CS1110 – Fall 2016 – “Work Schedule Optimizer” app
Assignment 5 - 100 points

<u>Objectives – to practice these concepts</u>:

- *Use a top comment in program, **including your name, lab section day/time, asgn number and name of application***
- *Learn to program using camel case naming convention for variables, methods, and classes*
- *Using appropriate data types for variable,*
- *Learn to work with single, double and parallel arrays*
- *Learn to calculate statistics on the array values such as min/max/total/average*
- *Learn to write modular program*
- *Learn to Javadoc, HINT: look at wrapper classes methods for basic variable types (int & double) i.e. use parseDouble & parseInt methods*
- *Use incremental development*
- *Print output report to console (print & println methods & concatenation)*

<u>PROJECT OVERVIEW</u>

The text data file contains information about an employee’s work schedule over three months. Project management office (PMO) is trying to optimize plan so that it can devise a more efficient program that supports work-life balance. To do so, the project manager in charge of the operation would like to know following statistics about the employee work habits.

Employees were given full freedom regarding work schedule. They had to choose among 3 available shifts for the day.

- Morning shifts - 8:00am to 12:00 noon
- Afternoon shifts - 12:00 (noon) tp 16:00 pm
- And Evening Shifts - 16:00pm to 20:00pm

Also, each day of the week had different payout giving more weight to employees who work during the weekend then regular weekdays.

Table 1 Weekly Pay Scale

Monday	**Tuesday**	**Wednesday**	**Thursday**	**Friday**	**Saturday**	**Sunday**
4xBase Pay	Base Pay	0.75xBase Pay5	Base Pay	1.5xBase Pay	2xBase Pay	5xBase Pay

- Calculate during which shifts did the employee worked min, max and total number of hours
- Calculate min, max and total pay earned for working on each of the shifts forgiven weekdays
- Find out which shift were most rewarding i.e. you made maximum pay over the entire schedule.
- Find out which shift were least rewarding i.e. you made minimum pay over the entire schedule
- Find the average salary for the employee based on the three months’ work schedule

Procedure:

A start-up assignment code base is given along with this description to help kick start the initial coding hurdles. The program is divided into four classes. Each class’s description and purpose is as follows

1. WorkScheduleStatisticsDriver.java

This class is the main driver for the program. It provide implementation for method ***populateDataStore,*** that should read and parse the data in from the data file “timesheets.txt”. Format of the data file is given in input section. In short data file is divided into five columns. First column gives the day of the week, second is the date and followed by 3 columns during which the employee worked. It is assumed that employee did not work if the values in the correspondingcolumn is empty.

2. DataStore.java

This class acts as our main database for the program. It consists of parallel arrays to keep track of

- Shifts during which employee worked for the week.
- Total number of hours worked during each shift
- Total shif earnings

“populateParallelArry” *method should be used inside* WorkScheduleStatisticsDriver->***populateDataStore*** method to fill values in parallel arrays. Implementation of ***calcNumHrsInShift*** is already given which takes shift start and end time as input to calculate the shift length in hours.

3. WorkScheduleStatisticsWorker.java

This is the main worker for the program. It should implement methods to do all the important calculations. As an example template method ***totalShiftHrs*** is given.

4. Constants.java

All the constants used the application are given in this class as static variables.

INPUT

sunday	9/1/2016					16:00	20:00
monday	9/2/2016	8:00	12:00				
tuesday	9/3/2016	8:00	12:00	12:00	16:00		
wednesday	9/4/2016					16:00	20:00
Thursday	9/5/2016	8:00	12:00				
friday	9/6/2016			12:00	16:00		
Saturday	9/7/2016					16:00	20:00

OUTPUT

```
--------------------------------------------------------------------------------------------------

Total number of morning shifts worked                                        156

Total number of afternoon shifts worked                                      120
…. <<<Other output as suggested in project description>>>
Total number of evening shifts worked                                        128

Total number of evening shifts worked                                      117.600
--------------------------------------------------------------------------------------------------
```

APPENDIX A

Incremental Development

There are several ways/styles to write a program. The incremental build model is a method of software development where the product is designed, implemented and tested incrementally (a little more is added each time) until the product is finished. It involves both development and maintenance (Wikipedia!). i.e., “always keeping the program in a working state (which produces SOME output”

1. Write some code to carry out a step in the algorithm
2. “Compile” (Build) it and correct all syntax errors
3. Run program to check thatoutput is correct thus far
4. Repeat steps 1,2,3