Assignment 04 — 10/10/2018 – v1.0a
Smalltalk: Reflection

Please submit this exercise by mail to sma@list.inf.unibe.ch before 17 October 2018, 10:15am.

Exercise 1 - Hierarchy traversal (2 Points)
Write a method that finds the class with the longest inheritance chain among all Smalltalk classes in the Pharo programming environment.

Hint: To access all classes of Smalltalk, you can use Smalltalk allClasses.

Exercise 2 - Method overrides (2 Points)
Write a method to find all methods that override an abstract method in the Pharo system.

Exercise 3 - Query methods (2 Points)
Write a method that finds all classes in the Pharo environment with at least one query method.

Hint: Query methods test a property of an object. In Pharo these methods are prefixed with is, was or will.

Exercise 4 - Root methods (2 Points)
Find all root methods in the off-the-shelf Pharo image.

Hint: A “root method” is a method whose selector has been implemented in a class, such that the super classes of that class does not understand it.

Exercise 5 - Dynamic coding (2 Points)
Based on the code used in assignment 02 (downloadable from here), do the following exercise:

Step 1: Redefine the method Call doesNotUnderstand: aMessage and within this method dynamically add two elements to the class Call, namely the instance variable numberOfArguments, and the method described below.

```
call>>#numberOfArguments
    numberOfArguments := args size.
    ↑ numberOfArguments
```

Step 2: Execute the code

```
(CallGraph fromFile: 'Calls.txt') calls collect: [ :each | each numberOfArguments]
```

and ensure it prints the number of arguments for every call in the call graph without raising a doesNotUnderstand error.