CSE 122 – HW 0 Rubric

Student Name: Marci McBride

Grader Name: Matthew Crepeau

Questions	Score	Total
Code Requirements		
1.) FizzBuzz		

Requirements:
• Diff with solution fizzbuzz.sol (**4 pts**)
   o This is all or nothing.
• Used the algorithm correctly (**10 pts**)
   o Correct code, even if the output doesnt match with diff.
• Naming is correct (**1 pts**)
   o Source code "**fizzbuzz.c**"
   o Out put file "**fizzbuzz.out**"
	15	15

2.) Function Timing		

Requirements:
• Solution should be similar to **sum.sol (8 pts**)
   o This **can't** be diffed.
   o Everything but the last column should be the same as sum.sol
• Used the algorithm correctly (**6pts**)
   o Used a **for** loop
   o Used unsigned ints for input and output
   o Time should be printed to 24 decimal places
• Naming is correct (**1 pt**)
   o Source code " **sum.c "**
   o Output file "**sum.out**"
	15	15

3.) Function Timing – Non Iterative		

Requirements:
• Solution should be similar to **sum_no_for.sol (8 pts**)
   o This **can't** be diffed.
   o Everything but the last column should be the same as sum_no_for.sol
• Used the algorithm correctly (**6pts**)
   o Uses the equation to calculate the sum (no loops)
   o uses unsigned ints for input and output
	15	15

CSE 122 – HW 0 Rubric

• Naming is correct (**1 pts**)
     o Source code " **sum_no_for.c "**
     o Output file "**sum_no_for.out**"		

4.) Written Answers – **to_for_or_not_to_for.txt**		

Requirements:
 • Ask this for the questions 2 and 3. (**5 pts each**)
     o What is the largest k before overflow?
     o Is k the same for both functions?
     o Which is preferred? Why?
 • Should be reasonable answers (no for loop preferred, overflow with the for loop).
	9	10

5.) Cube Root		

Requirements:
 • Diff with solution cube_binary.sol (**4 pts**)
     o This is all or nothing.
 • Used the algorithm correctly (**10 pts**)
     o Finds the cube root of the numbers 1-100
     o Accuracy is to 9 decimal places
     o Uses **binary search** (6/10 pts)
 • Naming is correct (**1 pts**)
     o Source code "**cube_binary.c**"
     o Output file "**cube_binary.out**"
	11	15

6.) Newton's Square Root		

Requirements:
 • Diff with solution newton_sqrt.sol (**3 pts**)
     o This is all or nothing.
 • Used the algorithm correctly (**6 pts**)
     o Uses Newton's Method (8/10 pts)
     o Accuracy is to 9 decimal places
 • Naming is correct (**1 pts**)
     o Source code " newton_sqrt **.c**"
     o Output file " newton_sqrt.**out**"
	7	10

CSE 122 – HW 0 Rubric

7.) Newton's Cube Root		

Requirements:
• Diff with solution newton_cube.sol (**3 pts**)
   o This is all or nothing.
• Used the algorithm correctly (**6 pts**)
   o Uses Newton's Method (8/10 pts)
   o Accuracy is to 9 decimal places
• Naming is correct (**1 pts**)
   o Source code " newton_cube **.c**"
   o Output file " newton_cube.**out**"
	7	10

Code Requirements		

• Tar ball MUST be named (2/10 pts)
   cse122_fname_lname_hw0.tar.gz
• Must use a Makefile (2/10 pts)
• Must have doxygen comments (3/10 pts)
• Must have correct coding style (3/10 pts)
   o 8 space tabs
   o Proper bracket placement
   o etc.
	5	10

	84	**100**

Comments:

In to_for_or_not_to_for.txt you said that there was no overflow in sum_no_for, which is incorrect. It overflows at the same spot. (-1 pt)

cube_binary.out did not match correct output exactly (incorrect first column, plus an extra space in the last column) (-4 pts)

Incorrect first column in newton_sqrt.out (-3 pts)

Same problem with newton_cube.out (-3 pts)

No doxygen comments (-3 pts)
No Makefile (-2 pts)