**Submission of Your Work**

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

* Your NAME only on page 1
* For ***each*** question, include in this order:
  + Specify the question number, if more than one question.
  + 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 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
  + Unless you don’t have time to finish your testing, there should never be a Fail in final test plan

**Note: you must perform file exist and empty file checks in your programs.**

**Program 1**:

Let A be an array of *n* elements. Write a *template* function, minMaxFunc(…), which takes two parameters:

1. an unsorted array of type <typename T> as an input parameter
2. whether you want the minimum or maximum value

Use TWO stacks within the function to determine the minimum or maximum value and return that element:  
 Stack 1 (LINKED LIST) holds values in ascending/descending order.   
 Stack 2 (ARRAY) is a “temporary” work stack as you insert values to keep them in order.

Once all values have been processed you must also print the values in the stack. In this question, you may use global constant SIZE=5. You must write all functions you call (PUSH, POP, PEEK, ISFULL ISEMPTY). You must also write a main() to test the following data read from data file (data.dat) [two sets of content. Work with the first set then replace with the second set]:

4 1 13 3 2

* 1. 4.1 8.1 5.2 2.3

Finally, with no changes to your function, instantiate your function for type char and report the outcome when using the first test data above.

**Program 2**:

Given the following segments of code, add a copy constructor to both classes and thoroughly test it using a main routine.

**class A {  
 int valuea;  
 public:  
 int getValuea() const { return valuea; }  
 void setValuea (int x) { valuea = x; }** // copy constructor   
 **};**

**class B : public A {  
 int valueb;  
 public:  
 int getValueb() const { return valueb; }  
 void setValueb (int x) { valueb = x; }** // copy constructor   
 **};**

Modify the definitions of the above two classes to *template* classes, i.e., “int valuea;” and “int valueb;” should be changed to a generic data type. After the modification, you need to test these two template classes with the following test cases:

1. Create an instance of class B with float data type (valuea = 1.34 and valueb = 3.14)
2. Create an instance of class B with integer data type (valuea = 1 and valueb = 3)
3. Create an instance of class B with char data type (valuea = ‘a’ and valueb = ‘c’)
4. Create an instance of class B with string data type (valuea = “good” and valueb = “morning”)
5. Create an instance of class B with Date data type (valuea = {27,10,2015} and valueb = {2,11,2015}). Date is a struct and defined as:

struct Date {  
 int day;   
 int month;  
 int year;  
};

You must use cout from main to display values stored in the class. No other testing is required.