

**LANGARA COLLEGE**  
*DEPARTMENT OF COMPUTING SCIENCE AND INFORMATION SYSTEMS*  
**CPSC 1160 - ALGORITHMS AND DATA STRUCTURES I**  
**Assignment 07 – Lab 07**  
**November 02, 2017**

**Instructions**

- This assignment is worth 10 points, and is due on November 05 at 04:00 PM.
- This assignment has multiple parts; **the first part is required to be submitted during the lab hour on November 2<sup>nd</sup>.**
- All the program files (.cpp and .h files) are required to be put in a folder named **Lab07**; the whole folder then should be submitted as a single zipped file on D2L.

**Part1: The Stack ADT [5 points]**

- a) Implement the Stack ADT according to the following class diagram:

Stack<T>
-elements: T *
-top: int
+Stack() +Stack(size: int) +push(newElement: T) : void +pop() : T +top() : T const +isEmpty() : bool const +isFull() : bool const

**Notes:**

- It is assumed that the stack size cannot be changed after the construction.
  - Your code is required to throw exceptions when applicable.
- b) Add the following function to your ADT to double the stack size when stack is full and a push attempt is done.
- c) Re-implement your stack to allow expanding the stack in a constant time.
- d) Use your Stack ADT to write a program that reads a parenthesised expression from the keyboard, and checks if all the parentheses are balanced, and prints an appropriate message on the screen.

**Part2: The Queue ADT [5 points]**

- a) Implement the Queue ADT as a contiguous structure; Use the circular implementation, and utilize two variables front and rear pointing to the first element in the queue and the first spot after the last element, respectively.

- b) Implement a DeQueue ADT that allows insertion and deletion to/from the both sides of the queue.
- c) Provide a space **and** time analysis for the DeQueue ADT.