

CIDM 4360/5360 Assignment #1: C#-review

Total Points: 50 (5% of the course's final grade)

Due: Wed Sep 4, 11:59pm

Objective : Assess your ability to write C# functions/methods to perform specific tasks

Skills needed: In this programming assignment, you'll need to Know how to:

- 1- use VS Code to create/edit and run C# programs
- 2- declare methods
- 3- Pass arguments to methods
- 4- Call methods
- 5- Return one or more than one value from methods
- 6- Use loop and switch statements
- 7- Split strings using * and = as delimiters
- 8- Parse (convert) string to float
- 9- Given a problem description, identify the tasks and write C# code using methods to handle each task.

Requirements & Submission:

1. You **must** add your name as a comment in the first line of the source code
2. You need to submit complete, error free, and runnable C# program source code
3. Submit on WTCClass ("Resources >> Assignments >> Assignment#1)
4. The C# source code file name must be your `first name + "-" + Last name + "-PA1.cs"`.
5. No public or global variables should be used. You need to consider passing arguments between the methods according to the ways described in the lecture. i.e. all variables should be declared inside the methods and passed to other methods by value/ref/out as needed.

Grading:

1. You'll earn full points for correct method of each task
2. Non-executable methods will get 0 points
3. Partial credit will be given for executable non-complete tasks

Description:

Write a program that tests the user skills in multiplication by allowing him to enter multiplication expression from the console in the form: $\text{Number1} * \text{Number2} = \text{UserAnswer}$ (e.g. $5.5 * 6.5 = 35.75$, or $3 * 5 = 10$). Then the program should evaluate the user expression by calculating the correct answer as ($\text{correctAnswer} = \text{Number1} * \text{Number2}$) and compare it with the user's answer. The program should keep track of the count of expressions entered and the count of correct user answers to them. The user gets 1 point if his answer was correct, and 0 points if it was not. The program should repeatedly display a menu of choices until the user chooses to exit the program.

- **The program should work as follows:**

- 1- Initially set the score and expressions counters to 0's
- 2- Then displays a menu with the following choices :
 - a. Enter expression
 - b. Check the answer
 - c. Display score
 - d. Exit
- 3- When the user selects option (a), he will be asked to enter a multiplication expression with an answer. Then the program should increment the expressions counter if correct expression is entered.
- 4- When the user selects (b), the program evaluates the expression to get the correct answer and compares it with the user's answer, and display a message (either "correct" or "incorrect") to inform the user about his answer. If the user's answer was correct, the score counter is incremented. The program should not increment the score counter more than once for each correct answer (i.e. if the user selects option (b) multiple times without changing the expression.)
- 5- When the user selects option (c), the program will display the score and expressions counter, for example : Your score is 10 out of 12
- 6- The program exits when the user selects option (d)

- **The program must contain the following methods, and all should be called from the main method with the appropriate arguments:**

- 1- **[10 points] Menu:** to display the choices and reads user's selection and return it to the main method
- 2- **[10 points] *ReadExpression*:** prompt the user to enter an expression, then reads an expression from the console, and then parse the expression to extract the two numbers (number1 and number2) and the user's answer (userAnswer) and return them to the main method.
- 3- **[10 points] *CheckAnswer*:** this method uses the numbers (number1, and number2) in the user expression to calculate the correct result and compare it to the user's answer, display a correction message, then returns 1 if user's answer was correct, and 0 if not. The correction message should be in the form "your answer is/is not correct" $x*y=\text{correct answer}$.
- 4- **[10 points] *DisplayScore*:** this method will display the score and expressions counter
- 5- You can add/use any method as you need

- **[10 points]** The *Main* method should contain a switch statement with a case for each of the menu options. Following is a skeleton for the *Main* method.

```
static void Main(string[] args)
{
    char ch;

    do
    {
        ch = menu();
        switch (ch)
        {
            case 'a':
                // write code to do something for option (a) here

                break;
            case 'b':
                // write code to do something for option (b) here

                break;
            case 'c':
                // write code to do something for option (c) here

                break;
        }

    } while (ch != 'd');
}
```

[+ 5 points] extra credits:

- Verification of user inputs (i.e. check if the user input the expression in the proper format to prevent the code from behaving unexpectedly or crashing). For example, replacing * with other symbols will cause the parsing of the input expression incorrect.
- Organization and explanation using comments of the code

[- 5 points] Penalty:

- For violating each of the submission requirements above (up to -20 points)

Best of success