Faculteit der Exacte Wetenschappen	Tentamen Compilerbouw
Vrije Universiteit	21-12-2005

Questions can be answered in Dutch or English.

1. General knowledge: Explain the following terms:

- **a.** top-down parser
- **b.** attribute grammar
- **c.** bootstrapping
- **d.** register allocation

2. Lexical analysis:

- **a.** What is the dot motion rule for a lexical item of the form

$$[\, T \rightarrow \alpha \cdot (R)^* \, \beta \,]$$

- **b.** Explain this rule.

3. LR parsing: Construct the $LR(0)$ automaton for the grammar

$$S \rightarrow x\, x\, S \mid a$$

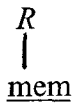
where x and a are terminal symbols.

4. Context handling:

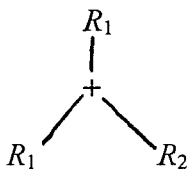
- **a.** Describe briefly the difference between simple and full symbolic interpretation.
- **b.** Give the dataflow equations for forward and backward dataflow analysis.

5. Code generation: Given a machine with 3 machine instructions:

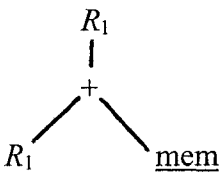
(1) $R := \underline{\text{mem}}$



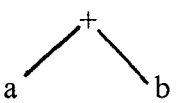
(2) $R_1 := R_1 + R_2$



(3) $R_1 := R_1 + \underline{\text{mem}}$



where mem denotes a memory location, and given the input tree



where a and b are memory locations. The instructions and the tree are presented to a bottom-up tree-rewriting code generator (BURS code generator).

a. Show the sets the BURS code generator builds at the nodes of the imput tree, and explain why is does so.

b. Show the tree or trees that result from the rewriting process.

6. Code post-processing, peephole optimization:

a. What is a replacement pattern?

b. How are the left-hand sides of replacement patterns found efficiently in the input stream?

7. Memory Management: How can one implement an extensible array so it can be extended in linear time?

8. Routines: What is a static link (lexical pointer) and what is it used for?

9. Logic programs: In the Prolog rule

grandparent (X, Z) :– parent (X, Y), parent (Y, Z).

the goal *parent (X, Y)* may match for more than one *Y*. How are these multiple values transferred to the second goal *parent (Y, Z)* ?

Assessment:

	1:	2:	3:	4:	5:	6:	7:	8:	9:
a:	4	5	10	7	7	2	7	6	8
b:	4	7		7	5	5			
c:	3								
d:	3								

14 + 12 + 10 + 14 + 12 + 7 + 7 + 6 + 8 = 90