"Start Symbol"=<program>

"Case Sensitive"='true'

number = {digit}+

string = '"'{letter}+'"'

char = '"'{letter}'"'

identifier = {letter}{Alphanumeric}\*

type = '@'{letter}+

<program> ::= <blocks>

<blocks> ::= <block>

| <block> <Blocks>

<block> ::= identifier '{' <statements> '}'

<General-Block> ::= '{' <statements> '}'

<identOrAssign> ::= <identification> | <Assignment>

<numOrString> ::= number

| string

<statements> ::= <statement>

| <statements> <statement>

<statement> ::= <identification>

| <Assignment>

| <arthimetic> '$'

| <for-stmt> '$'

| <switch-stmt> '$'

| <control-stmt> '$'

| <if-stmt> '$'

| <case-stmt> '$'

| <identification> '$'

| <function-stmt> '$'

| <call-stmt> '$'

| <incrementOrDecrement> '$'

| <>

<identification> ::= type identifier '=' number '$'

| type identifier '=' char '$'

| type identifier '=' string '$'

| type identifier '^' '=' number '$'

| type identifier '^' '=' string '$'

| type type identifier '^^' '=' number string '$'

| type identifier '=' <arthimetic> '$'

<operators> ::= '+' | '-' | '\*' | '/' | '\*=' | '/='

<Assignment> ::= identifier '=' string '$'

| identifier '=' number '$'

| identifier '=' char '$'

| identifier '=' <arthimetic> '$'

<arthimetic> ::= <arthimetic> '++' <operators> <arthimeticOp>

| <arthimetic> '--' <operators> <arthimeticOp>

| <arthimetic> '\*=' <arthimeticOp>

| <arthimetic> '/=' <arthimeticOp>

| <arthimeticOp>

<arthimeticOp> ::= <arthimeticOp> '+' <arthimeticOpMult>

| <arthimeticOp> '-' <arthimeticOpMult>

| <arthimeticOpMult>

<arthimeticOpMult> ::= <arthimeticOpMult> '/' <arthimeticOpNegate>

| <arthimeticOpMult> '\*' <arthimeticOpNegate>

| <arthimeticOpNegate>

<arthimeticOpNegate> ::= '-' <value>

| <value>

<value> ::= number

| identifier

| '(' <arthimetic> ')'

<incrementOrDecrement> ::= identifier'++' | identifier'--'

<logical-exp> ::= identifier '==' <logical-exp>

| identifier '<=' <logical-exp>

| identifier '>=' <logical-exp>

| identifier '<' <logical-exp>

| identifier '>' <logical-exp>

| identifier '=+' <logical-exp>

| identifier '=-' <logical-exp>

| identifier '=\*' <logical-exp>

| identifier '=/' <logical-exp>

| identifier

<if-stmt> ::= 'if''('<logical-exp>')'<General-Block>'else'<General-Block>

| 'if''('<logical-exp>')'<General-Block>

<for-stmt> ::= 'Loop''(''From'<identOrAssign>'To'<identOrAssign>'By'<Assignment>'While'<logical-exp>')'<General-Block>

<case-stmts> ::= <case-stmt>

| <case-stmts> <case-stmt>

<case-stmt> ::= 'case'':'<General-Block>

| 'default'':'<General-Block>

<control-stmt> ::= 'gout'

| 'continue'

| 'goto:'identifier

| identifier':'

| 'backWith'identifier

<switch-stmt> ::= 'Chooser''('identifier')''{'<case-stmts>'}'

<function-stmt> ::= 'func'identifier'('identifier'$'identifier')''{'<statements>'}'

<call-stmt> ::= 'call'identifier'('<numOrString>'$'<numOrString>')'