Rheinische Friedrich-Wilhelms-Universität Bonn
Institut für Informatik III
Prof. Dr. A. B. Cremers

Advanced Logic Programming
Summer Semester 2011
Dr. Günter Kniesel

Assignment 8

Due: Sunday, 6.06.2011, 23:59:59 via SVN

Submit your solution for this assignment into your group's SVN as an **Eclipse Project** named "assignment08".
Programming exercises should be submitted as **Prolog files** in this project.
Theoretical exercises should be submitted as **PDF files** in this project.

For the following tasks, submit:

1. the implementation of each predicate,
2. the Java code on which you tested your predicates and
3. the excerpt of your Prolog Console session that documents your successful tests.

The Java code should contain examples for all relevant cases. You may use any program of your own or from open source repositories. Your solutions (especially for task 3) do not have to work on Java code that contains any directly or indirectly recursive calls.

You may use predicates implemented the previous tasks, where appropriate.

See http://sewiki.iai.uni-bonn.de/research/jtransformer/ for the documentation of JTransformer's program element facts that you need.

Task 1. *Direct change of a field* (3 Points)

Write a predicate *changing_method(?Field, ?Method)* that succeeds if *Field* is the identity of a field that is assigned within a method with identity *Method*.

Task 2. *Direct call of a method* (2 Points)

Write a predicate *direct_calling_method(?Callee, ?Caller)* that succeeds if *Callee* is the identity of a method that is called somewhere inside the body of the method with identity *Caller.*

Task 3. *Transitive call of a method* (5 Points)

Write a predicate *calling_method(?Method, ?Caller)* that succeeds if the method represented by *Method* may be invoked through the execution of the method represented by *Caller.* That is *Caller* contains a call that either invokes *Method* itself or invokes some other method that again contains some code through which *Method* can be called.

It suffices that a call is contained inside a body of a method. It is not relevant whether it is actually executed at runtime. You don't have to solve the halting problem.

Task 4. *Calling class* (3 Points)

Write a predicate *calling_class(?Method, ?Class)* that succeeds if *Class* is the identity of a class within that contains at least one method that calls the method with identity *Method*.

Task 5. *Field changing class*(3 Points)

Write a predicate *changing_class(?Field, ?Class)* that succeeds, if *Class* is the identity of a class that may change the state of the field represented by *Field*.