Grammar:

$E \rightarrow E + E \mid E * E \mid (E) \mid \mathbf{id}$

Parsing table:

STATE	action						goto
	id	+	*	(	)	$	*E*
0	s3	e1	e1	s2	e2	e1	1
1	e3	s4	s5	e3	e2	acc	
2	s3	e1	e1	s2	e2	e1	6
3	r4	r4	r4	r4	r4	r4	
4	s3	e1	e1	s2	e2	e1	7
5	s3	e1	e1	s2	e2	e1	8
6	e3	s4	s5	e3	s9	e4	
7	r1	r1	s5	r1	r1	r1	
8	r2	r2	r2	r2	r2	r2	
9	r3	r3	r3	r3	r3	r3	

Fig. 4.53. LR parsing table with error routines.

```
set ip to point to the first symbol of w$;
repeat forever begin
    let s be the state on top of the stack and
        a the symbol pointed to by ip;
    if action[s, a] = shift s' then begin
        push a then s' on top of the stack;
        advance ip to the next input symbol
    end
    else if action[s, a] = reduce A → β then begin
        pop 2*|β| symbols off the stack;
        let s' be the state now on top of the stack;
        push A then goto[s', A] on top of the stack;
        output the production A → β
    end
    else if action[s, a] = accept then
        return
    else error()
end
```

E1:

- This routine is called from states 0, 2, 4 and 5, all of which expect the beginning of an operand, either an **id** or a left parenthesis.
- Instead an operator, + or *, or the end of the input was found.
- Push an imaginary **id** onto the stack and cover it with state 3 (the goto of states 0, 2, 4 and 5 on **id**)
- Issue diagnostic "missing operand"

E2:

- This routine is called from states 0, 1, 2, 4 and 5 on finding a right parenthesis.
- Remove the right parenthesis from the input
- Issue diagnostic "unbalanced right parenthesis"

E3:

- This routine is called from states 1 or 6 when expecting an operator and an **id** or left parenthesis is found.
- Push + onto the stack and cover it with state 4.
- Issue diagnostic "missing operator"

E4:

- This routine is called from state 6 when the end of the input is found
- State 6 expects an operator or a right parenthesis
- Push a right parenthesis onto the stack and cover it with state 9
- Issue diagnostic “missing right parenthesis”

Complete the following parse of **`id + )`**

Stack	Input	Error Message & Action
0	id+)$	