

CST 183 Programming Assignment #4 (20 points)

Purpose: The purpose of this assignment is to use an IDE (Integrated development environment) to create a well-documented Java program that utilizes user interactions, loops, an input file, and an output file.



Specifics: Create a program designed to handle multiple employees, calculate net pay after taxes, and outputs a Payroll Report for your company, **Delta Technical Services**.

Delta Technical Services has 10 employees and utilizes a pay code strategy for the following scenarios: Paycode 'M' gets minimum wage, Paycode 'O' gets minimum wage + \$3.00, and Paycode 'T' gets minimum wage + \$6.00. Your program will use the current minimum wage for Michigan and adjust as required for a given pay code.

Your program needs to open and read an input file of employees, pay codes, and hours worked. The input file can be found here : [payrolldata.txt](#)

Look at the input file and study the way it is formatted. Each line of input contains a name, an employee ID number, a pay code, and the number of hours worked. You are not allowed to change this file.

Set up a loop in your program to process the data file. For each record in this file, process/validate the input for each employee and calculate gross and net pay based on the following tax rates: (You should declare a constant for each of these.)

Fed Tax Rate = 15%

State Tax Rate = 5%

Social Security Tax Rate = 7.5%

Medicare Tax Rate = 1.5%

Any hours worked over 40 hours should be paid at time and a half.

Once you have calculated gross pay, apply the above tax rates to calculate net pay.

You will need to use input validation to confirm that you have a valid pay code and you should also check that no employee worked more than 80 hours for the week. If you encounter an invalid record, make sure you acknowledge/highlight the record in your output payroll report. Output to your output file, "payroll_report.txt," a complete report that lists the number of employees processed, the total payroll for this pay period (e.g. How much was paid to all

employees), and a nicely formatted “check stub” for each employee detailing information similar to the following example:

Here is a sample output: (yours should look similar with appropriate formatting)

```
*****
Paycheck info for employee: Lane Holden
Employee Code:          124578
Total Hours Worked:    45
Payrate:                $ 8.15
Earnings Before Taxes: $ 387.12

    Federal income tax:  $ 77.43
    State income tax:    $ 19.36
    Social security tax: $ 29.03
    Medicare tax:        $  5.81

Earnings After Taxes:   $ 255.50
*****
```

Step 1: Develop and document your program by writing pseudocode. Examine your pseudocode for logic errors and correct as needed. (Do this before you write the program!)

Step 2: Study loop and file concepts, specifically how to use input and output files.

Step 3: Type your code into the IDE (e.g. JGRASP)

Step 4: Compile your code and execute. You may need to fix errors. Look at your output and format appropriately. Do a test run with the provided data and correct any errors you encounter.

Deliverables: Submit the Java source code **AND** the output file to the **Program 4 Dropbox** within the Delta eLearning System.

Create a hardcopy for turn-in and grading containing the following:

- a) Title Page
- b) Pseudocode for program
- c) Source Code (copy and paste from your IDE)
- d) Output file (copy and paste into your turn-in doc or print and attach)