CS 36 Programming Assignment 2
(20 Points)
(if, switch,loops - Lesson 1 to 32)

1. You must turn in your program listing and output for each program set. Each program set must have your student name, student ID, and program set number/description. Late homework submission will not be accepted for whatever reasons you may have.

*********_**for this homework, you are to submit your Program sets to Canvas**_ under Homework 2 link*********

   a. Name your files: HW2_PS1_ lastname_firstinitial.c for Program Set 1 and HW2_PS2_ lastname_firstinitial.c for PS2 and so on. PS means program set. If there are two program sets you will submit two files one for each program set. Example if your name is Joe Smith then the filename will be HW2_PS1_smith_j.c
   b. You must submit your homework by the deadline at the specified upload link on Canvas under homework 2. If the deadline is past, Program Sets will not be graded. Homework submitted via email attachment, comment in Canvas, Canvas message, or by any other method is not accepted and will be given a zero for no submission.
   c. if you do not follow instructions on file naming provided in this section you will receive a zero for the question you did not correctly name the file.
   d. It is your responsibility to check if your homework is properly submitted to Canvas.

2. Please format your output properly, for example all dollar amounts should be printed with 2 decimal places. Make sure that your output values are correct (check the calculations).

3. Use only the 'tools' in the topics we covered from Lesson 1 to 32 only. **No functions, arrays, structs, #include<math.h>, #include<stdlib.h>, #include<iostream>, #include<string.h>, scanf("%[^\n]%*c",varname); and any topics not covered from lesson 1 to lesson 32 except to declare strings char strVar[n]. A zero grade will be given if any of the listed is used.**

4. Each student is expected to do their own work. **IF IDENTICAL PROGRAMS ARE SUBMITTED, EACH IDENTICAL PROGRAM WILL RECEIVE A SCORE OF ZERO.**

Grading:

If a program does not compile the program set will receive a zero score. If the program compiles and run but does not have proper declaration of variables, syntax, logic, and displays the correct output (given in the sample test runs) with proper formatting as specified in the question, the program set will receive a zero score. Each program set must compile and must run correctly with proper declaration of variables, syntactically, logically, and display the correct output (given in the sample test runs) as specified then you will receive the full points for that question. Then points will be deducted for not having proper:

a. Comments 1 pt deducted for each infraction
   - Title Banner --Your name, description at the beginning of each program set.
   - Short description of the what each section or function of your codes do.

b. Consistency/Readability 2 pt deducted for each infraction
   - Spacing(separate each section of codes with a blank line
   - Proper Indentation
   - Proper naming of variables no a, b, c – use descriptive and mnemonics)

accounts payable
acct_pay

c. Required elements 1 pt deducted for each infraction
   - proper formatting for output when specified
   - all monetary values must be in 2 decimal places when specified in given sample test runs

d. Use only 'tools' in the topics that have been covered in class. For example, for this homework we have not covered functions, arrays, structs. So, if you use a function, array, struct you will receive a **zero** for the whole program set. See also item 3 above.

e. Output (you **must provide the specified number of test runs or your program set will receive a zero score**)
   - to be displayed at the end of the program listing(codes) and commented
   - if no output(test runs) is provided from your uploaded file, a zero will be given for that program set.
   - must have the number of test runs as specified in each program set.
   - must use the data for test runs when they are provided for you in the question.

Points will be deducted from items a. to e. above until your Program Set reaches zero points.

USE ONLY THE TOOLS YOU HAVE BEEN TAUGHT IN CLASS ONLY(Lesson 1 to Lesson 32), IF YOU USE ANYTHING WE HAVE NOT COVERED YET YOU WILL RECEIVE A ZERO FOR THAT PROGRAM SET (eg functions, arrays, structs…)

The break, continue and goto C commands are not allowed to be used. A zero will be given if your program contains a break, continue or goto command. The break command is allowed only as part of the switch statement.

Programs that do not compile

If you submit a program that does not compile, I will not be able to execute it and, thus, will not be able to evaluate its correctness. It is your responsibility to turn in code that compiles and run. I reserve the right to assign zero points on the **Correctness and Robustness** portions of the project score for a program that does not compile, regardless of how much code there is or how close it is to working. This rule is in place not only because it is impractical for me to evaluate the correctness of non-compiling code, but to impress upon you the importance of this issue: in the "real world," there is no excuse for submitting code that does not compile.

At first blush, this sounds like a harsh policy, but it is not as difficult as it sounds to make sure your program compiles, if you develop your program one small piece at a time, ensuring that the program compiles and runs (and, ideally, runs correctly) before moving on to the next piece. It is never a good idea to write hundreds of lines of code before attempting to compile and run a program, yet many students do it. The "code-everything-then-compile" approach, which can be successful for very small programs, such as those you may have written in courses prior to this one, does not scale up well to problems the size of those you will face in this course. I urge you to consider working through your programs step-by-step, compiling and running them as you make progress.

Program Set 1 (10 points)

Use only if and switch statements. No loops for Program Set 1

Write a program that will allow the user to select a filing status through a menu, take Taxable Income(TI) as input, and calculate the amount of tax due.

1. Provide a menu for user to choose filing status. Where:

```
1 is for Single
2 is for Married Filing Jointly
3 is Married Filing Separately
4 is Head of Household
5 Exit Program
```

   - The program will expect the user to only enter a single digit from 0 to 9 or a single letter a to z or A to Z. The user will not enter anything more than a single digit or letter like 1234 or 1abc or abc.

- If the user enters any value other than 1 to 5, your program will print a message " `You entered a wrong status. Program Exit . . .`" and the program will exit. See sample test runs 3 and 4.
- If the user enters the value 5, then the program will print " `Exit program...`". See sample test run 2.
- If the user enters status 1 or 2 or 3 or 4, then user will input the Taxable Income(TI) and calculate and print the tax amount.

2. Allow the user to enter the Taxable Income (TI) (test only from 0 to 1 million range. You do not need to test income outside that range)

3. Calculate the Taxes for the Taxable Income that was entered based on the Tax Rate Schedule below.

4. Your output should look exactly like the test run sample shown below. Run the program 4 more times for Test Runs 5, 6, 7, 8 to be included for your submission. You must also include the sample Test Runs 1 to 4 for submission.

Page continues below…

The Fictitious Federal Income Tax Rate Schedule

Condition		**Tax**
		Filing Status : **Single**
0 < TI ≤ $24,000		15% of TI
$24,000 < TI ≤ $58,000		$3,600.00 + 28% of (TI - $24,000)
$58,000 < TI ≤ $121,300		$13,162.00 + 31% of (TI - $58,000)
$121,300 < TI ≤ $263,750		$32,738.50 + 36% of (TI - $121,300)
$263,750 < TI ……		$84,020.50 + 39.6% of (TI – 263,750)
		Filing Status : **Married Filing Jointly**
0 < TI ≤ $40,100		15% of TI
$40,100 < TI ≤ $96,900		$6,015.00 + 28% of (TI - $40,100)
$96,900 < TI ≤ $147,700		$21,919.00 + 31% of (TI - $96,900)
$147,700 < TI ≤ $263,750		$37,667.00 + 36% of (TI - $147,700)
$263,750 < TI ……		$79,445.00 + 39.6% of (TI – 263,750)
		Filing Status : **Married Filing Separately**
0 < TI ≤ $20,050		15% of TI
$20,050 < TI ≤ $48,450		$3,007.50 + 28% of (TI - $20,050)
$48,450 < TI ≤ $73,850		$10,959.50 + 31% of (TI - $48,450)
$73,850 < TI ≤ $131,875		$18,833.50 + 36% of (TI - $73,850)
$131,875 < TI ……		$39,722.50 + 39.6% of (TI – 131,875)
		Filing Status : **Head of Household**
0 < TI ≤ $32,150		15% of TI
$32,150 < TI ≤ $83,050		$4,822.50 + 28% of (TI - $32,150)
$83,050 < TI ≤ $134,500		$19,074.50 + 31% of (TI - $83,050)
$134,500 < TI ≤ $263,750		$35,074.00 + 36% of (TI - $134,500)
$263,750 < TI ……		$81,554.00 + 39.6% of (TI – 263,750)

Sample runs (use other status and TI values to check calculations):

Sample Test Run 1(user enters a valid status)

```
***********Menu***************
1) Single
2) Married Filing Jointly
3) Married Filing Separately
4) Head of Household
5) Exit

*******************************

Enter status : 1
Enter your taxable TI: $50000

The taxes owed are: $10880.00
```

Sample Test Run 2 (user enters 5 to exit program)

```
***********Menu***************
1) Single
2) Married Filing Jointly
3) Married Filing Separately
4) Head of Household
5) Exit

*******************************

Enter status : 5

Exit program...
```

Sample Test Run 3 (user enters an invalid status - single letter)

```
***********Menu***************
1) Single
2) Married Filing Jointly
3) Married Filing Separately
4) Head of Household
5) Exit

*******************************

Enter status : a

You entered a wrong status. Program Exit . . .
```

Sample Test Run 4 (user enters an invalid status)

```
***********Menu***************
1) Single
2) Married Filing Jointly
3) Married Filing Separately
4) Head of Household
5) Exit

*******************************

Enter status : 7

You entered a wrong status. Program Exit . . .
```

You must provide a total of 8 test runs for this Program Set. Use the four sample test runs provided plus four more of your own by testing the other valid status and check if the tax calculations are correct for the income the user input.

For Program Set 2 and 3

***Pick the best loop, must use the sample test runs format exactly as shown,

Program Set 2 (5 points)

Write a C program to calculate salary raise for employees.

If salary is between	$ 0 < $ 30000	the rate is 7.0%
If salary is between	$ 30000 <= $ 40000	the rate is 5.5%
If salary is greater than	$ 40000	the rate is 4.0%

1. Let the user enter salary. **Allow the user to enter as many salaries as the user wishes until the user enters a negative salary to quit for example -1(sample test run 1). User can also decide to quit immediately after starting the program(sample test run 2). Pick the proper loop.**

2. Calculate the raise, new salary, total salary, total raise, and total new salary.

Continue on next page . . .

3. Sample input and output (leftmost column in blue is user input):

Test Run 1

```
                          Salary      Rate      Raise      New Salary
-------------------------------------------------------------------------------------------
Salary : 25000
                        25000.00      7.00    1750.00       26750.00
Salary : 30000
                        30000.00      5.50    1650.00       31650.00
Salary : 35000
                        35000.00      5.50    1925.00       36925.00
Salary : 40000
                        40000.00      5.50    2200.00       42200.00
Salary : -1
-------------------------------------------------------------------------------------------
Total                  130000.00                7525.00     137525.00
```

Test Run 2

```
                          Salary      Rate      Raise      New Salary
-------------------------------------------------------------------------------------------
Salary : -1
-------------------------------------------------------------------------------------------
Total                       0.00                 0.00           0.00
```

Provide your own test runs 3, 4, and 5

Your output of **5** test runs must be exactly the same format as specified above. Use the data provided above from the sample test run 1 and 2. Your test run 2 should enter the salary of -1 as first value, your program should exit. Provide your own data for the rest of the 3 test runs using different numbers.

Continue next page for Program set 3 . . .

Program Set 3 (5 points)

1. Using the same information as in program set 1. **This program will first ask the user to how many salaries do you want to enter. Your program will then execute the exact number of salaries the user wanted to enter. Pick the proper loop. You cannot use the same loop as Program Set 1. Using the same type of loop as Program set 1 will result in a zero for Program Set 2**

2. Sample input and output (leftmost column is user input in blue):

Test Run 1

How many salaries do you want to enter? **4**

	Salary	Rate	Raise	New Salary
Salary : **25000**				
	25000.00	7.00	1750.00	26750.00
Salary : **30000**				
	30000.00	5.50	1650.00	31650.00
Salary : **35000**				
	35000.00	5.50	1925.00	36925.00
Salary : **40000**				
	40000.00	5.50	2200.00	42200.00
Total	130000.00		7525.00	137525.00

Provide your own test runs 2 and 3

Your output of 3 test runs must be exactly the same format as specified above. Use the data provided above for sample test run 1. Provide your own data for the rest of the 2 test runs.