

Exam *Generic Language Thechnology* (2IS15) 12th of April 2012, 09:00-12:00.

---

This exam consists of 4 questions.

*You are allowed to use all distributed material, slides, books, papers, and laptop.*

*You need to give a concise motivation for all the answers.*

---

## 1. Basic technology

- (a) Give at least two systems that can be considered as products of generic language technology? Describe briefly why they can be considered as such.
- (b) Given a regular expression, a non-deterministic finite automaton can be constructed. Why is it necessary to convert the non-deterministic finite automaton into a deterministic one when implementing a scanner?
- (c) Rewrite the regular expression  $(a|b)c * |dc*$  into a simpler one and then translate the resulting regular expression into a NFA.
- (d) Give the priority/associativity definition in YACC to disambiguate the sentence "1 \* 3 + 2 - 4". Assume that all operators are left associative and that the multiplication operator has a higher priority than the addition and subtraction operator, which have the same priority.
- (e) What is the difference between a context-free grammar and a language?
- (f) Why is grammar modularity not supported in tools like Yacc, Bison, ANTLR? What is the theoretical bottleneck?
- (g) What does the set of item sets in the LR parse table construction represents a finite automaton? Describe the effect of a shift and reduce operation in terms of this finite automaton.

## 2. Grammars and static semantics

- (a) Why may it be necessary to use type information when resolving the use of identifier in a block structured language? Illustrate this by means of a (non-trivial) example.
- (b) Which steps should be taken when defining a domain specific language? Motivate your answer, if possible by means of a small example.
- (c) Is OCL suited to define the typing rules for a DSL? What other formalisms can be used to implement typing rules for a DSL?

- (d) What is the difference between overloading and polymorphism? Is one of two more powerful?

### 3. Dynamic semantics

- (a) Why is a formally defined dynamic semantics of a modeling language necessary for the formal analysis of models (or programs) written in the language? What can be the consequences otherwise?
- (b) In the paper of F. Stappers e.a. on the formalization of the TRECS language, Structural Operational Semantics is used. Why has the natural (big-step) semantics not been considered adequate for this language?
- (c) Assume that two languages  $L_1$  and  $L_2$  both have a formal dynamic semantics defined in the same style (for instance, both have Structural Operational Semantics defined). One wants to show that a construct  $ccc$  from  $L_1$  can be expressed by the constructs of  $L_2$ . What needs to be proven to show that this is indeed the case?
- (d) Consider the extension of the **While** language with the `abort` statement and with non-determinism `or`. Which of the following statements are pair-wise semantically equivalent:

$S_1 \equiv (x := 1; \text{while true do skip}) \text{ or } y := 2$

$S_2 \equiv x := 1; ((\text{while true do skip}) \text{ or } y := 2)$

$S_3 \equiv (x := 1; \text{abort}) \text{ or } y := 2$

$S_4 \equiv x := 1; (\text{abort or } y := 2)$

$S_5 \equiv (x := 1; \text{skip}) \text{ or } y := 2$

$S_6 \equiv x := 1; (\text{skip or } y := 2)$

$S_7 \equiv x := 1 \text{ or } y := 2$

$S_8 \equiv x := 1; y := 2$

in the natural semantics of the extended **While**, or the structural operational semantics of the extended **While**? Explain your reasoning.

### 4. DSL Design

- (a) What is the fundamental difference between an abstract syntax and a meta-model?
- (b) Why is the Java/Swing library an example of a domain specific language? What other forms of domain specific language do you know?
- (c) Why is it necessary to perform a domain analysis phase before designing a domain specific language?

- (d) Xtext and EMFtext are both based on ANTLR. What are the consequences for both tools? Describe two aspects where Xtext and EMFtext fundamentally differ?
- (e) What is the role of a code generator in the design of domain specific language? Is a code generator an endogenous or exogenous transformation?
- (f) ATL, Xtend, and QVT are all model transformation languages. If you had to do a model transformation project, which language would you choose. Give at least 3 well motivated arguments.

Grading of exercises

1	<i>a</i>   20 <i>b</i>   10 <i>c</i>   10 <i>d</i>   20 <i>e</i>   10 <i>f</i>   10 <i>g</i>   20	2	<i>a</i>   20 <i>b</i>   20 <i>c</i>   20 <i>d</i>   20	3	<i>a</i>   10 <i>b</i>   10 <i>c</i>   10 <i>d</i>   30	4	<i>a</i>   10 <i>b</i>   20 <i>c</i>   10 <i>d</i>   20 <i>e</i>   20 <i>f</i>   30
---	---	---	--	---	--	---	--