Computer Programming Laboratory		
Laboratory 3: Expression and Operator	**School of Information Technology**	
Name: Thu Rein Oo	**ID: 6731501110**	**Section:3**

Course Learning Outcomes (CLO)

1. **CLO1.1:** Students do their own work, do not present the work of others as their own, do not cheat on exams, and are responsible for completing their work.
2. **CLO2.3:** Students understand the principles of programming including emerging technology, and how to utilize them together.
3. **CLO6.1:** Students are able to complete their assigned work for each particular concept.

Lesson Learning Outcomes (LLOs)

1. **LLO1:** Understand the concept of expressions in C# [Aligns with CLO2.3]
2. **LLO2:** Identify and use different types of operators in C# [Aligns with CLO2.3]

Lab Objectives

- To be able to develop programs to solve simple problems with expressions involving multiple operators [Aligns with CLO2.3 and CLO6.1]

Remark:

- **CLO LLO Lab Objectives**
- Students are expected to complete the lab on their own, with no plagiarism or cheating. [Aligns with **CLO1.1**]

Instruction:

- Complete and answer all required questions: **In-class (Question 1-2) and Homework (Question 1-2)**

Submission:

- Save this document file that contains all required answers as "**Lab03_YourID.pdf**" and submit it to the MFU LMS system by the deadline.

Remark:

- **Not accept late submission!!!**

Programming TIP

Simple program often takes the following step:

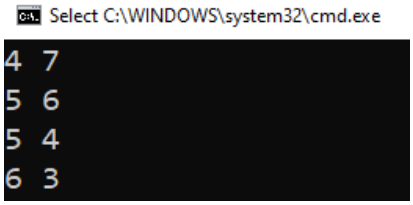
1. Prepare: declare variables and explain the program to the user
2. Input: prompt for and get input from user
3. Process: perform the task at hand
4. Output: display the results

We can call this structure PIPO as abbreviation.

[In-class] Question 1

Complete the given program to display the numbers below.
Hint: use the prefix and postfix to i and j. For example, you have to use i++, ++i, - -i, i - -.

Expected Results




```
class PrePostFix
{
    public static void Main(String[] arg)
    {
        int i = 3, j = 7;
        Console.WriteLine(++i + " " + j--);
        Console.WriteLine(        ??     );
        Console.WriteLine(        ??     );
        Console.WriteLine(        ??     );

    }
}
```

Answer: source code

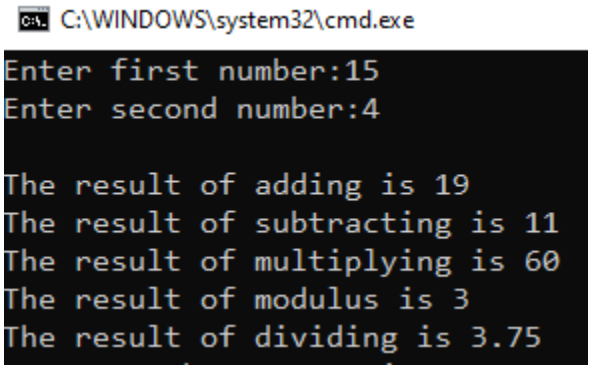
```
using System;

class PrePostFix
{
   public static void Main(string[] arg)
   {
      int i = 3, j = 7;
```

```
        Console.WriteLine(++i + " " + j--);
        Console.WriteLine(++i + " " + j--);
        Console.WriteLine(i-- + " " + --j);
        Console.WriteLine(++i + " " + --j);
    }
}
```

[In-class] Question 2

Write C# program to allow a **user to input two integer values** and then print the results of adding, subtracting, multiplying, dividing, modulus between the two values.

Expected Results



Answer: source code

```
using System;


class MathOperations
{
    public static void Main(string[] args)
    {
        Console.Write("Enter first number: ");
        int num1 = int.Parse(Console.ReadLine());


        Console.Write("Enter second number: ");
```

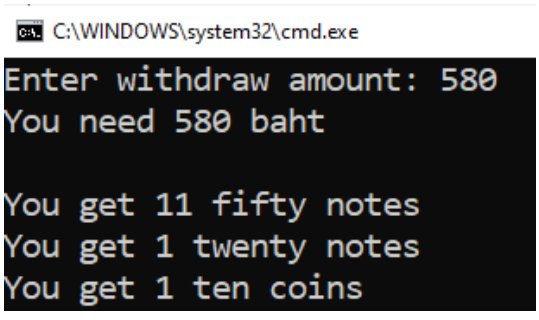
```
        int num2 = int.Parse(Console.ReadLine());


        int addition = num1 + num2;
        int subtraction = num1 - num2;
        int multiplication = num1 * num2;
        int modulus = num1 % num2;
        double division = (double)num1 / num2;


        Console.WriteLine("The result of adding is " + addition);
        Console.WriteLine("The result of subtracting is " + subtraction);
        Console.WriteLine("The result of multiplying is " + multiplication);
        Console.WriteLine("The result of modulus is " + modulus);
        Console.WriteLine("The result of dividing is " + division);
    }
}
```

[Homework] Question 1

Write a program to simulate a simple ATM.

- An ATM has **50, 20 banknotes** and **10 coins**
- Write a program to ask **users to input money** to withdraw (integer)
- Then tell the users how many banknotes they will get.
- Assume that the ATM will always give the largest banknotes first.

Expected Results



Answer: source code

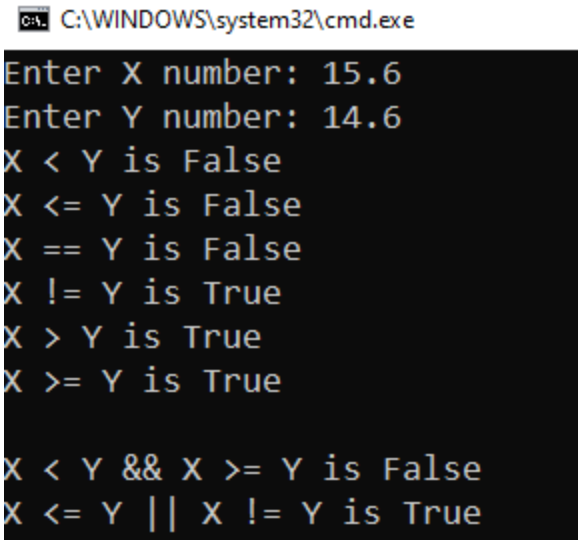
```
using System;
```

```
class ATM
{
   public static void Main(string[] args)
   {
      Console.Write("Enter withdraw amount: ");
      int amount = int.Parse(Console.ReadLine());

      Console.WriteLine("You need " + amount + " baht");

      int fiftyNotes = amount / 50;
      amount %= 50;

      int twentyNotes = amount / 20;
      amount %= 20;

      int tenCoins = amount / 10;

      Console.WriteLine("You get " + fiftyNotes + " fifty notes");
      Console.WriteLine("You get " + twentyNotes + " twenty notes");
      Console.WriteLine("You get " + tenCoins + " ten coins");
   }
}
```

[Homework] Question 2

Write C# program to allow the user to input **two floating point values** (double type) and then print the results of the Boolean expression between the two values.

Expected Results



Answer: source code

```
using System;

class BooleanExpressions
{
   public static void Main(string[] args)
   {
      // Input floating point values
      Console.Write("Enter X number: ");
      double x = double.Parse(Console.ReadLine());

      Console.Write("Enter Y number: ");
      double y = double.Parse(Console.ReadLine());

      // Output results of various boolean expressions
      Console.WriteLine("X < Y is " + (x < y));
      Console.WriteLine("X <= Y is " + (x <= y));
      Console.WriteLine("X == Y is " + (x == y));
      Console.WriteLine("X != Y is " + (x != y));
      Console.WriteLine("X > Y is " + (x > y));
      Console.WriteLine("X >= Y is " + (x >= y));

      // Combined expressions
      Console.WriteLine("X < Y && X >= Y is " + ((x < y) && (x >= y)));
      Console.WriteLine("X <= Y || X != Y is " + ((x <= y) || (x != y)));
   }
}
```

[Optional Question – Challenge] - This part is not required for completion and submission.

So you want to be a powerful programmer? Feel free to solve it and learn.

Question #1: You are about to send the NUCLEAR LAUNCH CODE to the command center via a public network. Anyone on the network can intercept your communication (no network layer encoding). Now you want to make a program to encode your CODE to CIPHERTEXT.

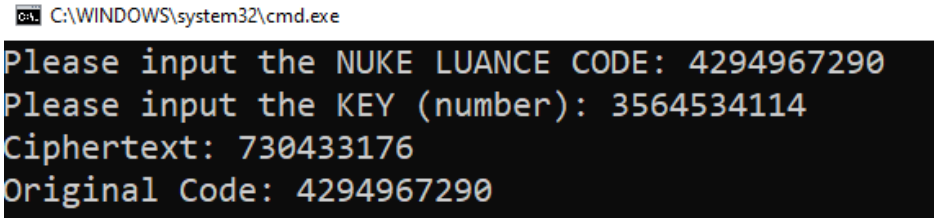
You can assume that the command center has a program to decode it.

TEST CASE:
CODE = 4294967290
KEY = 3564534114

HINT#1:
CODE xor KEY = CIPHERTEXT
CIPHERTEXT xor KEY = CODE

xor is amazing right?
Welcome to encoding 101

Expected Result



Got System.OverflowException eh?

HINT #2:
int or int 32 bits, it can hold the value up to 2,147,483,647
Then you ask HOW?
(binary 0 or 1) 2^31(this will be 32 bits for unsigned one - it need 1 bit to tell if the number is negative or positive - only 31 bits left) - 1 (for zero) = 2,147,483,647

You can try uint or unsigned int, which will give your power to hold up to 4294967295
(binary 0 or 1) 2^32 (this will be 32 bits for unsigned one) - 1 (for zero) = 4,294,967,295
OR
You might use int 64 bits for this, called long - use a bit more RAM.

Answer: source code

```
using System;

class Program
{
   public static void Main(string[] args)
   {

      Console.Write("Please input the NUKE LAUNCH CODE: ");
      ulong code = ulong.Parse(Console.ReadLine());

      Console.Write("Please input the KEY (number): ");
      ulong key = ulong.Parse(Console.ReadLine());

      ulong ciphertext = code ^ key;


      Console.WriteLine("Ciphertext: " + ciphertext);
      Console.WriteLine("Original Code: " + (ciphertext ^ key));
   }
}
```