Assignment 1 - Pay Cheque

Background

You are working for a small, family-run business as the 'Tech Guy'. The company currently uses an Excel spreadsheet to calculate the pay for each employee. This works pretty well, but the owner wants you to create a 'C' program, based on the spreadsheet, that can then be used to pipe output to a file for storage.

Hours Worked	46.5	← Enter values here
Hourly Wage	$ 12.50	← Enter values here
Gross Pay	$ 581.25	
Vacation	$ 23.25	
Total Income	$ 604.50	
Annual Income	$ 15,717.00	
Annual CPP	$ 641.39	
Annual EI	$ 248.33	
Federal Tax	$ 2.04	
Provincial Tax	$ 10.92	
Total Tax	$ 12.95	
CPP	$ 24.67	
EI	$ 9.55	
Net Pay	$ 557.33	

2020 Pay Information	
# of Pays in a Year	26
Employment Amt	$ 1,245.00
CPP Exception Amt	$ 3,500.00
Fed Personal Amt	$ 13,229.00
Prov Personal Amt	$ 11,481.00
Federal Tax Deduction	15%
Provincial Tax Deduction	8.48%
CPP Deduction	5.25%
EI Deduction	1.58%

Figure 1 - Excel Spreadsheet Example

```
$ ./paycheck
Enter the number of hours worked: 46.5
Enter the wage per hour: $12.50

Gross Pay:       $ 581.25
Vacation:        $  23.25
Total Income:    $ 604.50

Federal Tax:     $   2.04
Provincial Tax:  $  10.92
Total Tax:       $  12.95

CPP:             $  24.67
EI:              $   9.55

Net Pay =        $ 557.33
$
```

Figure 2 - Solution Example

Task

1. You have been given an Excel spreadsheet (Figure 1) as an example of a working payroll 'program' from your instructor. Your first task is to examine the spreadsheet to try and understand how it works as you have **no other information from which to build your application**!
2. Determine which cells are data entered from the user, which cells are constant values and which cells are calculated values.
3. Determine what variables you will need, what the data types should be and what are good names for the variables (use a naming convention!).
4. Determine how many different calculations are required and what the mathematical formula is for each calculation.
5. Then, develop a 'C' program that performs similarly to the Excel spreadsheet based on the information you have gathered. Your program doesn't have to look the same, but it does need to calculate everything the same!
6. Test your program by comparing its output with the output generated from the Excel spreadsheet.

Notes

- Figure 2 is an example of what your program **could** look like when completed.
- Submit the completed assignment to the appropriate BrightSpace dropbox **before** the due date.
- Refer to the attached rubric to ensure you are completing all the required elements of the assignment.

Criteria	Unsatisfactory	Acceptable	Good	Exceptional	Marks
	0	1	2	3	
Input & Output	- the wrong information was selected for input or nothing was input - the output was unclear and currency was not formatted	- some information was correctly input - all information is output, but it needs to be formatted better	- input is handled - output is clear and appealing, but a small change would improve it - currency is formatted	- user input is correctly handled - output to user is very clear and visual appealing - currency is well formatted	
Variables & Constants	- all variables use the wrong data type - very poor names were used - no naming convention was used	- some data types were chosen well - a few variables were well-named - a naming convention was not well used	- most data types were chosen well - some variables were well-named - naming convention was mostly followed	- most appropriate data types used - well-named - uses appropriate naming conventions	
Income	- all income amounts were calculated incorrectly or not at all	- some of the pay calculations were correct - a couple of errors in pay calculations exist	- most of the pay calculations were correct - a small error in pay calculations exist	- gross and holiday pay is correctly calculated - total income and estimated annual income are also correctly calculated	
Tax	- all tax amounts were calculated incorrectly or not at all	- some of the tax calculations were correct - a couple of errors in tax calculations exist	- most of the tax calculations were correct - a small error in tax calculations exist	- fed, prov and total tax are correctly calculated - annual tax is correctly calculated	
CPP, EI and Net Pay	- all deductions and the net pay were calculated incorrectly or not at all	- either CPP or EI was calculated correctly - a couple of errors exist in the calculations	- most of the CPP and EI calculations were correct - a small error exists in CPP, EI or Net Pay	- CPP and EI are correctly calculated - annual CPP and EI are correctly calculated - net pay is correct	
Comments	- little to no comments used	- comments are used, some are meaningful and easily understood - some files and functions have headers	- comments are used extensively, most are meaningful and easily understood - most files/ functions have headers	- not over/under commented - comments are meaningful and easily understood - files/functions have headers - program is self-documenting	
Formatting	- was not indented - there were too many deviations in indentation or placement of braces - very difficult to read	- source code formatting was fairly consistent, but contained some inconsistency with whitespace, brackets, etc	- source code formatting was very consistent with respect to whitespace, brace brackets, parentheses, etc	- standard indentation is used throughout without deviation - placement of braces consistent - easy to read	
				Total	**21**