Homeworks - Part 1

Easy exercises (1 point each)

1. Calculate the remainder after dividing 31 079 into 170 166 719.
2. Using seq() and rep() as needed, create the vectors:

```
# 0 0 0 0 0 1 1 1 1 1 2 2 2 2 2 3 3 3 3 3 4 4 4 4 4

#1 2 3 4 5 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5
```

3. Using seq() and rep() as needed, create the vector:

```
#1 2 3 4 5 2 3 4 5 6 3 4 5 6 7 4 5 6 7 8 5 6 7 8 9
```

4. Write a function which, for a given natural number n, returns a sequence where each i $<$ n is repeated i times, in ascending order. E.g., for n = 4 the function should return c(1, 2, 2, 3, 3, 3, 4, 4, 4, 4).
5. Write a function which outputs whether a given number is positive or negative
6. Write a function which outputs whether a given number exceeds Pi in absolute value
7. Write a function which returns the geometric (or the harmonic) mean of two given numbers
8. Write a function which outputs which of two given character strings is shorter. (Hint: ? "nchar".)
9. Write a function which outputs a given character string once for each character in the string. (Hint: ? "for".)
10. Write a function which computes the total number of digits (in the decimal representation) in a given sequence of natural numbers.
11. Write a function which takes three numbers as arguments and returns the sum of the squares of the two larger numbers.

More complex exercises (2 points each)

0. Write a function that simulates 3-by-3 "random tictactoe". (Extend code from the lesson). It should stop when someone makes three in a row, column or diagonal. The output should be the end state of the board, the name of the winner, and the number of steps it took the game to end. (no points as we'll discuss this)
1. Write a function that rolls a die n times and gives back the index of the greatest number (all of them if there is a tie).
2. Write a function that rolls 2 dice and stops when both are the same. It should print out all the rolls.
3. Write a function that rolls 2 dice 10, 100 and 1000 times, and count how many times the first is greater.

4. Write a function that returns the first n numbers of the Fibonacci sequence. (https://en.wikipedia.org/wiki/Fibonacci_number)

5. Write a function that returns the nth row of the Pascal's triangle. (https://en.wikipedia.org/wiki/Pascal%27s_triangle)