JAVA PROGRAMMING LANGUAGE
Assignment 2

1. The value e^x can be approximated by the following sum:

   $1 + x + x^2/2! + x^3/3! + \ldots + x^n/n!$

   Write a program that takes a value x as input and outputs this sum for n taken to be each of the values 1 to 10, 50, and 100. Your program should repeat the calculation for new values of x until the user says she or he is through.

   The expression n! is called the factorial of n and is defined as n! = 1 * 2 * 3 * … * n

   Use variables of type double to store the factorials (or arrange your calculation to avoid any direct calculation of factorials); otherwise, you are likely to produce integer overflow, that is, integers larger than Java allows.

2. In cryptarithmetic puzzles, mathematical equations are written using letters. Each letter can be a digit from 0 to 9, but no two letters can be the same. Here is a sample problem:

   SEND + MORE = MONEY

   A solution to the puzzle is S = 9, R = 8, O = 0, M = 1, Y = 2, E = 5, N = 6, D = 7.

   Write a program that finds a solution to the cryptarithmetic puzzle of the following:

   TOO + TOO + TOO + TOO = GOOD

   The simplest technique is to use a nested loop for each unique letter (in this case T, O, G, D). The loops would systematically assign the digits from 0 to 9 to each letter. For example, it might first try T = 0, O = 0, G = 0, D = 0, then T = 0, O = 0, G = 0, D = 1, then T = 0, O = 0, G = 0, D = 2, etc., up to T = 9, O = 9, G = 9, D = 9. In the loop body, test that each variable is unique and that the equation is satisfied. Output the values for the letters that satisfy the equation.