LL(1) Grammar

1. exp → exp2 xor
2. xor → ^ exp
3. | ε
4. exp2 → factor and
5. and → & exp2
6. | ε
7. factor → 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
8. | (exp)

FIRST, FOLLOW and FIRST+ sets

FIRST(#1) = FIRST (exp2) = FIRST(factor) = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, (}
FIRST(#2) = {^}
FIRST(#3) = {ε}
FIRST(#4) = FIRST(factor) = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, (}
FIRST(#5) = {&}
FIRST(#6) = {ε}
FIRST(#7) = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9,}
FIRST(#8) = {(}

FOLLOW(exp) = {EOF,)}
FOLLOW(xor) = FOLLOW(exp) = {EOF,)}
FOLLOW(exp2) = {EOF, ^,)}
FOLLOW(and) = FOLLOW(exp2) = {EOF, ^,)}
FOLLOW(factor) = {EOF, ^, &,)}

FIRST+(#1) = FIRST(#1)= {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, (}
FIRST+(#2) = FIRST(#2) = {^}
FIRST+(#3) = FIRST(#3) U FOLLOW(xor) = {ε, EOF,)}
FIRST+(#4) = FIRST(#4) = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, (}
FIRST+(#5) = FIRST(#5)= {&}
FIRST+(#6) = FIRST+(and) = FIRST(and) U FOLLOW(and) = {&, ε, EOF, ^,)}
FIRST+(#7) = FIRST(#7) = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}
FIRST+(#8) = FIRST(#8) = {(}

Lookup Table

	0, … , 9	**^**	**&**	**(**	**)**	**EOF**
exp	and xor	Error	Error	exp2 xor	and xor	(do nothing)
xor	Error	exp	Error	Error	ε	ε
exp2	and	factor and	Error	factor end	factor end	(do nothing)
and	Error	ε	& exp2	Error	ε	ε
factor	0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9	Error	Error	(exp)	(do nothing)	(do nothing)