

# MODIFIED GRAMMAR, FIRST SETS AND FOLLOW SETS

## MODIFIED GRAMMAR

```
<program> ==> <otherFunctions> <mainFunction>
<mainFunction> ==> TK_MAIN <stmts> TK_END
<otherFunctions> ==> <function> <otherFunctions> | eps
<function> ==> TK_FUNID <input_par> <output_par> TK_SEM <stmts> TK_END
<input_par> ==> TK_INPUT TK_PARAMETER TK_LIST TK_SQL <parameter_list>
TK_SQR
<output_par> ==> TK_OUTPUT TK_PARAMETER TK_LIST TK_SQL <parameter_list>
TK_SQR | eps
<parameter_list> ==> <dataType> TK_ID <remaining_list>
<dataType> ==> <primitiveDatatype> | <constructedDatatype>
<primitiveDatatype> ==> TK_INT | TK_REAL
<constructedDatatype> ==> TK_RECORD TK_RECORDID
<remaining_list> ==> TK_COMMA <parameter_list> | eps
<stmts> ==> <typeDefinitions> <declarations> <otherStmts> <returnStmt>
<typeDefinitions> ==> <typeDefinition> <typeDefinitions> | eps
<typeDefinition> ==> TK_RECORD TK_RECORDID <fieldDefinitions>
TK_ENDRECORD TK_SEM
<fieldDefinitions> ==> <fieldDefinition> <fieldDefinition> <moreFields>
<fieldDefinition> ==> TK_TYPE <primitiveDatatype> TK_COLON TK_FIELDID
TK_SEM
<moreFields> ==> <fieldDefinition> <moreFields> | eps
<declarations> ==> <declaration> <declarations> | eps
<declaration> ==> TK_TYPE <dataType> TK_COLON TK_ID TK_COLON
<global_or_not> TK_SEM
<global_or_not> ==> TK_GLOBAL | eps
<otherStmts> ==> <stmt> <otherStmts> | eps
<stmt> ==> <assignmentStmt> | <iterativeStmt> | <conditionalStmt> |
<ioStmt> | <funCallStmt>
<assignmentStmt> ==> <singleOrRecId> TK_ASSIGNOP <arithmeticExpression>
TK_SEM
<singleOrRecId> ==> TK_ID | TK_RECORDID TK_DOT TK_FIELDID
<funCallStmt> ==> <outputParameters> TK_CALL TK_FUNID TK_WITH
TK_PARAMETERS <inputParameters>
<outputParameters> ==> TK_SQL <idList> TK_SQR TK_ASSIGNOP | eps
<inputParameters> ==> TK_SQL <idList> TK_SQR
<iterativeStmt> ==> TK_WHILE TK_OP <booleanExpression> TK_CL <stmt>
<otherStmts> TK_ENDWHILE
<conditionalStmt> ==> TK_IF <booleanExpression> TK_THEN <stmt>
<otherStmts> <conditionalSuffix>
<conditionalSuffix> ==> TK_ELSE <otherStmts> TK_ENDIF | TK_ENDIF
<ioStmt> ==> TK_READ TK_OP <allVar> TK_CL TK_SEM | TK_WRITE TK_OP
<allVar> TK_CL TK_SEM
<allVar> ==> <var> | TK_RECORDID
<arithmeticExpression> ==> <arithmeticT> <arithmeticE1>
```

```
<arithmeticE1> ==> <operator_plusMinus> <arithmeticT> <arithmeticE1> |  
eps  
<arithmeticT> ==> <arithmeticF> <arithmeticT1>  
<arithmeticT1> ==> <operator_mulDiv> <arithmeticF> <arithmeticT1> | eps  
<arithmeticF> ==> TK_OP <arithmeticExpression> TK_CL | <var>  
<operator_plusMinus> ==> TK_PLUS | TK_MINUS  
<operator_mulDiv> ==> TK_MUL | TK_DIV  
<booleanExpression> ==> TK_OP <booleanExpression> TK_CL <logicalOp>  
TK_OP <booleanExpression> TK_CL | <var> <relationalOp> <var> | TK_NOT  
<booleanExpression>  
<var> ==> TK_ID | TK_NUM | TK_RNUM  
<logicalOp> ==> TK_AND | TK_OR  
<relationalOp> ==> TK_LT | TK_LE | TK_EQ | TK_GT | TK_GE | TK_NE  
<returnStmt> ==> TK_RETURN <optionalReturn> TK_SEM  
<optionalReturn> ==> TK_SQL <idList> TK_SQR | eps  
<idList> ==> TK_ID <more_ids>  
<more_ids> ==> TK_COMMA <idList> | eps
```

## FIRST SETS

```
<program> ==> {TK_FUNID, TK_MAIN}  
<mainFunction> ==> {TK_MAIN}  
<otherFunctions> ==> {TK_FUNID, eps}  
<function> ==> {TK_FUNID}  
<input_par> ==> {TK_INPUT}  
<output_par> ==> {TK_OUTPUT, eps}  
<parameter_list> ==> {TK_INT, TK_REAL, TK_RECORD}  
<dataType> ==> {TK_INT, TK_REAL, TK_RECORD}  
<primitiveDatatype> ==> {TK_INT, TK_REAL}  
<constructedDatatype> ==> {TK_RECORD}  
<remaining_list> ==> {TK_COMMA, eps}  
<stmts> ==> {TK_RECORD, TK_ID, TK_SQL, TK_TYPE, TK_RECORDID, TK_WHILE,  
TK_CALL, TK_IF, TK_READ, TK_WRITE, TK_RETURN}  
<typeDefinitions> ==> {TK_RECORD, eps}  
<typeDefinition> ==> {TK_RECORD}  
<fieldDefinitions> ==> {TK_TYPE}  
<fieldDefinition> ==> {TK_TYPE}  
<moreFields> ==> {TK_TYPE, eps}  
<declarations> ==> {TK_TYPE, eps}  
<declaration> ==> {TK_TYPE}  
<global_or_not> ==> {TK_GLOBAL, eps}  
<otherStmts> ==> {TK_ID, TK_SQL, TK_RECORDID, TK_WHILE, TK_CALL, TK_IF,  
TK_READ, TK_WRITE, eps}  
<stmt> ==> {TK_ID, TK_SQL, TK_RECORDID, TK_WHILE, TK_CALL, TK_IF,  
TK_READ, TK_WRITE}  
<assignmentStmt> ==> {TK_ID, TK_RECORDID}  
<singleOrRecId> ==> {TK_ID, TK_RECORDID}  
<funCallStmt> ==> {TK_SQL, TK_CALL}  
<inputParameters> ==> {TK_SQL}  
<outputParameters> ==> {TK_SQL, eps}  
<iterativeStmt> ==> {TK_WHILE}  
<conditionalStmt> ==> {TK_IF}  
<conditionalSuffix> ==> {TK_ELSE, TK_ENDIF}
```

```
<ioStmt> ==> {TK_READ, TK_WRITE}
<allVar> ==> {TK_ID, TK_NUM, TK_RECORDID, TK_RNUM}
<arithmeticExpression> ==> {TK_OP, TK_ID, TK_NUM, TK_RNUM}
<arithmeticE1> ==> {TK_PLUS, TK_MINUS, eps}
<arithmeticT> ==> {TK_OP, TK_ID, TK_NUM, TK_RNUM}
<arithmeticT1> ==> {TK_MUL, TK_DIV, eps}
<arithmeticF> ==> {TK_OP, TK_ID, TK_NUM, TK_RNUM}
<operator_plusMinus> ==> {TK_PLUS, TK_MINUS}
<operator_mulDiv> ==> {TK_MUL, TK_DIV}
<booleanExpression> ==> {TK_OP, TK_ID, TK_NUM, TK_NOT, TK_RNUM}
<var> ==> {TK_ID, TK_NUM, TK_RNUM}
<logicalOp> ==> {TK_AND, TK_OR}
<relationalOp> ==> {TK_LT, TK_LE, TK_EQ, TK_GT, TK_GE, TK_NE}
<returnStmt> ==> {TK_RETURN}
<optionalReturn> ==> {TK_SQL, eps}
<idList> ==> {TK_ID}
<more_ids> ==> {TK_COMMA, eps}
```

## FOLLOW SETS

```
<program> ==> {$}
<mainFunction> ==> {$}
<otherFunctions> ==> {TK_MAIN}
<function> ==> {TK_MAIN, TK_FUNID}
<input_par> ==> {TK_OUTPUT, TK_SEM}
<output_par> ==> {TK_SEM}
<parameter_list> ==> {TK_SQR}
<dataType> ==> {TK_ID, TK_COLON}
<primitiveDatatype> ==> {TK_ID, TK_COLON}
<constructedDatatype> ==> {TK_ID, TK_COLON}
<remaining_list> ==> {TK_SQR}
<stmts> ==> {TK_END}
<typeDefinitions> ==> {TK_TYPE, TK_ID, TK_RECORDID, TK_WHILE, TK_IF,
TK_READ, TK_WRITE, TK_SQL, TK_CALL, TK_RETURN}
<typeDefinition> ==> {TK_RECORD, TK_TYPE, TK_ID, TK_RECORDID, TK_WHILE,
TK_IF, TK_READ, TK_WRITE, TK_SQL, TK_CALL, TK_RETURN}
<fieldDefinitions> ==> {TK_ENDRECORD}
<fieldDefinition> ==> {TK_TYPE, TK_ENDRECORD}
<moreFields> ==> {TK_ENDRECORD}
<declarations> ==> {TK_ID, TK_RECORDID, TK_WHILE, TK_IF, TK_READ,
TK_WRITE, TK_SQL, TK_CALL, TK_RETURN}
<declaration> ==> {TK_TYPE, TK_ID, TK_RECORDID, TK_WHILE, TK_IF,
TK_READ, TK_WRITE, TK_SQL, TK_CALL, TK_RETURN}
<global_or_not> ==> {TK_SEM}
<otherStmts> ==> {TK_RETURN, TK_ENDWHILE, TK_ELSE, TK_ENDIF}
<stmt> ==> {TK_RETURN, TK_ENDWHILE, TK_ELSE, TK_ENDIF, TK_ID,
TK_RECORDID, TK_WHILE, TK_IF, TK_READ, TK_WRITE, TK_SQL, TK_CALL}
<assignmentStmt> ==> {TK_RETURN, TK_ENDWHILE, TK_ELSE, TK_ENDIF, TK_ID,
TK_RECORDID, TK_WHILE, TK_IF, TK_READ, TK_WRITE, TK_SQL, TK_CALL}
<singleOrRecId> ==> {TK_ASSIGNOP}
<funCallStmt> ==> {TK_RETURN, TK_ENDWHILE, TK_ELSE, TK_ENDIF, TK_ID,
TK_RECORDID, TK_WHILE, TK_IF, TK_READ, TK_WRITE, TK_SQL, TK_CALL}
<outputParameters> ==> {TK_CALL}
```

```
<inputParameters> ==> {TK_RETURN, TK_ENDWHILE, TK_ELSE, TK_ENDIF,
TK_ID, TK_RECORDID, TK_WHILE, TK_IF, TK_READ, TK_WRITE, TK_SQL, TK_CALL}
<iterativeStmt> ==> {TK_RETURN, TK_ENDWHILE, TK_ELSE, TK_ENDIF, TK_ID,
TK_RECORDID, TK_WHILE, TK_IF, TK_READ, TK_WRITE, TK_SQL, TK_CALL}
<conditionalStmt> ==> {TK_RETURN, TK_ENDWHILE, TK_ELSE, TK_ENDIF,
TK_ID, TK_RECORDID, TK_WHILE, TK_IF, TK_READ, TK_WRITE, TK_SQL, TK_CALL}
<conditionalSuffix> ==> {TK_RETURN, TK_ENDWHILE, TK_ELSE, TK_ENDIF,
TK_ID, TK_RECORDID, TK_WHILE, TK_IF, TK_READ, TK_WRITE, TK_SQL, TK_CALL}
<ioStmt> ==> {TK_RETURN, TK_ENDWHILE, TK_ELSE, TK_ENDIF, TK_ID,
TK_RECORDID, TK_WHILE, TK_IF, TK_READ, TK_WRITE, TK_SQL, TK_CALL}
<allVar> ==> {TK_CL}
<arithmeticExpression> ==> {TK_CL, TK_SEM}
<arithmeticEl> ==> {TK_CL, TK_SEM}
<arithmeticT> ==> {TK_PLUS, TK_MINUS, TK_CL, TK_SEM}
<arithmeticT1> ==> {TK_PLUS, TK_MINUS, TK_CL, TK_SEM}
<arithmeticF> ==> {TK_MUL, TK_DIV, TK_PLUS, TK_MINUS, TK_CL, TK_SEM}
<operator_plusMinus> ==> {TK_OP, TK_ID, TK_NUM, TK_RNUM}
<operator_mulDiv> ==> {TK_OP, TK_ID, TK_NUM, TK_RNUM}
<booleanExpression> ==> {TK_CL, TK_THEN}
<var> ==> {TK_LT, TK_LE, TK_EQ, TK_GT, TK_GE, TK_NE, TK_CL, TK_THEN,
TK_PLUS, TK_MUL, TK_MINUS, TK_DIV, TK_SEM}
<logicalOp> ==> {TK_OP}
<relationalOp> ==> {TK_ID, TK_NUM, TK_RNUM}
<returnStmt> ==> {TK_END}
<optionalReturn> ==> {TK_SEM}
<idList> ==> {TK_SQR}
<more_ids> ==> {TK_SQR}
```