7 LR-Grammars

7.1 Consider the following CFG G (S is the start symbol):

$$S \to A\$$$
$$A \to AB \mid B$$
$$B \to (A) \mid ()$$

a) Construct an NFA_ϵ which shows the valid LR(0) items for each viable prefix. (You may choose to skip this step and go directly to b).

b) Construct an equivalent DFA (exclude the error state and all transitions to it).

c) Is G an LR(0) grammar?

7.2 Is the following CFG an LR(0) grammar? (S' is the start symbol.)

$$S' \to S$$
$$S \to bA \mid aB$$
$$A \to a \mid aS \mid bAA$$
$$B \to b \mid bS \mid aBB$$

7.3 Show how the following strings are parsed by the LR(0) parser whose finite control is given by the DFA in exercise 7.1.b. For each step show the stack, the remaining input and whether the operation is "shift" or "reduce". For reduce operations show which productions are involved.

a) $(()())\$$

b) $()((())())\$$

7.4 Is the following CFG an LR(0) grammar? (S is the start symbol.)

$$S \to A$$
$$A \to aAa \mid bAb \mid c$$

7.5 Consider the following CFG G (S is the start symbol):

$$S \to A$$
$$A \to AB \mid \epsilon$$
$$B \to b \mid aB$$

a) Construct an NFA_ϵ showing the valid LR(1) items for each viable prefix. (You may choose to skip this step and construct a DFA instead).

b) Is G an LR(1) grammar?

7.6 Consider the following CFG G (S is the start symbol):

$$S \to E$$
$$E \to E + T \mid T$$
$$T \to a \mid (E)$$

a) Construct a DFA showing the valid LR(1) items for a viable prefix.

b) Show how the string $a+(a+a)\$$ is parsed by the LR(1) parser corresponding to the DFA in a) ($ is used to denote 'end-of-input').

   For each step show the stack, the remaining input, the kind of action ("shift" or "reduce") and the production used in "reduce".

7.1 a) The NFA$_\epsilon$ M_{35} is shown in figure 35.

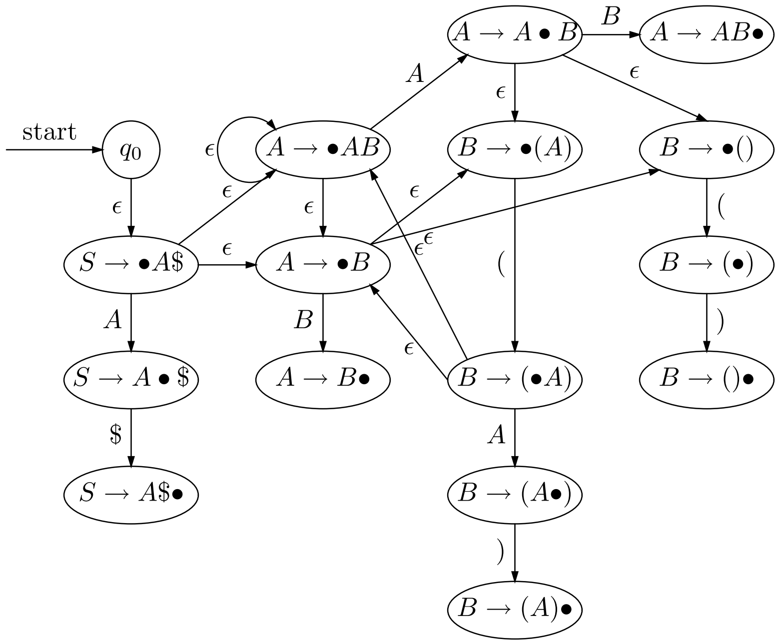

Figure 35: M_{35}

b) The DFA M_{36} is shown in figure 36.

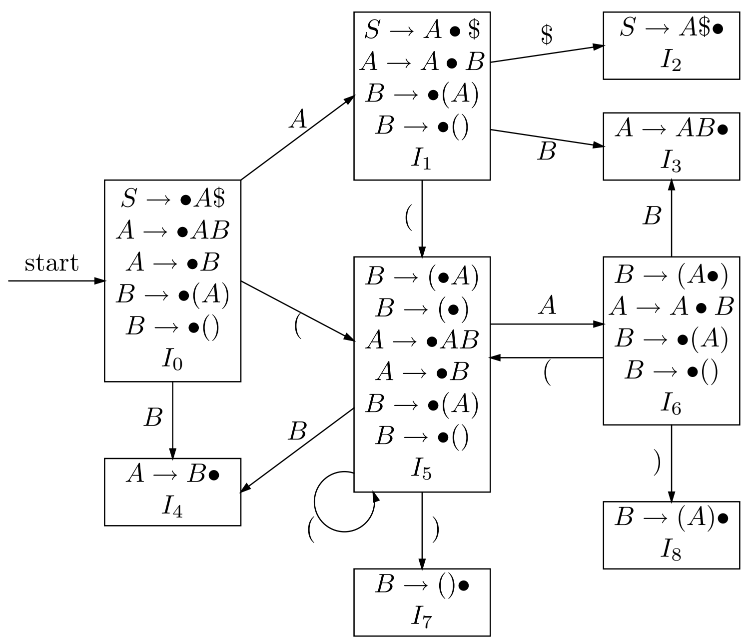

Figure 36: M_{36}

c) Yes, the grammar is LR(0).

7.2 If we construct a DFA which determines the set of valid LR(0) items for each viable prefix (see figure 37), then we will find that both $A \to a\bullet$ and $S \to \bullet bA$ are valid for the viable prefix ab, for instance. Since $A \to a\bullet$ is a complete item, and another item is valid for the same viable prefix, the grammar is *not* LR(0).

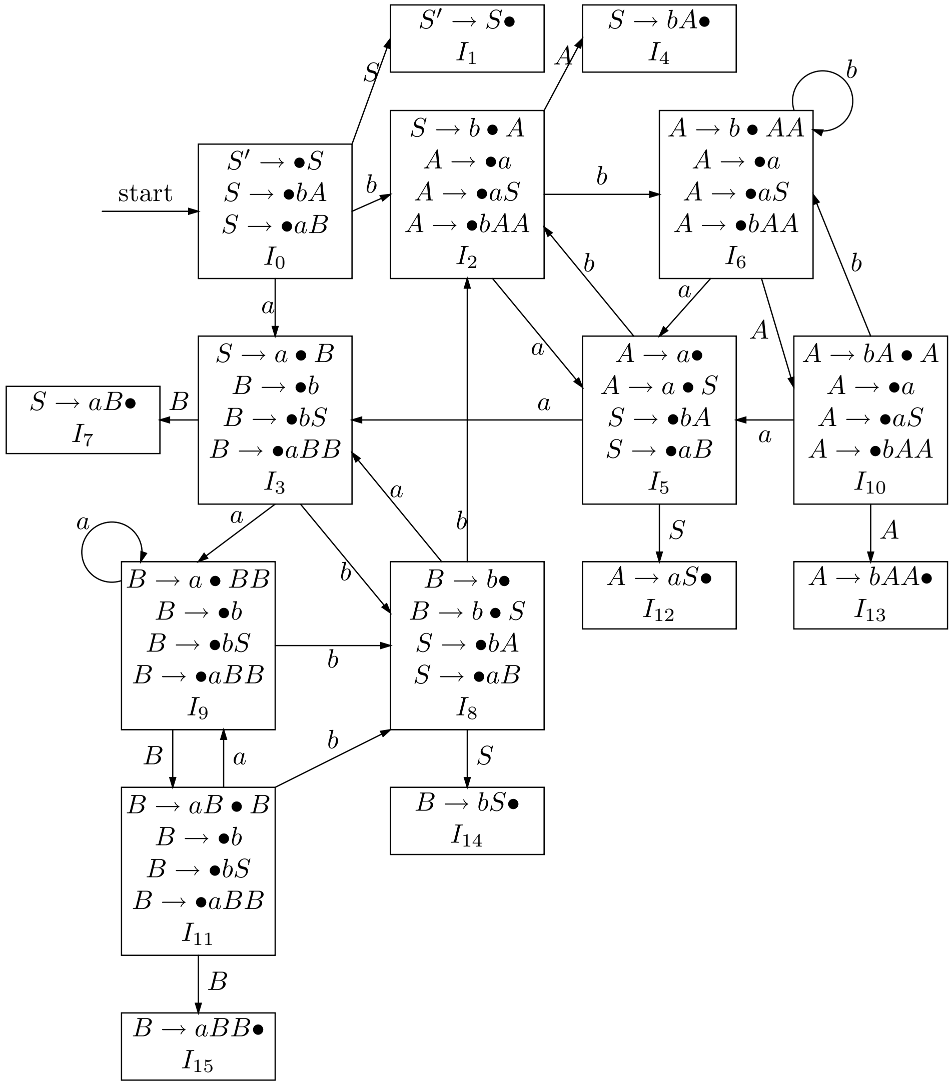

Figure 37: M_{37}

7.3 a)

Stack	**Remaining input**	**Comment**
0	(()())$	Start
0(5	()())$	Shift
0(5(5	)())$	Shift
0(5(5)7	())$	Shift
0(5B4	())$	Reduce by $B \rightarrow ()$
0(5A6	())$	Reduce by $A \rightarrow B$
0(5A6(5	))$	Shift
0(5A6(5)7	)$	Shift
0(5A6B3	)$	Reduce by $B \rightarrow ()$
0(5A6	)$	Reduce by $A \rightarrow AB$
0(5A6)8	$	Shift
0B4	$	Reduce by $B \rightarrow (A)$
0A1	$	Reduce by $A \rightarrow B$
0A1$2	-	Shift
0S	-	Reduce by $S \rightarrow A\$$ and accept

7.4 If a DFA is constructed as in exercise 7.1 (see M_{38} in figure 38), we will find that the grammar is LR(0).

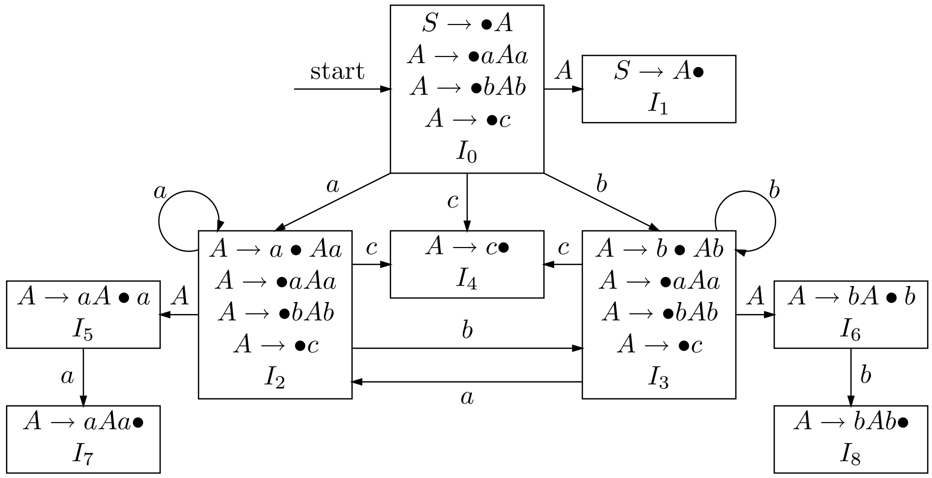

Figure 38: M_{38}

7.5 a) See M_{39} in figure 39 ($ denotes 'end-of-input').

b) If a DFA equivalent to the NFA$_\epsilon$in a) is constructed (see figure 40), we will find that the grammar is LR(1).

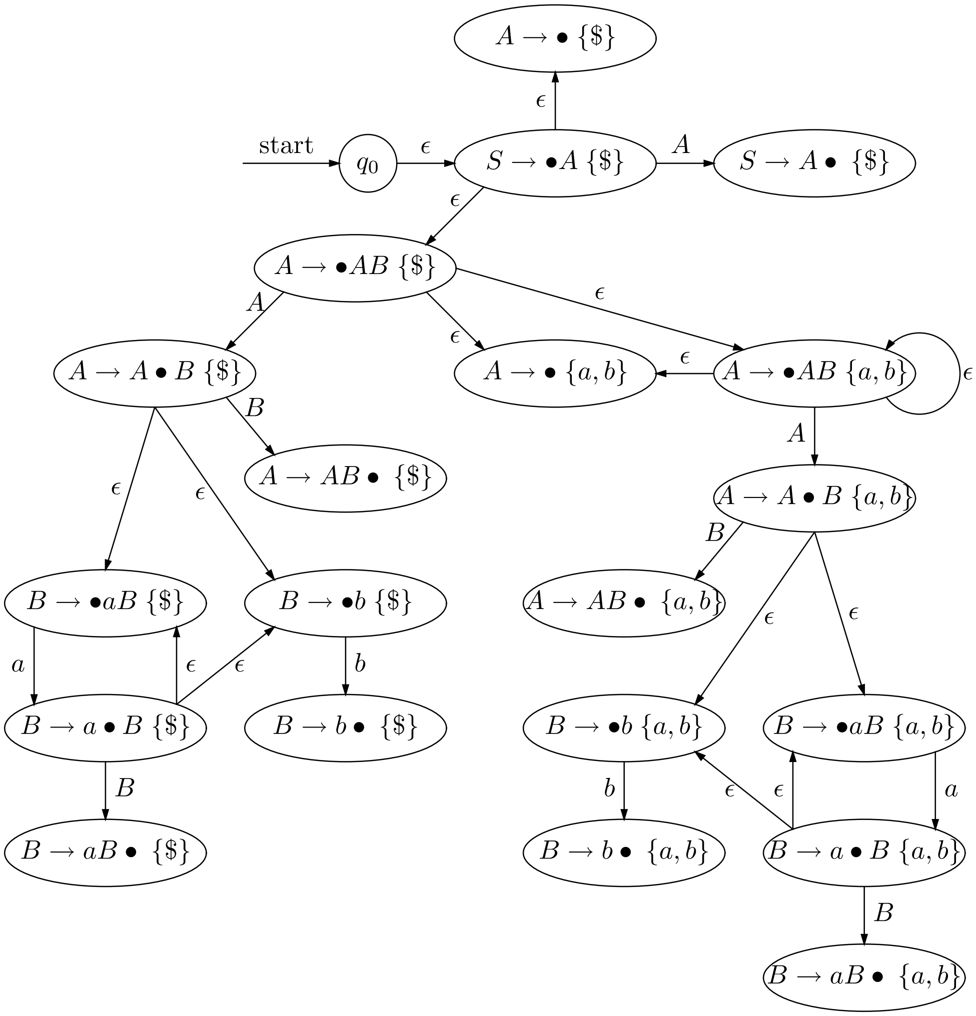

Figure 39: M_{39}

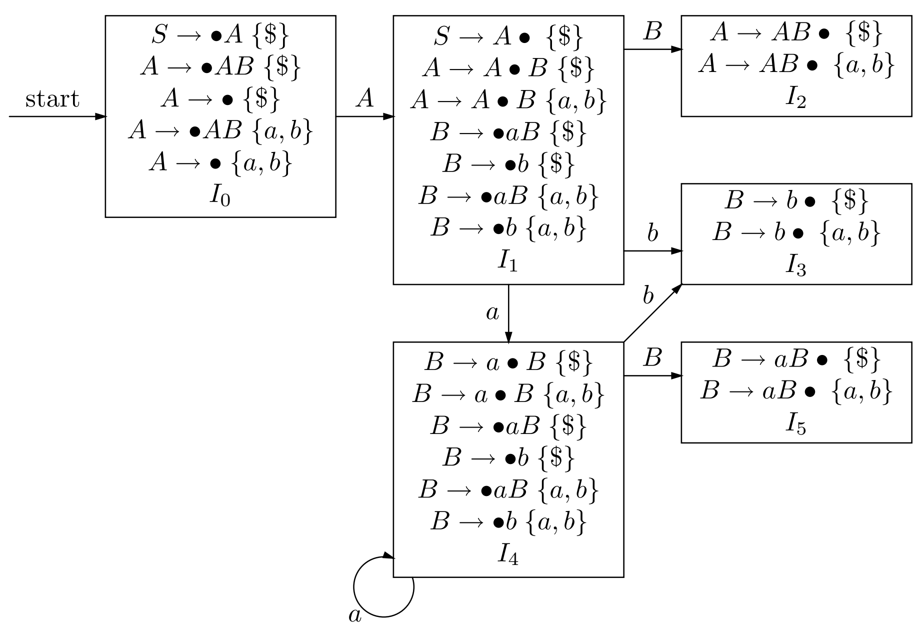

Figure 40: M_{40}

7.6 a) See figure 41.

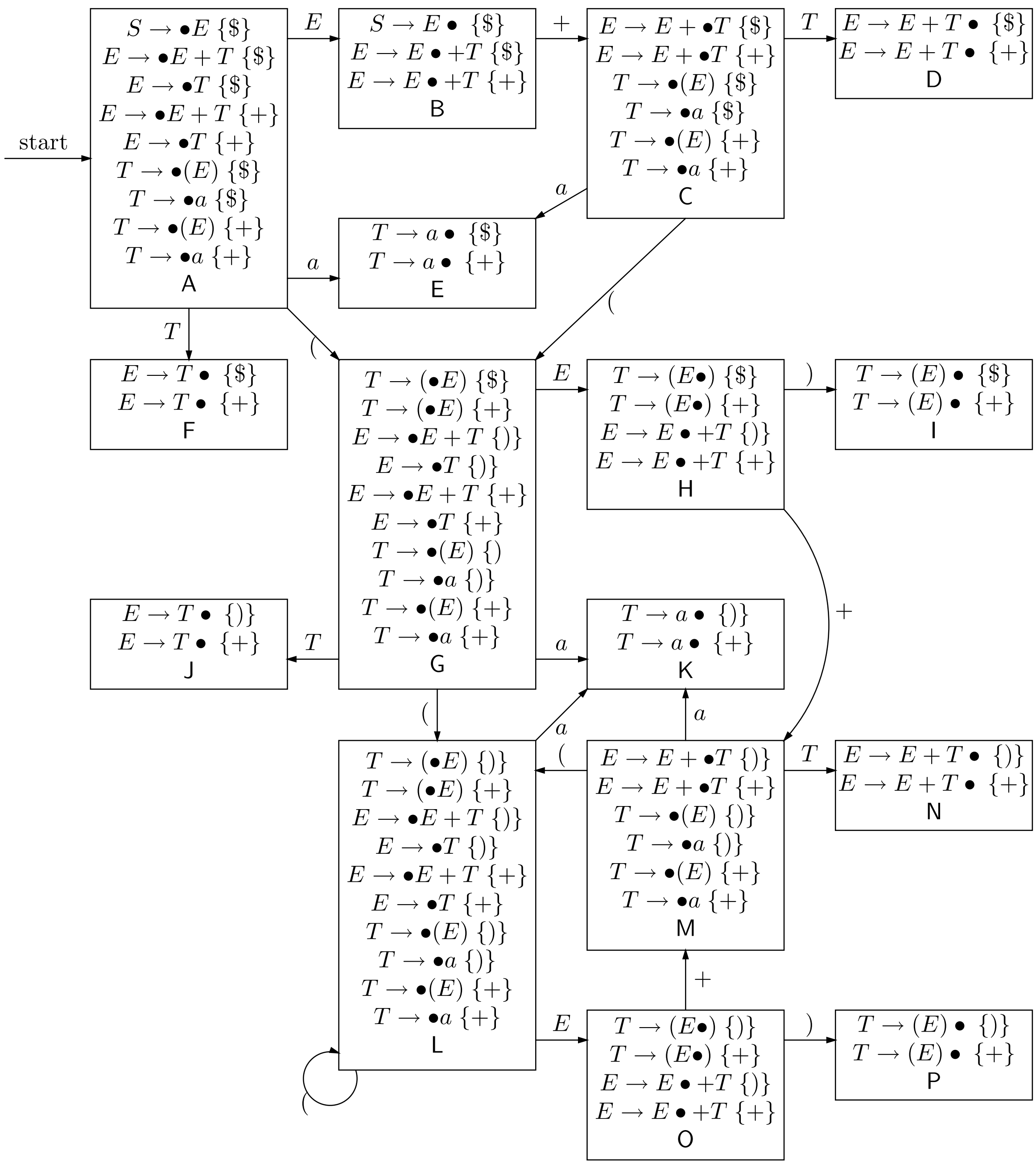

Figure 41: M_{41}

b)

Stack	**Remaining input**	**Comment**
A	$a + (a + a)\$$	Start
AaE	$+(a + a)\$$	Shift
ATF	$+(a + a)\$$	Reduce by $T \to a$
AEB	$+(a + a)\$$	Reduce by $E \to T$
AEB+C	$(a + a)\$$	Shift
AEB+C(G	$a + a)\$$	Shift
AEB+C(GaK	$+a)\$$	Shift

AEB+C(GTJ	$+a)\$$	Reduce by $T \rightarrow a$
AEB+C(GEH	$+a)\$$	Reduce by $E \rightarrow T$
AEB+C(GEH+M	$a)\$$	Shift
AEB+C(GEH+MaK	$)\$$	Shift
AEB+C(GEH+MTN	$)\$$	Reduce by $T \rightarrow a$
AEB+C(GEH	$)\$$	Reduce by $E \rightarrow E + T$
AEB+C(GEH)I	$\$$	Shift
AEB+CTD	$\$$	Reduce by $T \rightarrow (E)$
AEB	$\$$	Reduce by $E \rightarrow E + T$
AS	$\$$	Reduce by $S \rightarrow E$ and accept

An LR(1) parser can be described by a decision table. For an input symbol and a DFA state from the stack, the table gives the parser's action. The decision table corresponding to the DFA above is shown in table 8.

Stack top	Next input symbol				
	a	+	(	)	\$
A	Shift		Shift		
B		Shift			Reduce by $S \rightarrow E$ and accept
C	Shift		Shift		
D		Reduce by $E \rightarrow E + T$			Reduce by $E \rightarrow E + T$
E		Reduce by $T \rightarrow a$			Reduce by $T \rightarrow a$
F		Reduce by $E \rightarrow T$			Reduce by $E \rightarrow T$
G	Shift		Shift		
H		Shift		Shift	
I		Reduce by $T \rightarrow (E)$			Reduce by $T \rightarrow (E)$
J		Reduce by $E \rightarrow T$		Reduce by $E \rightarrow T$	
K		Reduce by $T \rightarrow a$		Reduce by $T \rightarrow a$	
L	Shift		Shift		
M	Shift		Shift		
N		Reduce by $E \rightarrow E + T$		Reduce by $E \rightarrow E + T$	
O		Shift		Shift	
P		Reduce by $T \rightarrow (E)$		Reduce by $T \rightarrow (E)$	

Table 8: Decision table