2. In the game of craps, a pass line bet proceeds as follows. Two six-sided dice are rolled; the first roll of the dice in a craps round is called the "come out roll." A come out roll of 7 or 11 automatically wins, and a come out roll of 2, 3, or 12 automatically loses. If 4, 5, 6, 8, 9, or 10 is rolled on the come out roll, that number becomes "the point." The player keeps rolling the dice until either 7 or the point is rolled. If the point is rolled first, then the player wins the bet. If a 7 is rolled first, then the player loses.

   Write a program that simulates a game of craps using these rules without human input. Instead of asking for a wager, the program should calculate whether the player would win or lose. The program should simulate rolling the two dice and calculate the sum. Add a loop so that the program plays 10,000 games. Add counters that count how many times the player wins, and how many times the player loses. At the end of the 10,000 games, compute the probability of winning [i.e., `Wins / (Wins + Losses)`] and output this value. Over the long run, who is going to win the most games, you or the house?

   Note: To generate a random number `x`, where `0 < x <= 1`, use `x = Math.random( );`. For example, multiplying `Math.random( )` by 6 and converting to an integer results in a random integer that is between 0 and 5.

3. One way to estimate the adult height of a child is to use the following formula, which uses the height of the parents:

   $$H_{male\_child} = ((H_{mother} \times 13/12) + H_{father})/2$$
   $$H_{female\_child} = ((H_{father} \times 12/13) + H_{mother})/2$$

   All heights are in inches. Write a program that takes as input the gender of the child, the height of the mother in inches, and the height of the father in inches, and outputs the estimated adult height of the child in inches. The program should allow the user to enter a new set of values and output the predicted height until the user decides to exit. The user should be able to input the heights in feet and inches, and the program should output the estimated height of the child in feet and inches. Use the `int` data type to store the heights.

4. It is difficult to make a budget that spans several years, because prices are not stable. If your company needs 200 pencils per year, you cannot simply use this year's price as the cost of pencils two years from now. Because of inflation, the cost is likely to be higher than it is today. Write a program to gauge the expected cost of an item in a specified number of years. The program asks for the cost of the item, the number of years from now that the item will be purchased, and the rate of inflation. The program then outputs the estimated cost of the item after the specified period. Have the user enter the inflation rate as a percentage, such as 5.6 (percent). Your program should then convert the percent to a fraction, such as 0.056, and should use a loop to estimate the price adjusted for inflation.

2. In the game of craps, a pass line bet proceeds as follows. Two six-sided dice are rolled; the first roll of the dice in a craps round is called the "come out roll." A come out roll of 7 or 11 automatically wins, and a come out roll of 2, 3, or 12 automatically loses. If 4, 5, 6, 8, 9, or 10 is rolled on the come out roll, that number becomes "the point." The player keeps rolling the dice until either 7 or the point is rolled. If the point is rolled first, then the player wins the bet. If a 7 is rolled first, then the player loses.

   Write a program that simulates a game of craps using these rules without human input. Instead of asking for a wager, the program should calculate whether the player would win or lose. The program should simulate rolling the two dice and calculate the sum. Add a loop so that the program plays 10,000 games. Add counters that count how many times the player wins, and how many times the player loses. At the end of the 10,000 games, compute the probability of winning [i.e., `Wins / (Wins + Losses)`] and output this value. Over the long run, who is going to win the most games, you or the house?

   Note: To generate a random number x, where `0 < x <= 1`, use `x = Math.random( );`. For example, multiplying `Math.random( )` by 6 and converting to an integer results in a random integer that is between 0 and 5.

3. One way to estimate the adult height of a child is to use the following formula, which uses the height of the parents:

   $H_{male\_child} = ((H_{mother} \times 13/12) + H_{father})/2$
   $H_{female\_child} = ((H_{father} \times 12/13) + H_{mother})/2$

   All heights are in inches. Write a program that takes as input the gender of the child, the height of the mother in inches, and the height of the father in inches, and outputs the estimated adult height of the child in inches. The program should allow the user to enter a new set of values and output the predicted height until the user decides to exit. The user should be able to input the heights in feet and inches, and the program should output the estimated height of the child in feet and inches. Use the `int` data type to store the heights.

4. It is difficult to make a budget that spans several years, because prices are not stable. If your company needs 200 pencils per year, you cannot simply use this year's price as the cost of pencils two years from now. Because of inflation, the cost is likely to be higher than it is today. Write a program to gauge the expected cost of an item in a specified number of years. The program asks for the cost of the item, the number of years from now that the item will be purchased, and the rate of inflation. The program then outputs the estimated cost of the item after the specified period. Have the user enter the inflation rate as a percentage, such as 5.6 (percent). Your program should then convert the percent to a fraction, such as 0.056, and should use a loop to estimate the price adjusted for inflation.

5. You have just purchased a stereo system that cost $1,000 on the following credit plan: no down payment, an interest rate of 18% per year (and hence 1.5% per month), and monthly payments of $50. The monthly payment of $50 is used to pay the interest, and whatever is left is used to pay part of the remaining debt. Hence, the first month you pay 1.5% of $1,000 in interest. That is $15 in interest. So, the remaining $35 is deducted from your debt, which leaves you with a debt of $965.00. The next month, you pay interest of 1.5% of $965.00, which is $14.48. Hence, you can deduct $35.52 (which is $50 – $14.48) from the amount you owe. Write a program that tells you how many months it will take you to pay off the loan, as well as the total amount of interest paid over the life of the loan. Use a loop to calculate the amount of interest and the size of the debt after each month. (Your final program need not output the monthly amount of interest paid and remaining debt, but you may want to write a preliminary version of the program that does output these values.) Use a variable to count the number of loop iterations and hence the number of months until the debt is zero. You may want to use other variables as well. The last payment may be less than $50 if the debt is small, but do not forget the interest. If you owe $50, your monthly payment of $50 will not pay off your debt, although it will come close. One month's interest on $50 is only 75 cents.

6. The Fibonacci numbers F_n are defined as follows: F_0 is 1, F_1 is 1, and

   $F_{i+2} = F_i + F_{i+1}$

   $i = 0, 1, 2, \ldots$. In other words, each number is the sum of the previous two numbers. The first few Fibonacci numbers are 1, 1, 2, 3, 5, and 8. One place where these numbers occur is as certain population growth rates. If a population has no deaths, then the series shows the size of the population after each time period. It takes an organism two time periods to mature to reproducing age, and then the organism reproduces once every time period. The formula applies most straightforwardly to asexual reproduction at a rate of one offspring per time period. In any event, the green crud population grows at this rate and has a time period of 5 days. Hence, if a green crud population starts out as 10 pounds of crud, then in 5 days, there is still 10 pounds of crud; in 10 days, there is 20 pounds of crud; in 15 days, 30 pounds; in 20 days, 50 pounds; and so forth. Write a program that takes both the initial size of a green crud population (in pounds) and a number of days as input and outputs the number of pounds of green crud after that many days. Assume that the population size is the same for 4 days and then increases every fifth day. Your program should allow the user to repeat this calculation as often as desired.

7. The value e^x can be approximated by the 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

$$n! = 1 * 2 * 3 * \ldots * n$$

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.

8. 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:

```
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.