• **INSTRUCTIONS:** This is the forth homework for CS170 (Section 003) and covers the following material: if-else statement, switch statement, while statement. Read the instructions carefully. In this homework you will need to write several Java programs. It is not on paper homework. You will need to write some Java programs, compile and test them and then you will need to submit them the same way you submitted lab assignments.

**Program style:** Style of your programs will give you 10% of the score. If your program is poorly written it won’t get perfect score even if it works correctly. By good style we mean:

– Code is well commented, you don’t need to comment all statements, just some major parts of the program.
– Variable naming reflects the meaning of the variable. For example, if a variable stores html code, the name should reflect this.
– Indentation: statement inside blocks should have indentation (usually 4 spaces).

• **HONOR CODE:** Like all work for this class, the Emory Honor Code applies. The assignment is an individual assignment and should be completed alone. For all programming assignments, you must write comment at the top of each file which include the following information:

```java
/*
 THIS CODE IS MY OWN WORK. IT WAS WRITTEN WITHOUT CONSULTING CODE
 WRITTEN BY OTHER STUDENTS OR MATERIALS OTHER THAN THIS SEMESTER’S
 COURSE MATERIALS. _Your_Name_Here_
*/
```

If the honor code isn’t present, the assignment will not be graded!

**Homework submission**
In your account on lab machines (you can connect and work on them remotely, please check class website for the instructions) put all program files in the hw4 subdirectory of cs170 directory (/home/<your username>/cs170/hw4/). You can create such directory using the following commands:

1. mkdir ~/cs170/hw4
2. save your solution program files to this directory: /home/<your username>/cs170/hw4
3. To submit programs: cd ~/cs170/hw4
4. /home/cs170003/turnin-hw EvalExpression.java hw4a
5. /home/cs170003/turnin-hw NumberSystem.java hw4b
6. You can submit multiple times, only the last submission will be stored.
Problem 1

1. (50 points) Evaluate expression
   Name your program EvalExpression Topics: conditional statements

   In this exercise you will write a program that evaluates simple arithmetic expressions. You need to write a Java program that asks user to input a simple expression in the form `<number><op><number>`, where `<>` is some number from 0 to 1000000 and `<op>` is one of the following operations: +, -, /, %. Examples of the correct expressions are: 2+5, 1000-1000, 6/2, etc. Note, that there is no space between numbers and operation, so you will need to read this expression as a string using next (or nextLine) method of Scanner. Your program needs to print the result of the expression entered.

   Example:

> java EvalExpression
Please enter expression: 21/7
21/7 = 3
> java EvalExpression
Please enter expression: 190382%2
190382%2 = 0

Extra[+??? pts]: make your program better in any way you may think of and you may get bonus credit. For example, check for division by zero or make your program work with more complicated expressions or data types, etc. Get something from your fantasy and implement it in Java.
Problem 2

2. (50 points) Number Systems
   Name your program NumberSystem Topics: strings, loops

   You learned how to convert numbers from one number system to another. But this task could be boring, why not teach the computer to do this for you. Write a program that reads some number and the corresponding number system and converts this number into decimal.

   > java NumberSystem
   Please enter number and its number system: 111 2
   The decimal value is: 7
   > java NumberSystem
   Please enter number and its number system: 123 10
   The decimal value is: 123
   > java NumberSystem
   Please enter number and its number system: A 16
   The decimal value is: 10

   Your program should be able to work with any number system from base 2 to base 16. Note, that the number that user may type can be hexadecimal and contain letters a-f, so you probably need to use string to store this number.

   **Hint**: Before you start writing Java code, write down the algorithm in natural language. Think about how would you solve the problem. Try to express this process with simple operations that you will be able to write in Java.

   **Extra [+50 pts]**: modify your program so that it could translate a number from any number system (from binary to hexadecimal) into any other number system. You will need to implement 2 algorithms: one is the same as for the main part of the problem, the other is to convert a number from decimal to any other system.