12 Number piramids

Write a program that output a number piramid like the one below.

```
1
22
333
4444
55555
666666
7777777
\ldots
```

13 All the primes up to 1,000 (*)

Write a program that prints all on screen all prime numbers up to 1,000.

14 Up to 1,000 primes (*)

Modify the program that you wrote for the former exercise so that it writes on screen the first 1,000 primes.

15 Guess my number (*)

Write a program that thinks of a random number between 0 and 1000, and then lets the user try to guess it. For every guess, the computer says whether the guess is correct, or too low, or too high. When the user finds the number, the computer will tell how many guesses were needed. The output could be similar to the following example:

```
Try to guess my number!
Tell me a number: 2
No! My number is higher.
Tell me a number: 800
No! My number is lower.
Tell me a number: 500
No! My number is lower.
Tell me a number: 350
No! My number is higher.
Tell me a number: 376
CORRECT!
You needed 5 guesses.
```

(Hint: to get a random number between 0 and 1000, use the following line:)

```
int numberToGuess = Math.abs(1000 * Math.random())
```

16 Rock, Paper, Scissors (*)

Write a program that reads 2 characters from either the keyboard or a file. The characters are either PP, PR, PS, RP, RR, RS, SP, SR, or SS. They correspond to the selections made by 2 players playing the game of rock-paper-scissors.

Make the program accept inputs until one player's score is more than 3 points ahead of the other.

Hint: remember that you can use `.substring()` to get the elements of a String.

17 Optimus Prime (**)

Write a program that reads an integer number from the user, and then outputs the closest prime number. If there are two prime numbers at the same distance, it should output both. For instance, if the user enters 5116, the output should be 5113 *and* 5119.

18 π (**)

Pi, the ratio of a circle's circumference to its diameter. can be computed by adding the following terms:

$$\pi = 4\sum_{k=0}^{\infty} \frac{(-1)^k}{2k+1} = \frac{4}{1} - \frac{4}{3} + \frac{4}{5} - \frac{4}{7} + \frac{4}{9} - \frac{4}{11} + \cdots$$

Create a program that asks the user for a number n and then calculates n of this infinite sum. How many terms do you need to get the first three digits right (3.14)? How many for the first 10 digits (3.14159265358.....)?