Complete TIPS Grammar Productions

`<program> → TOK_PROGRAM TOK_IDENT TOK_SEMICOLON <block>`   First Token Set: { TOK_PROGRAM }

`<block> → [ TOK_VAR TOK_IDENT TOK_COLON (TOK_INTEGER | TOK_REAL) TOK_SEMICOLON`
           `{ TOK_IDENT TOK_COLON (TOK_INTEGER | TOK_REAL) TOK_SEMICOLON } ]`
        `<compound>`
 	
   First Token Set: { TOK_VAR, TOK_BEGIN }

`<statement> → <assignment>	<compound>	<if>	<while>	<read>	<write>` First Token Set: {TOK_IDENT, TOK_BEGIN, TOK_IF, TOK_WHILE, TOK_READ, TOK_WRITE}
`<assignment> → TOK_IDENT TOK_ASSIGN <expression>`   First Token Set: { TOK_IDENT }
`<compound> → TOK_BEGIN <statement> { TOK_SEMICOLON <statement> } TOK_END`   First Token Set: {TOK_BEGIN}
`<if> → TOK_IF <expression> TOK_THEN <statement> [ TOK_ELSE <statement> ]`   First Token Set: { TOK_IF }
`<while> → TOK_WHILE <expression> <statement>`   First Token Set: { TOK_WHILE }
`<read> → TOK_READ TOK_OPENPAREN TOK_IDENT TOK_CLOSEPAREN`   First Token Set: { TOK_READ }
`<write> → TOK_WRITE TOK_OPENPAREN ( TOK_IDENT | TOK_STRINGLIT ) TOK_CLOSEPAREN`   First Token Set: { TOK_WRITE }

`<expression> → <simple expression>`
          `[ ( TOK_EQUALTO | TOK_LESSTHAN | TOK_GREATERTHAN | TOK_NOTEQUALTO )`
            `<simple expression> ]`
 	
   First Token Set: { TOK_INTLIT, TOK_FLOAT_LIT, TOK_IDENT, TOK_OPENPAREN,
                TOK_NOT, TOK_MINUS }

`<simple expression> → <term> { ( TOK_PLUS | TOK_MINUS | TOK_OR ) <term> }`
 	
   First Token Set: { TOK_INTLIT, TOK_FLOAT_LIT, TOK_IDENT, TOK_OPENPAREN,
                TOK_NOT, TOK_MINUS }

`<term> → <factor> { ( TOK_MULTIPLY | TOK_DIVIDE | TOK_AND ) <factor> }`
 	
   First Token Set: { TOK_INTLIT, TOK_FLOAT_LIT, TOK_IDENT, TOK_OPENPAREN,
                TOK_NOT, TOK_MINUS }

`<factor> → TOK_INTLIT	TOK_FLOATLIT	TOK_IDENT	`         `(TOK_OPENPAREN <expression> TOK_CLOSEPAREN)	`         `(TOK_NOT <factor>)	(TOK_MINUS <factor>)` First Token Set: { TOK_INTLIT, TOK_FLOAT_LIT, TOK_IDENT, TOK_OPENPAREN, TOK_NOT, TOK_MINUS }

Note: Symbols shown in red are EBNF symbols, not actual symbols expected.