

```
int  gender;    // integer value for the gender,
                // 1 = Male , 2 = Female ,
```

Sample Run :

```
*** Welcome to My stack / Queue Program ***

The function of this program is validate that
a stack and a queue are identical.
Stacks / queues are  same if
they have the same number of elements
and the their elements at the
corresponding positions are the same.

Select from the following menu
1.   Enter Stack / Queue Values.
9.   Terminate the program.    1

Enter Stack Values terminated by ;    123;

Enter Queue Values terminated by ;    123;


Stack and Queue are identical


Select from the following menu
1.   Enter Stack / Queue Values.
9.   Terminate the program.    1

Enter Stack Values terminated by ;    123;

Enter Queue Values terminated by ;    132;


Stack and Queue are not identical
```

```
Select from the following menu
1.   Enter Stack / Queue Values.
9.   Terminate the program.    1

Enter Stack Values terminated by ;    123;

Enter Queue Values terminated by ;    12345;


Stack and Queue are not identical


Select from the following menu
1.   Enter Stack / Queue Values.
9.   Terminate the program.    2

Invalid Option

Select from the following menu
1.   Enter Stack / Queue Values.
9.   Terminate the program.    9


*** Program is terminated. Written by Husain Gholoom ***
```

Rules :

1. Your program **must compile** and run.
2. The entire program must be **documented . Must also document all templates functions , all templates implementations of functions , and the main program.**
3. You must use the appropriate libraries (except stack / queue library) in writing this program.
4. You must name your program as :
   - **Gholoom_Husain_PG5_SQ.cpp**
5. **Use your Last_name / First_name when naming your program and also in the program output**
6. Every one must upload the electronic version of the program no later than the starting of class time on the due date. **No late assignments will be accepted.** **DO NOT** send your assignment solution via email. **Group members must upload identical copy of the assignment.**

   To upload your program , go to the CS department's website, click on resources , then select homework upload.
7. You must **also** turn in hard copy of your source code no later than the starting of class time on the due date . should the hard copy consist of more than one page , then , the hard copy must be **stapled**. if you are unable to turn in a printout during class, you can take the program to the computer science department and hand it to the front desk personal (Comal 211) before the deadline. Make sure that the front office stamps the program. Make sure that include the date and time. Finally ,make sure that they place the program in my mailbox. **Only one copy per group.**

   DO NOT slide your program under my office door – It will **NOT** be accepted
8. We will compile and run these programs using CodeBlocks or Eclipse softwares that are available in our CS Labs . Make sure your programs will compile and run on CS Lab computers.
9. **Violating any item from the above rules will result in Grade ZERO for the entire assignment. NO EXCEPTIONS.**