```
/* GROUP 51

Arnav Agarwal:                2019B2A70966P

Aviral Omar:                  2019B3A70411P

Chandra Sekhar Reddy E:       2019B4A70634P

Vatsal Pattani:               2019B5A70697P

*/

program: dollar

moduleDeclarations: DEF DRIVERDEF

otherModules: DRIVERDEF dollar

driverModule: DEF dollar

moduleDeclaration: DECLARE DEF DRIVERDEF

module: DEF DRIVERDEF dollar

moduleDef: DEF dollar DRIVERDEF

inputPList: SQBC

ret: START

outputPList: SQBC

dataType: COMMA SQBC SEMICOL

iPList2: SQBC

type: COMMA SQBC SEMICOL

oPList2: SQBC

arrRange: SQBC

signedIndex: RANGEOP SQBC

statements: END BREAK

statement: GET_VALUE PRINT DECLARE SWITCH FOR WHILE ID SQBO USE END BREAK

ioStmt: GET_VALUE PRINT DECLARE SWITCH FOR WHILE ID SQBO USE END BREAK

simpleStmt: GET_VALUE PRINT DECLARE SWITCH FOR WHILE ID SQBO USE END BREAK

declareStmt: GET_VALUE PRINT DECLARE SWITCH FOR WHILE ID SQBO USE END BREAK
```

conditionalStmt: GET_VALUE PRINT DECLARE SWITCH FOR WHILE ID SQBO USE END BREAK

iterativeStmt: GET_VALUE PRINT DECLARE SWITCH FOR WHILE ID SQBO USE END BREAK

varPrint: BC

arrIndex: BC

boolConst: BC COLON COMMA SEMICOL MUL DIV PLUS MINUS SQBC LT LE GT GE EQ NE AND OR

moduleReuseStmt: GET_VALUE PRINT DECLARE SWITCH FOR WHILE ID SQBO USE END BREAK

assignmentStmt: GET_VALUE PRINT DECLARE SWITCH FOR WHILE ID SQBO USE END BREAK

whichStmt: GET_VALUE PRINT DECLARE SWITCH FOR WHILE ID SQBO USE END BREAK

lValueIDStmt: GET_VALUE PRINT DECLARE SWITCH FOR WHILE ID SQBO USE END BREAK

lValueArrStmt: GET_VALUE PRINT DECLARE SWITCH FOR WHILE ID SQBO USE END BREAK

expression: SEMICOL

indexWithExpressions: SQBC

sign: NUM ID RNUM TRUE FALSE BO

index: RANGEOP SQBC

optional: USE

actualPList: SEMICOL

param: COMMA SEMICOL

actualPList2: SEMICOL

arrIndexWithExpressions: COMMA SEMICOL MUL DIV PLUS MINUS BC LT LE GT GE EQ NE AND OR

idList: SQBC COLON

idList2: SQBC COLON

arithmeticOrLogicalExpr: SEMICOL BC

unaryOpExpr: SEMICOL

unaryOp: BO ID NUM RNUM

unsignedArithExpr: SEMICOL

arithmeticExpr: BC LT LE GT GE EQ NE AND OR SEMICOL

varIDNum: SEMICOL

anyTerm: AND OR SEMICOL BC

logicalOpExpr: SEMICOL BC

logicalOp: BO NUM RNUM TRUE FALSE ID

relationOpExpr: AND OR SEMICOL BC

relationalOp: BO NUM RNUM TRUE FALSE ID

term: PLUS MINUS BC LT LE GT GE EQ NE AND OR SEMICOL

addSubExpr: BC LT LE GT GE EQ NE AND OR SEMICOL

addSubOp: BO NUM RNUM TRUE FALSE ID

factor: MUL DIV PLUS MINUS BC LT LE GT GE EQ NE AND OR SEMICOL

mulDivExpr: PLUS MINUS BC LT LE GT GE EQ NE AND OR SEMICOL

mulDivOp: BO NUM RNUM TRUE FALSE ID

arrExpr: SQBC BC

arrTerm: PLUS MINUS SQBC BC

arrAddSubExpr: SQBC BC

arrFactor: MUL DIV PLUS MINUS SQBC BC

arrMulDivExpr: PLUS MINUS SQBC BC

caseStmts: DEFAULT END

defaultCase: END

caseValue: COLON

caseStmts2: DEFAULT END

forLoopRange: BC

forLoopIndex: RANGEOP BC