CIS 200

Programming Assignment 1

**Submission of Your Work**

You need to prepare and submit ONE SINGLE MS Word document to Canvas (in your lab section) as LastName\_FirstName\_Project1.doc. It must contain:

* Your NAME only on page 1
* For the project task,
  + Source code. Copy/Paste your final source code. You must include standard “comment header” even if code is provided. *Do Not* paste a snippet of your source code, it must be copy/pasted.
  + Initial test plan. After reading the question requirements, but **before** beginning any coding, create the test case table, below, completed through column Expected Output. Include in your report.
  + Final test plan. Write your program then complete the **test table** with actual output results and include in your report *AFTER* your source code.
  + Output results. Paste in a snippet of output showing results for **every listed test case in your final test plan**, labeled with test case #

Test Table:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test # | Valid / Invalid Data | Description of test | Input Value | Expected Output | Actual Output | Test Pass / Fail |
| 1 |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |

* Add / delete rows from Test Table as necessary
* Modify column widths as necessary
* Test both valid and invalid input
* Test for every output expected
* If failure is an expected output and it happens then that test Passes
* Any test that fails means the program must be fixed so that it passes the test
  + Failing tests need a new test row, ie 1a, 1b, etc, showing corrections from original

You are to create your own version of a string class named myString.

myString will store a string of characters (use dynamic memory – ptr to char)

Include a reference “status variable” to indicate error state.

You will also need to supply the following methods as part of your class:

myString( string) – you may have a constructor that takes a string parameter

size() – returns how many characters are in the string (empty string is size zero)

addStart(myString) – adds the string in the input parameter to the front of current string

addEnd(myString) – adds the string in the input parameter to the end of the current string

partString(startPos, length) – returns as myString that portion from startPos for length given. Handle startPos < 0 & > size; startPos = size returns null string

replPartString(myString, startPos) – replaces characters starting at startPos with parameter, which may be <, >, or = in size to what is replaced

replWholeString (myString) – replaces current string data value with parameter string

compareString(myString) – compare current value of string with parameter string. Returns 0 if strings match, otherwise return character position (NOT index) where mismatch occurs. If parameter is first alphabetically then return is positive, otherwise negative.

initString() – resets/initializes string to null string

setString(string) – assign to myString the parameter string

getString() – returns string of data from myString

printStringScreen() – prints myString data value to the monitor (value only, nothing else)

numericString() – returns Boolean telling if data value is an integer or real (signs, decimal point. etc.), or not

alphabeticString() – returns Boolean telling if data value is all alphabetic characters

You may use the C++ string class **only** for user input and the setString & getString methods. Only size from the string library may be used, and only within setString method.

Write a main that will test all of functionalities of the myString class, displaying the actions both on the screen and to an output file. Main must use a myString method to write the results to the file.

Create an output file log of actions, which must show action, original value of myString, parameters/results, success/error message. You determine appearance of log file – format into columns for readability -- and what the error messages will be.