Let L be the language over the alphabet $\Sigma = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, +, -, *, \wedge, (,)\}$ consisting of all arithmetic expressions involving addition, subtraction, multiplication, exponentiation, and negation of non-negative integers, written as base ten numerals. A numeral is any non-empty string of digits, *i.e.*, leading zeros are allowed. For example, the following strings are in L:

00

$2 + 2$

$31 - 7 - -5$ (The value of this expression is 29.)

$-22 * -300 + -(02 - (4 * (8 - 10)) - 173)$

$2 \wedge 3 \wedge 2$ (The value of this expression is 512.)

$2 \wedge -1$ (This expression is a member of L, although it does not have an integral value.)

The following context-free grammar generates L, but is ambiguous:

1. $E \to E + E$
2. $E \to E - E$
3. $E \to E * E$
4. $E \to E \wedge E$
5. $E \to -E$
6. $E \to (E)$
7. $E \to N$
8. $N \to ND$
9. $N \to D$
10. $D \to 0$
11. $D \to 1$
12. $D \to 2$
13. $D \to 3$
14. $D \to 4$
15. $D \to 5$
16. $D \to 6$
17. $D \to 7$
18. $D \to 8$
19. $D \to 9$

The ambiguities are resolved by the following precedence rules:

> Addition, subtraction, and multiplication are left-associative. Exponentiation is right-associative.
>
> The unary operator "$-$" has precedence over exponentiation, which has precedence over multiplication, which has precedence over addition and subtraction. Addition and subtraction have equal precedence.
>
> **Note that the value of $0 - 2 \wedge 2$ is -4, but the value of $-2 \wedge 2$ is 4. Do you see why?**

Design an LALR parser for L using this grammar, such that the resulting parse tree respects the semantics defined above. To aid in this task, I have rewritten the grammar with stack states:

1. $E \rightarrow E_{1,2} +_8 E_3$
2. $E \rightarrow E_{1,2} -_9 E_4$
3. $E \rightarrow E_{1,2,3,4} *_{10} E_5$
4. $E \rightarrow E_{1,2,3,4,5,6} \wedge_{11} E_6$
5. $E \rightarrow -_{12} E_7$
6. $E \rightarrow (_{14} E_2)_{15}$
7. $E \rightarrow N_{16}$
8. $N \rightarrow N_{16} D_{17}$
9. $N \rightarrow D_{18}$
10. $D \rightarrow 0_{20}$
11. $D \rightarrow 1_{21}$
12. $D \rightarrow 2_{22}$
13. $D \rightarrow 3_{23}$
14. $D \rightarrow 4_{24}$
15. $D \rightarrow 5_{25}$
16. $D \rightarrow 6_{26}$
17. $D \rightarrow 7_{27}$
18. $D \rightarrow 8_{28}$
19. $D \rightarrow 9_{29}$

Fill in the ACTION and GOTO tables below. The bottom of the stack state is 0, and the last stack state before halting is 1.

	0	1	2	3	4	5	6	7	8	9	+	−	*	∧	(	)	**eof**	*E*	*N*	*D*
0																				
1																	**halt**			
2																				
3																				
4																				
5																				
6																				
7																				
8																				
9																				
10																				
11																				
12																				
14																				
15																				
16																				
17																				
18																				
20																				
21																				
22																				
23																				
24																				
25																				
26																				
27																				
28																				
29																				

Challenge Problem

Let L_{valid} be the subset of L consisting of all those expressions every subexpression of which has an integral value. Is L_{valid} context-free?

If you are not sure, guess. If you cannot prove your answer, write a paragraph explaining what motivated you to make that guess.