CS 150 Lecture 9 Exercises

Complete each of the exercises below and upload to Canvas before the deadline.

# Exercise 1 - Extreme Values

Work along with your instructor to complete this assignment, which recapitulates the material from the textbook section covering libraries and testing. You are going to create a library named **exlib**, that has two functions, **min()** and **max()**, in the **namespace** **ex**, which return the appropriate value from three input parameters of type **double**.

* Place your code in the folder **exlib**
* Open the file named **client.cpp**. In the **runTests()** function:
* Add **beginFunctionTests**/**endFunction** test blocks for **max()** and one for **min()**
* Create a file-wide constant for **EPSILON** as **1.0e-14**
* Write three different **assertDoubleEquals()** tests for each function. Make sure that you can manually calculate the expected value.
* Type **make** **client**. **IT SHOULD FAIL TO COMPILE** because you have only tested your functions; you haven't yet declared or defined them. Shoot a screenshot of the error message you get.

| *Copy and paste a screenshot of the error message you get at this point* |
| --- |

## Create Your Header File

## Now, create a header file, extreme.h containing the prototypes for the two functions. Make sure you add header guards to the file, the namespace, and document each of the functions using the Doxygen documentation we discussed. Then, #include the file in the appropriate place in client.cpp and type make client. Your program should now compile, but not link. When it does, paste a picture of the error message.

| *Copy and paste a screenshot of the error message you get at this point* |
| --- |

## 

## Create Your Implementation File

Now, create an **implementation** file, **extreme.cpp** containing the implementation of your two functions. Make sure you **#include** the header file (and any other library functions that you use, such as **string** or **cmath**). Create a **makefile**. You should be able to now build and test your program with **make test**. Shoot screenshots of the source code for all three files as well as a run of the program into the text areas below.

| *Copy and paste a screenshot of the source code for extreme.h* |
| --- |

| *Copy and paste a screenshot of the source code for extreme.cpp* |
| --- |

| *Copy and paste a screenshot of the source code for client.cpp* |
| --- |

| *Copy and paste a screenshot of running client.cpp (testing your code)* |
| --- |

| *Copy and paste a screenshot of your makefile* |
| --- |

| *Copy and paste a screenshot of the generated documentation for your functions* |
| --- |

# 

# Exercise 2 - Overloaded Functions and Default Arguments

[Follow this link](https://codecheck.it/files/2009011757733s37hay2r9qcvp0y6yt8cul) to complete two versions of the **void** function **print()**. The first prints a **string**, and the second prints a **double**. Both have a default **bool** argument for whether to print a newline. The second has an additional default argument indicating how many decimals to display. Examine the tester program to see how the functions are used.

| *Copy and paste a screenshot of the source code for overload.cpp* |
| --- |

| *Copy and paste a screenshot of your testing results* |
| --- |