

## 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><div><identifier>

|<integer><plus><integer>

|<integer><minus><integer>

|<integer><mult><integer>

|<integer><div><integer>

|<float><plus><float>

|<float><minus><float>

|<float><mult><float>

|<float><div><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>