CS 1400 – Fall 2019
Assignment #6

Introduction

You will complete two programs to give you experience with:

- Software Development Process
- Functions
- Strings
- Objects
- Formatting

Task 1

Programming Exercise 3.5 (from book)

Save your code in the Task 1 coding area. Make sure to read the rubric to see all requirements.

Task 2

Programming Exercise 3.9 (from book)

Your program should look like the following (which is different than the book). Note that the first 5 lines are asking for input from the user. Make sure you have them enter tax rates in the appropriate form. Also note where those rates are displayed and the form they're shown in. Save your code in the Task 2 coding area.

```
Enter employee's name: Chad Mano
Enter number of hours worked in a week: 40
Enter hourly pay rate: 12.75
Enter federal tax withholding rate (ex. 0.12): 0.11
Enter state tax withholding rate (ex. 0.06): 0.07

       CHAD MANO PAY INFORMATION

                  Pay
               Hours Worked:          40
                   Pay Rate: $     12.75
                  Gross Pay: $    510.00

               Deductions
Federal Withholding (11.0%): $     56.10
   State Withholding (7.0%): $     35.70
           Total Deduction: $     91.80

                 Net Pay: $    418.20
```

Make sure to read the rubric to see all requirements.

Software Development Lifecycle Plan

Follow the software development lifecycle in your assignment. Put the steps as comments at the top of your file below your name and other header comment information. This should be simple. Similar to what we have done in class. Submit a report with:

1. Requirements specification
2. System analysis
3. System design
4. Implementation – This is just your code. Don't include it plan area
5. Testing
    1. Write at least 2 test cases you will run to assess your program
    2. Each test case should have
        1. Input
        2. Expected Output
    3. Report if you ran into any problems and what you did to fix them. Write "Passed" if all tests passed on your first try.

Rubric

Task 1: Software Development Lifecycle	3.0 pts
Task 1: User is prompted for input and can enter values properly	2.0 pts
Task 1: All tested input results in proper output These are tests the graders will use, not the student's tests	5.0 pts
Task 1: Output is rounded to 5 decimal places using the round() function.	5.0 pts
Task 1: Math module functions are used for tan() and pow() with different techniques for importing. You must call the tan() and pow() functions specifically as math.pow() and tan() This means you should have two import statements.	10.0 pts
Task 2: Software Development Lifecycle Plan	3.0 pts
Task 2: User is prompted for input and can enter values properly	2.0 pts
Task 2: Headings of sections are centered and spaces between sections	5.0 pts
Outcome Task 2: Main output header is all capitalized	5.0 pts
Task 2: Output numbers are right aligned with decimals aligned	5.0 pts
Task 2: Colons and $ signs are aligned	5.0 pts

Task 2: No arithmetic calculations inside the print string (use variables)	10.0 pts
Task 2: A single print statement is used for output	10.0 pts
Task 2: All tested input results in proper output These are tests the graders will use, not the student's tests	20.0 pts
Outcome Both Tasks: Proper naming conventions for variable names	5.0 pts
Both Tasks: Proper comments at the top of the file with your name, assignment number, and class.	5.0 pts

Helpers

Remember that you can find solutions to the even programming exercises online. Check Canvas for a link. These are suggestions for you to do. They are not part of the assignment, and you do not have to turn them in.

Exercises: 3.4, 3.6, 3.8