Name:

Student ID:

Compiler Construction, Spring 2020
Quiz 4

Given the context-free grammar (CFG), please answer the following questions:

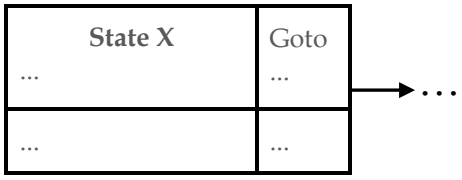
1) build the **complete** transition diagram (i.e., characteristic finite-state machine, CFSM) (3pt), and
2) write the **complete** parse table using the proper LR table construction method. Please state the table construction method you used for parse table construction. (2pt)

Context-free grammar:

```
1| S -> E $
2| E -> T
3|     | E + T
4| T -> id
5|     | ( E )
```

Note: (a) Reducible states are **double-boxed** nodes. (b) The transitions should be shown with **labeled edges** between the states.

***Example* of a transition diagram node:**



Note: (a) Table entry should use s or r to indicate it is a shift or reduce operation, where the s or r should be followed by a state or rule number, e.g., s1 or r2. (b) In addition, an Accept entry means a successful parsing, whereas an empty entry means a syntax error.

***Example* of the (simplified) parse table:**

State	*S*	*E*	*T*
0	**s1**		**Accept**
1		**r2**	
2			

Answer:

LR(0) is used and its parse table is as below.

State	*id*	+	(	)	$ (end)	*S*	*E*	*T*
0	s5		s7			Accept	s1	s6
1		s3			s2			
2	r1							
3	s5		s7					s4
4	r3							
5	r4							
6	r2							
7	s5		s7				s8	s6
8		s3		s9				
9	r5							

States (*gray* table entries are reducible states):

State	Items
0	S -> ·E$ E -> ·T E -> ·E + T T -> ·id T -> ·(E)
1	S -> E·$ E -> E·+ T
2	S -> E$·
3	E -> E+·T T -> ·id T -> ·(E)
4	E -> E + T·
5	T -> id·
6	E -> T·
7	T -> (·E) E -> ·T E -> ·E + T T -> ·id T -> ·(E)
8	T -> (E·) E -> E·+T
9	T -> (E) ·