Exercise Set 4: Conditional Loops

Topics:
Patterns: Repetition - Conditional
Objects: type boolean
Java language: relational operators < > <= >= != ==
System: infinite loop
Exercises: repetition controlled by a condition - unknown number of iterations

Overview:

In these exercises you practice writing loops where the number of iterations is not known at the start. Sometimes it is possible to actually compute the number of repetitions or use some other method to rewrite the loop as a counted loop, but for the purposes of clarity, the 'while' loop control is a preferred method.

Exercise strategies for students:

Read carefully the solved problems. Once you have read a couple of them, try to solve the additional once without looking at the solution and then compare your solution with the given one. They may not be the same. Observe the differences, make sure you understand them, then decide which one is easier for the user to understand. It may not be the posted solution!
Exercise Set 4: Conditional Loops

Exercises:

1. Write a program that will print the values of \( n \) and \( n! \) as long as \( n! < 10000 \).

2. Set up a driver that will ask the user to type in monthly payments to the savings plan the interest rate (yearly) and the desired savings goal. User types in the monthly payments, and the yearly interest rate. Call the function `SavingsPlan` and print the number of months needed to reach the goal. Allow the user to repeat the exercise with new set of input data - until no data is given. Write a function `SavingsPlan` that will compute how many months will it take to reach savings goal.

3. Write a program that will simulate the toss of a coin and count the number of tosses before two heads are tossed consecutively. Print all tosses to make sure the program runs correctly.

4. Write a program that will simulate the roll of a die and count the number of rolls before a five is rolled. Print all the rolls to see that your program runs correctly.

5. Write a program that will simulate the roll a pair of dice till you roll 'snake eyes' i.e. both rolls will be one. Print all pairs of rolls to make sure your program runs correctly.

6. Write a function `CirclesInaRow` that will paint as many circles in a row, with center at the vertical \( v \) and of the given radius \( r \) as will fit into the graphics window. Print how many circles were painted.

   Write a program that will ask the user for a radius of a circle and a vertical coordinate. The program will then call a function `CirclesInaRow` that will paint as many circles as will fit into the graphics window and print how many circles were painted. Allow the user to repeat the requests.

7. Repeat Exercise 6, but now start painting circles from the right.

8. Repeat Exercise 6, but make instead a column of circles, starting from the top.

9. Repeat Exercise 8, but this time start from the bottom.

10. Write a program that will paint a staircase going from the bottom left corner of the graphics window up to the right. User supplies the width of the steps and the height of the steps. Count the number of steps.