

Grammar of the Language

program--> ['{'],command,['}'].

command --> command,command.

command --> id, [=], expr,[:].

command --> [const], id, [=], expr,[:].

command --> [if], ['('], bool_expr, [')'], ['{'], command, ['}'],[else], ['{'], command, ['}'].

command --> bool_expr,[?],command,[:],command.

command --> [while], ['('], bool_expr, [')'], ['{'], command, ['}'].

command --> [for],['('], id, [=], expr, [:], bool_expr, [:],id, [=], expr,[')'],['{'],command,['}'].

command --> [do], ['{'], command, ['}'],[while], ['('], bool_expr, [')'], [';'].

command --> [print],['('],text,[')'],[:].

text --> id,text.

text --> id.

bool_expr --> expr,['=='],expr.

bool_expr --> expr,['<='],expr.

bool_expr --> expr,['<'],expr.

bool_expr --> expr,['>='],expr.

bool_expr --> expr,['>'],expr.

bool_expr --> [not],bool_expr.

bool_expr --> bool_expr,[and],bool_expr.

bool_expr --> bool_expr, [or],bool_expr.

bool_expr --> [true].

bool_expr --> [false].

expr --> id,[=],expr1.

expr --> expr1.

expr1 --> expr1, [+], term.

expr1 --> expr1, [-], term.

expr1--> term.

term--> term, [*], num.

term--> term, [/], num.

term --> ['(',')',expr,['(',')']].

term --> num.

term --> id.

num --> [X], {number(X)}.

id --> [X], {atom(X)}.