System Test Plan - `Paychecks` Program

Requirements

Raleigh's Parks and Recreation Department hires landscapers to care for and maintain the city's parks.

An employee has one of three skill levels; each with a hourly pay rate:

```
Skill Level - Hourly Pay Rate ($)
---------------------------------
Level 1 - $19.00
Level 2 - $22.50
Level 3 - $25.75
```

All employees may opt in for insurance, which results in a deduction from their pay check.

```
Deduction - Weekly Cost ($)
-----------------------------------
Option 1 - Medical Insurance - $24.50
Option 2 - Dental Insurance - $15.30
Option 3 - Vision Insurance - $5.25
```

Employees at skill level 3 may also opt to place up to 6% of their gross pay into a retirement account.

The following information is printed about the employee's pay check: 1) employee's name 2) hours worked for a week 3) hourly pay rate 4) regular pay for up to 40 hours worked 5) overtime pay (1.5 pay rate) for hours over 40 worked 6) gross pay (regular + overtime) 7) total deductions 8) net pay (gross pay – total deductions). If the net pay is negative, meaning the deductions exceeds the gross pay, then an error is printed.

The `Paychecks` program prompts the user for an input file with information about zero or many employees' paychecks and writes all generated paycheck information to the user specified output file. It inputs employee information through a file and outputs results to a file and error messages to the console. Error messages that are printed to the console can include input files that do not exist, improperly formatted/structured input files, invalid input values for employees, and invalid net pay values. Each file can contain information for multiple employees. Each line of the file will represent a single employee. Each line is structured in a tab delimited manner:

```
 name TAB level TAB hours worked TAB medical TAB dental TAB vision TAB retirement %
```

Such that medical, dental, and vision are boolean values that represent whether the employee has the given type of insurance.

The following commands compile and run the `Paychecks` program:

```
$ javac -d bin src/Paychecks.java
$ java -cp bin Paychecks
```

The test files should be in the test-tiles directory. There are input files and expected output files. The input files contain valid or invalid values and the expected output files record the expected results of the execution. You can compare the actual results of execution with the expected results visually, of the `diff` command may be used to compare the actual results and expected results. If there are any differences, they will be printed to the console. If there are no differences, the prompt will appear and there will be no output.

```
$ java -cp bin Paychecks
Input File: test-files/input_level01.txt
Output File: test-files/output_act_level01.txt
…
$ diff test-files/output_exp_level01.txt test-files/output_act_level01.txt
$
```

System Test Plan

The text in bold in the **Description** column represents the test inputs to the program. You can stop program execution by pressing Ctrl+C.

System Test Plan File IO

Test ID	Description	Expected Results	Actual Results

Multiple Level 1 Employees	**Preconditions:** Paychecks program started
	Input File: **test-files/input_level01.txt**
	Output File: **test-files/output_act_level01.txt**
		`Error reading line: Alice Anderson 1 0 true true true`
		`Error reading line: Bob Baker 1 1 false true true`
		
		See output_exp_level01.txt for expected file output	

Multiple Level 2 Employees	**Preconditions:** Paychecks program started Input File: **test-files/input_level02.txt** Output File: **test-files/output_act_level02.txt**	`Error reading line: Alice Anderson 2 0 true true true` See output_exp_level02.txt for expected file output	
Multiple Level 3 Employees	**Preconditions:** Paychecks program started Input File: **test-files/input_level03.txt** Output File: **test-files/output_act_level03.txt**	`Error reading line: Alice Anderson 3 0 true true true 0` See output_exp_level03.txt for expected file output	
Single Employee	**Preconditions:** Paychecks program started Input File: **test-files/input_single.txt** Output File: **test-files/output_act_single.txt**	See output_exp_single.txt for expected file output	

Multiple Invalid Employees	**Preconditions:** Paychecks program started
	Input File: **test-files/input_MULTIPLE_INVALID.txt**
	Output File: **test-files/output_act_MULTIPLE_INVALID.txt**
		`Error reading line: Alice Anderson 3 0 true true true`
		`Error reading line: Bob Baker 3 1 N true true 1`
		`Error reading line: Carol Cole 3 10 true false Y 1`
		`Error reading line: Frank Frankenstein 3 40 true false false 7`
		`Error reading line: George George 3 41 false true false -1`
		`Error reading line: Hilda Henderson 4 50 false false false 3`
		`Error reading line: Hilda Henderson 0 50 false false false 3`
		`Error reading line: A very long name and that's it`
		
		See output_exp_MULTIPLE_INVALID.txt for expected file output	

Incorrect input filename	**Preconditions:** Paychecks program started Input File: **test-files/input_level** <Stop program>	`Cannot read file.` `Input File:`	