Faculteit der Exacte Wetenschappen	Tentamen Compilerbouw
Vrije Universiteit	19-01-2000

Questions can be answered in Dutch or English.

1. Explain the following terms:

a. transition table

b. intermediate code

c. inference technique

2. When generating LL(1) parsers, two types of conflicts may occur.

a. Which are they?

b. How do they originate?

3. Refer to the transition diagram for an LR(0) automaton in Figure 2.85, 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. Suppose dataflow equations are used to track the initialization status of a variable *x*.

a. What information should be recorded for *x* between each node pair?

b. Give the KILL and GEN sets for a node containing *x* := expression.

5. a. Explain briefly how 'compilation on the stack' (also known as 'compilation by symbolic interpretation') works.

b. What information is recorded in the 'regvar' descriptor?

6. a. Explain briefly how 'peephole optimization' works.

b. Describe an efficient method to locate matching patterns in the target instruction list.

7. **a.** Explain briefly how garbage collection by 'reference counting' works.

   b. What serious problem does this method have?

8. In code generation for object-oriented languages, methods are basically translated as routines.

   a. How can the code for a method application find the proper routine to call, in the presence of dynamic binding?

   b. How can the called routine find the data of the object it should work on?

Assessment:

	1:	2:	3:	4:	5:	6:	7:	8:	
a:	3	4	5	5	7	7	6	5	
b:	3	8	5	5	8	8	3	5	
c:	3								
	9	12	10	10	15	15	9	10	Total : 90

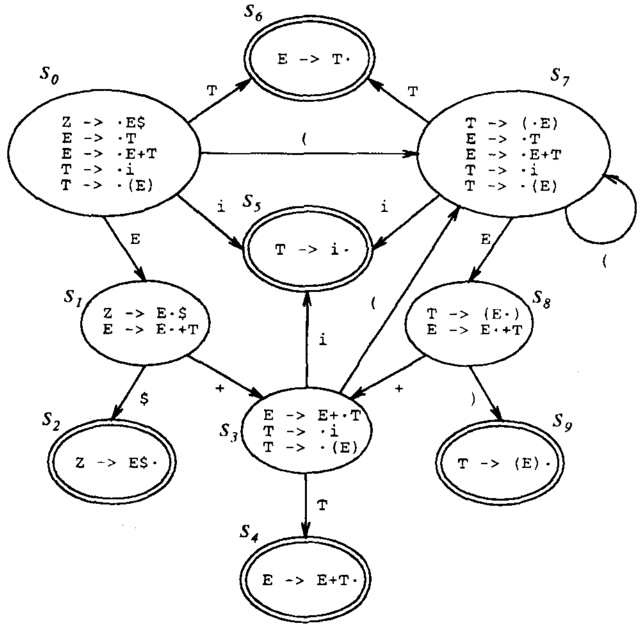

Figure 2.85 Transition diagram for the LR(0) automaton for the grammar of Figure 2.81.