1) Types and Constants

<LP> -> "("

<RP> -> ")"

<coma> -> ","

<dot> -> "."

<colon> -> ":"

<comment> -> "*"

<exit> -> ">>"

<assign> -> "-->"

<multiplication> -> ".*"

<division> -> "./"

<addition> -> ".+"

<subtraction> -> ".-"

<equal> -> ".="

<not_equal> -> ".!"

<greater_or_equal> -> ".>="

<lower_or_equal> -> ".<="

<greater> -> ".>"

<lower> -> ".<"

<or> -> "//"

<and> -> "^^"

<lowercase> -> a|b|...|z

<uppercase> -> A|B|...|Z

<non_zero_digit> -> 1|2|...|9

<digit> -> 0|1|...|9

<sign> -> +|-

<non_digit_char> -> <underscore>|<lowercase>|<uppercase>

<number> -> <non_zero_digit>|<digit><number>

<identifier> -> (<lowercase>|<uppercase>)(<non_digit_char>|<digit>)

<integer> -> <sign>?<number>

2)Program Definition

<program> -> <start><LP><RP><statement><end>ele

<statements> -> <statement>|<statement><colon><statements>

<statement> -> <comment>|<assignment>|<expr>|<loops>|<conditional_stmt>

<assignment> -> <identifier><assign><args_type>

<expr> -> <integer><lower><integer>

|<integer><greater><integer>

|<integer><greater_or_equal><integer>

|<integer><lower_or_equal><integer>

|<identifier><lower><identifier>

|<identifier><greater><identifier>

|<identifier><lower_or_equal><identifier>

|<identifier><greater_or_equal><identifier>

|<identifier><and><identifier>

|<identifier><or><identifier>

|<identifier><equal><identifier>

|<identifier><not_equal><identifier>

|<identifier><plus><identifier>

|<identifier><minus><identifier>

|<identifier><mult><identifier>

|<identifier> <identifier>

|<integer><plus><integer>

|<integer><minus><integer>

|<integer><mult><integer>

|<integer> <integer>

|<float><plus><float>

|<float><minus><float>

|<float><mult><float>

|<float> <float>

<loops> -> <loops_stmt>|<for_stmt>

<loop_stmt> -> <loop><colon><funct_call><exit>

<funct_call> -> <identifier><colon><args><exit>

<args> -> <composite_args>

<composite_args> -> <args_type><comma><composite_args>|<args_type>

<args_type> -> <integer>|<float>

<for_stmt> -> <for><colon><expr><exit>

<conditional_stmt> -> <condition><colon><funct_call><exit>

| <else><colon><statemenet><exit>