Assignment 1 - Payroll Calculation

Due Jan 31 by 11:59pm **Points** 20 **Submitting** a file upload
File Types h and cpp **Available** until Feb 3 at 12:01am

This assignment was locked Feb 3 at 12:01am.

ST. CHARLES COMMUNITY COLLEGE

CPT-182 - Programming in C++

Programming Assignment - Payroll Calculation (20 Points)

(Number in Question Bank: Assignment **1.1**)

Program Overview

In this assignment, you are going to write a C++ program that prompts the user for his/her *last name*, *SSN* (last **4** digits), *hourly pay rate*, and *number of working hours*. In an attractive format (see below), your program will display all the **input data** and the following **calculated data**.

Item	Definition
Gross Pay	*Hourly Pay Rate* × *Number of Working Hours*
Federal Withholding Tax	**15**% of *Gross Pay*
State Withholding Tax	**5**% of *Gross Pay*
Net Pay	*Gross Pay* − *Federal Withholding Tax* − *State Withholding Tax*

Program Input

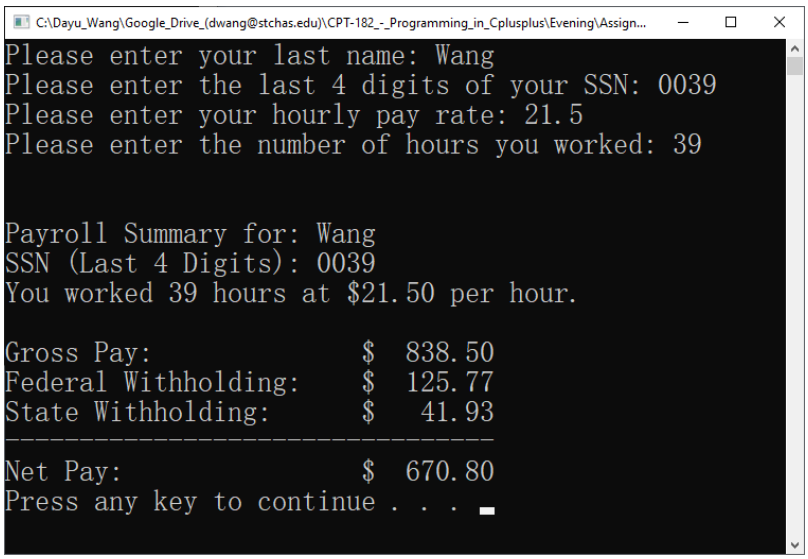
Your program asks the user to provide the following **required** information via keyboard:

- **Last Name**. This is a **`string`** that does **not** contain any ~~spaces~~.
- **Last 4 Digits of SSN**. It is your responsibility to choose an appropriate variable type to store the user input data. For example, if the last **4** digits of the user's SSN is "**0036**", your program should write it in the console as **"0036"** later, **not** ~~"36"~~.
- **Hourly Pay Rate** (in USD). This is a number that may contain a fraction part (e.g., **15.5**, **21.75**). So, what variable type should be used here?
- **Number of Working Hours**. This is a **non**-negative integer. Let's assume that the user always works for whole hours. Please use **unsigned integer** (**not** ~~integer~~) to store the user input data.

Please note that the program user **cannot** see or understand your code. Therefore, you need to give sufficient **screen prompts** to let the user clearly know what information to provide in each step.

Program Output

- Your program writes all the output messages and values in the console.
- The output format of your program should be the same as the sample output below:



- In your program output, the **dollar sign** (**'$'**) should be nicely aligned (see above).
- In your program output, the **digits** in your output values should be nicely aligned (see above).
- For each floating-point output value, you need to keep **2** decimal places. You may need some **extra self-study** here.

Assignment Submission and Grading (Please Read)

- Please upload all your **.h** (if any) and **.cpp** files (**not** the ~~entire Microsoft Visual Studio project folder~~) on Canvas.
- Before the assignment deadline, you can submit your work unlimited times. However, only your latest submission will be graded.
- At least **20**% of your code should be **comments**. All variable, function (if any), and class (if any) names should "make good sense". You should let the grader put **least effort** to understand your code. Grader will **take off points**, even if your program passes all test cases, if he/she has to put extra **unnecessary** effort to understand your code.
- Please **save a backup copy** of all your work in your computer hard drive, in case technical errors occur on Canvas.
- Your program will be graded (tested) using multiple different sets of valid input data to check whether it can generate the expected (correct) output. As long as the input values are valid,

your program should generate correct output. In other words, your program should work for **any** valid input data, **not** just the sample input data provided in the assignment instructions.

- In this class, you can assume that all input data are always **valid** and **have correct format**. You do **not** need to deal with ~~invalid input~~ or ~~error handling~~.
- Your work will be graded after the assignment deadline. All students will receive their assignment grades at (almost) the same time.