<expression> --> <expression> or <or\_exp> | < expression > xor <or\_exp> | <or\_exp>

                     <or\_exp> --> <or\_exp> and <and\_exp> | <and\_exp>

< and\_exp > --> <and\_exp> = <expr> | <and\_exp /= <expr> | <and\_exp> < <expr> | <and\_exp> <= <expr> | <and\_exp> >= <expr> | <and\_exp> > <expr> | <expr>

<expr> --> - <unary\_expr> | <unary\_expr>

<unary\_expr> --> <unary\_expr> + <term> | <unary\_expr> - <term> | <unary\_expr> & <term> | <unary\_expr> mod < term> | <term>

< term > --> < term > \* < factor > | < term > / < factor > | not < factor > | < factor >

< factor > --> ( < expr > ) | <operand>

<operand> --> a | b | c | d | e

<expr> → <expr> or <e1> | <expr> xor <e1> | <e1>

<e1> → <e1> and <e2> | <e2>

<e2> → <e2> = <e3> | <e2> /= <e3> | <e2> < <e3>

| <e2> <= <e3> | <e2> > <e3> | <e2> >= <e3> | <e3>

<e3> → <e4>

<e4> → <e4> + <e5> | <e4> – <e5> | <e4> & <e5> | <e4> mod <e5> | <e5>

<e5> → <e5> \* <e6> | <e5> / <e6> | not <e5> | <e6>

<e6> → a | b | c | d | e | const | ( <expr> )

BNF