10. Extending CodeScope

For the following exercises use the same image as in Exercise 9, CodeScope 1.

Now we want to extend the CodeScope application. Here is the list of information that we would like to gather from a package and some hints how you can get this information.

Methods

- Number of arguments
- Number of temps

Classes

- Number of instances existing in the system
- Hierarchy nesting level.
- Total number of lines of code
- Number of instance variables
- Percentage of methods that have accesses to instance variables
- Percentage of uncommented methods
- Percentage of extending methods (sending super)
- Percentage of overriding methods. (methods that are overridden in a subclass)
- Average lines of code per methods
- Average number of arguments per methods
- Average number of temps of methods

Packages

- Number of methods in a package
- Total number of lines of code
- Average number of methods per class for this package
- Average lines of code per class
- Average lines of code per method
- Percentage of uncommented classes (classes with no class comment)
- Percentage of uncommented methods

Hints

Q: How do I access the comment of a class?

A: Have a look at class ClassDescription, category *accessing comment*

Q: How do I access the comment of a method?

A: Have a look at class Behavior, method *firstCommentAt:*

Q: How do I check for super sends or instance variable accesses?

A: Have a look at class CompiledMethod, category *scanning*

Q: How do I check for number of temps/arguments of a method?

A: Have a look at class CompiledMethod, category *accessing*

Q: How do get all instances of a class?

A: Have a look at class Behavior, method *allInstances*

Please save the Monticello package `CodeScope` and send it by mail to st-staff@iam.unibe.ch. Attach your written solutions that are not part of the source-code to the mail or hand them in as hardcopy at the beginning of the next exercise session. Your mail and solutions should be clearly marked with names and matrikel numbers of the solution authors.