

Note :

1. Solve any 4 questions for a level 1, 6 questions for level 2, 8 questions for level 3, all questions for level 4
3. Use individual files for each problem. Label all your program files in this manner: question1.c, question5.c, etc
4. Make a special folder called "ICS2O-Assignment-LASTNAME" and store your files in there. (You will submit this)
5. Due at the end of Thursday's class, Thursday December 17th, 2015

//Remember: All code will be assessed for efficiency, proper formatting, and completeness

PROBLEMS

1. Accept any integer from the user. Display whether the number is divisible by 100 or not.

2. Write a program:

- a) to check whether an entered number is odd / even.
- b) to calculate sum of three numbers.

3. Accept principal amount, rate of interest, and duration from the user.

Display Interest Amount and Total Amount (Principal + Interest).

Example: \$20000 principal amount, 5% interest, 10 years = \$10000 interest, \$30000 total

4. Accept the salary of an employee from the user. Calculate the net salary on the following basis (must account for income tax, how much will they get after taxes deducted? And how much for each tax)

| Salary | Federal Tax | Provincial Tax |
|---------------|-------------|----------------|
| 1 - 10000 | 10% | 0% |
| 10001 - 25000 | 15% | 10% |
| 25001 - 40000 | 20% | 20% |
| 40001 - 80000 | 25% | 25% |
| 80001+ | 30% | 30% |

5. Write a program to accept 10 values in an integer array. Display the number of odd, even, and negative numbers.

6. Accept a numeric date from the user (year, month, day). Display the date in words. If the numbers are not valid, display an error message.

Example: Year: 2015, Month: 7, Day: 27. Output: July twenty-seventh, two thousand and fifteen

7. Display all prime numbers between 50 and 150.

8. Accept any two strings from the user. Display whether both the strings are equal or not. (do not use any built in functions, must check this yourself character by character!)

9. Write a program to print the following patterns (using loops):

| | | | |
|----|-----------|----|-----------|
| a) | 1 | b) | 1 |
| | 1 2 | | 2 2 |
| | 1 2 3 | | 3 3 3 |
| | 1 2 3 4 | | 4 4 4 4 |
| | 1 2 3 4 5 | | 5 5 5 5 5 |

10. Using arrays (bonus points for using structures!), create a method of storing a student number, name, address and average grade for three students. Have the user enter this information, and print out the information at the end of your program.