

## CS 215 – Spring 2021

### Project 1

75 pts – Due Feb 12

#### Learning Objectives:

- Use of MS Visual Studio C++ to write a programming solution.
- Use of simple user input/output and formatting in a program.
- Use of decisions in a program.
- Use of loops and looping strategies in a program.
- Writing data to a text file in a program.

#### General Description:

Write a small application to process payroll for a group of employees. The program starts by printing a logo as shown in the example (you may be more creative), which includes *your name*. It should then open a file called "payroll.txt" for writing. When the file fails to open, the program prints "Unable to open payroll.txt" and exits the program (use **return 0**; inside an if statement in main()).

```
+-----+
|               Payroll Express               |
|               by XXXXXX                     |
+-----+
```

The program should ask the user to enter the report date (which may have spaces) and the number of employees. The number of employees should be validated to be between 0 and 50, repeating the error message and prompt as shown in the example, until a valid answer is entered.

```
Enter date for report:    Dec 25, 2020
Enter number of employees: -3
Enter a number 1 to 50
Enter number of employees: 51
Enter a number 1 to 50
Enter number of employees: 4
```

For each employee, the following processing occurs:

- Prints a blank line followed by "Employee n:" where n=1,2,3...
- The program asks for the employee's name (which may have spaces)
- It asks for the employee type (H, S or C: for Hourly, Salaried or Contractor), using a validation loop to repeat the question as shown when an invalid answer is entered.

```
Employee 1:
Enter name: Vixen Reindeer
H=hourly S=salaried C=contractor
Enter employee type: lazy
Invalid entry. Enter H, S or C!
Enter employee type: l
Invalid entry. Enter H, S or C!
Enter employee type: h
```

Only the upper case value of the first letter entered is checked for correctness. Here is some C++ example code on how to get the uppercase of the first value entered:

```
string answer;
char f;
cout << "Enter something: ";
if (cin.peek() == '\n') cin.ignore();
getline(cin, answer);
f = toupper(answer[0]); // answer[0]=first letter of answer
```

- When the employee is Hourly, the program asks for the hourly wage and number of hours worked. It calculates the gross as hours worked X wage when the number of hours worked is 40 or less. The wage is increased by a factor of 1.5 for any hours worked over 40 (overtime).

```
Enter hourly wage:    22.50
Enter hours worked:   45
```

Example:

wage = 10.00, hours = 45 [so 40 hours regular wage plus  
 $(40 * \text{wage}) + (1.5 * \text{wage} * (\text{hours} - 40))$  5 hours overtime wage]

- When the employee is Salaried, the program simply asks for the gross salary.

```
Enter employee type: s
Enter gross salary: 1200.00
```

- When the employee is a Contractor, the program asks for the contract pay.

```
Enter employee type: c
Enter contract pay: 575.00
```

- Inputs should line up as shown by the yellow line in the examples (see full sample execution below).

- Taxes and net pay are calculated for Hourly and Salaried employees (**but not Contractors**) as:
  - o local tax = 1.5% of the gross
  - o state tax = 6 % of the gross
  - o fed tax = fed tax rate of the gross. The rate is calculated as follows:

When the gross is:	Fed Tax Rate
0.00 to 459.99	2%
460.00 to 899.99	8%
900.00 to 1499.00	14%
1500.00 or higher	22%

- o net pay is the gross minus the taxes
- o **all of these rates and limits must be declared as constants in the program and used. NO NAKED LITERALS!** (except 1 and 0).
- Once all values are calculated, they are written to the data file on one line. See the sample output file for order and formatting:
  - name (field width 20)
  - gross & net (fld width 8)
  - taxes (field width 7)
- Details on formatting:
  - o all floating point numbers are printed two digits past the decimal
  - o all individual calculations are rounded off to two digits:  
 $f = \text{round}(f * 100) / 100;$
  - o numbers in columns should line up as shown on the decimal points.
  - o numbers are right justified; the name is left justified.

Payroll Report for Dec 25, 2020						
Employee	TYP	Gross	Local	State	Fed	Net Pay
Vixen Reindeer	H	1068.75	16.03	64.13	149.63	\$ 838.96
Dasher Reindeer	H	740.00	11.10	44.40	592.00	\$ 92.50
Elf Ona Shelf	S	1200.00	18.00	72.00	168.00	\$ 942.00
Jack Frost	C	575.00	0.00	0.00	0.00	\$ 575.00
TOTALS:		3583.75	45.13	180.53	909.63	\$2448.46
Highest paid employee: Elf Ona Shelf(\$1200.00)						
Hourly Employees: 2						
Salaried Employees: 1						
Contractors: 1						

Once data for all employees has been entered, a report is written to the output file including:

- totals for the gross, taxes and net columns for all employees, under the columns as shown.
- name and gross of the employee paid the most, with gross in parenthesis as shown.

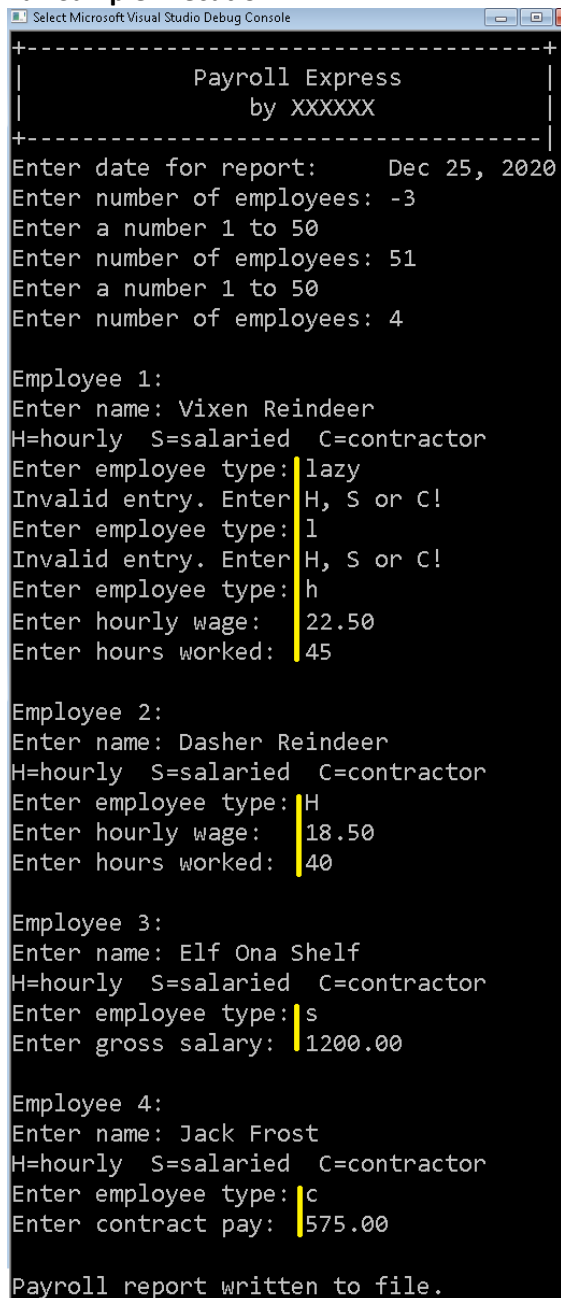
- count of the number of hourly, salaried and contracted employees, one per line as shown.

The program ends by closing the output file and reporting completion to the user as shown in the sample execution.

### Submission:

Submit *only* your **.cpp** file in Canvas; other files are not needed.

### Full Sample Execution:



```

Select Microsoft Visual Studio Debug Console
+-----+
|               Payroll Express               |
|               by XXXXXX                     |
+-----+
Enter date for report:      Dec 25, 2020
Enter number of employees: -3
Enter a number 1 to 50
Enter number of employees: 51
Enter a number 1 to 50
Enter number of employees: 4

Employee 1:
Enter name: Vixen Reindeer
H=hourly S=salaried C=contractor
Enter employee type: lazy
Invalid entry. Enter H, S or C!
Enter employee type: l
Invalid entry. Enter H, S or C!
Enter employee type: h
Enter hourly wage: 22.50
Enter hours worked: 45

Employee 2:
Enter name: Dasher Reindeer
H=hourly S=salaried C=contractor
Enter employee type: H
Enter hourly wage: 18.50
Enter hours worked: 40

Employee 3:
Enter name: Elf Ona Shelf
H=hourly S=salaried C=contractor
Enter employee type: s
Enter gross salary: 1200.00

Employee 4:
Enter name: Jack Frost
H=hourly S=salaried C=contractor
Enter employee type: c
Enter contract pay: 575.00

Payroll report written to file.

```