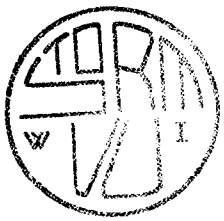
Faculteit der Exacte Wetenschappen	Tentamen Compilerbouw
Vrije Universiteit	23-01-2002

Questions can be answered in Dutch or English.



1. Explain the following terms:

a. grammar

b. symbol table

c. register allocation

d. retargetable compiler

2. Lexical analysis: Give two reasons why it can be profitable to have a lexical identification module between the lexical analyser and the parser.

3. Refer to the transition diagram for an LR(0) automaton in Figure 2.8, attached.

a. Some states, for example S_4, have no outgoing arrows. What happens in these states?

b. Some states have no outgoing arrows for some tokens, for example S_8 for the token '('. What happens in such states when such a token is found?

4. Attribute grammars: Consider the attribute grammar

$S \rightarrow A(i, s)$	$[\, i := s \,]$
$A(i_1, s_1) \rightarrow A(i_2, s_2)\ a$	$[\, i_2 := i_1 ;\ s_1 := s_2 \,]$
$A(i_1, s_1) \rightarrow B(i_2, s_2)$	$[\, i_2 := i_1 ;\ s_1 := s_2 \,]$
$B(i, s) \rightarrow b$	$[\, s := i \,]$

in which the $\underline{i}$s are inherited attributes and the $\underline{s}$s are synthesized attributes. The attribute evaluation rules are given between square brackets.

Show by constructing the IS/SI graph of A that the attribute grammar is circular (contains a cycle).

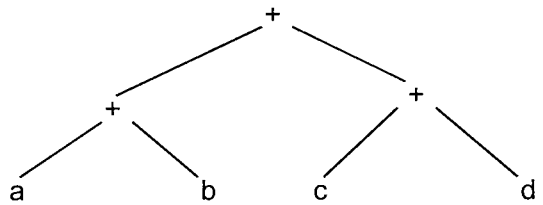
5. Context handling: Explain briefly how 'symbolic interpretation' works.

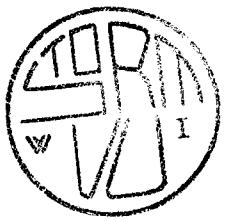
6. Code generation: Given a machine with only two instructions:

$R_n := R_n + R_m$

$R_n :=$ variable

How many registers are needed for the translation of the tree (without restructuring the tree)





in which a, b, c, d are variables? Show the calculations for obtaining the result.

7. Memory management: Explain briefly how garbage collection by 'reference counting' works.

8. Routines: Given the nested routines (in C-like notation)

```
void level_0(void) {
    void level_1(void) {
        void level_2(void) {
            ...
            goto L_1;
            ...
        }
        ...
L_1:...
        ...
    }
    ...

}
```

and given that the calling sequence "`level_0` calls `level_1` calls `level_2` calls `level_2`" has occurred and that the last `level_2` has executed a jump to `L_1`.

a. Draw and explain the chain of activation records before and after the jump.

b. Is the static link (lexical pointer) involved in the jump?

9. Logic programs: In the Prolog rule

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

the goal *parent (X, Y)* may match 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:	3	7	10	15	7	8	6	12	8	
b:	3							3		
c:	4									
d:	4									
	14	7	10	15	7	8	6	15	8	Total : 90

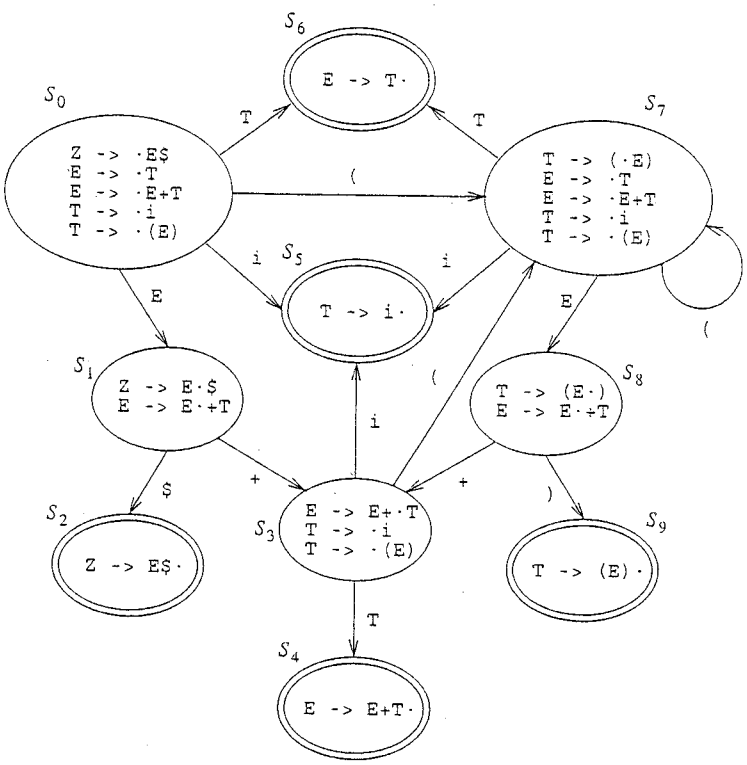


Figure 2.89 Transition diagram for the LR(0) automaton for the grammar of Figure 2.85.