Assignment 1

Handout: **Tuesday, 20 September 2016**
Due: **11:30 am, Thursday, 29 September 2016**

Goals:

- Review how floating point numbers are encoded in binary;
- Understand better the range and precision of different numeric data types;
- Write your first Java class;
- Get used to the IntelliJ Idea IDE;

1. Floating Point Vs. Integer Numbers

Suppose we have a new data type called `miniFloat`, which is very similar to `float` but uses only eight bits. From left to right, the meaning of the bits is as the following:

- 1 bit for the sign: 1 for positive values and 0 for negative values;
- 4 bits in two's complement for the exponent;
- 3 bits for the mantissa;

Consider for example a `miniFloat` value with the bit sequence `00100110`. The value is positive (`0`), the exponent is 4_{10} (= 0100_2), and the significand is 1.75_{10} (=1.110_2). Therefore the whole value is $1.75 \times 2^4 = 28$.

What to do:

Task 1: complete method `miniFloatFromString`;
Method `miniFloatFromString` takes a `String` of eight `1`'s and `0`'s, and returns the value of the `miniFloat` (as a `float` value);

Task 2: complete method `printIntegralMiniFloats`;
Method `printIntegralMiniFloats` prints all `miniFloat` values that are integers as well as the total number of such `miniFloat` values, as shown below

```
00000000 == 1
…
Total number of integral miniFloat values: <the total number>
```

To achieve this, the method calls `getValidMiniFloatBitSequences` to get all the (2^8=256) different 8-bit sequences, then compute the corresponding `miniFloat` value for each bit sequence and print the value out if it is an integer.

What to hand in:

The whole Assignment 1.1 folder with the completed methods.

2. Complex Numbers

> A complex number is a number that can be expressed in the form `a + bi`, where `a` and `b` are real numbers and `i` is the imaginary unit, that satisfies the equation $i^2 = -1$. In this expression, `a` is the real part and `b` is the imaginary part of the complex number.
>
> -- Wikipedia

Write a Java class for complex numbers. The class should have

1. two fields of type `double`, one for the real part and the other for the imaginary part;
2. a constructor with two parameters, for the real and the imaginary part respectively;
3. two methods called `add` and `multiply` respectively; Both methods take a parameter, do the calculation using `this` complex and the parameter, and return the result complex.
4. an `asString` method which returns the string representation of `this` in the form `(real, imaginary)`.

What to do:

Task 3: add the missing fields to class `Complex`;

Task 4: complete the constructor, method `add`, method `multiply`, and method `asString`;

Task 5: create `Complex` objects in the `main` method, add or multiply them, and print the results out to see if they are correct.

What to hand in:

The whole Assignment 1.2 folder with the completed methods and class.