**CONTEXT-FREE GRAMMAR**

|  |  |  |  |
| --- | --- | --- | --- |
|  | <program> | 🡪 | <global>Task Lead: Start <body> End.<TaskDef> |
|  | <global> | 🡪 | <varlet><array><task><object><global> |
|  | <global> | 🡪 | Λ |
|  | <object> | 🡪 | Object id Start Varid as <dtype>. <objdec>End<varname>. |
|  | <object> | 🡪 | Λ |
|  | <objdec> | 🡪 | <objvar><var><objdec> ending |
|  | <objdec> | 🡪 | Λ |
|  | <dtype> | 🡪 | Int |
|  | <dtype> | 🡪 | Double |
|  | <dtype> | 🡪 | Char |
|  | <dtype> | 🡪 | String |
|  | <dtype> | 🡪 | Boolean |
|  | <var> | 🡪 | Var id as <dtype>. |
|  | <var> | 🡪 | Λ |
|  | <objvar> | 🡪 | Object id id. |
|  | <objvar> | 🡪 | Λ |
|  | <varname> | 🡪 | id <varnames> |
|  | <varname> | 🡪 | Λ |
|  | <varnames> | 🡪 | ; <varname> |
|  | <varnames> | 🡪 | Λ |
|  | <task> | 🡪 | Task id <tparam> as <dtype>. |
|  | <task> | 🡪 | Task id <tparam> as Null. |
|  | <task> | 🡪 | Λ |
|  | <tparam> | 🡪 | (Var id as <dtype><tparams>) |
|  | <tparam> | 🡪 | Λ |
|  | <tparams> | 🡪 | ;Var id as <dtype><tparams> |
|  | <tparams> | 🡪 | Λ |
|  | <array> | 🡪 | Array id of <size> as <dtype>. |
|  | <array> | 🡪 | Λ |
|  | <varlet> | 🡪 | <type> id <init>. |
|  | <varlet> | 🡪 | Λ |
|  | <type> | 🡪 | Var |
|  | <type> | 🡪 | Let |
|  | <init> | 🡪 | <value1><ids1> as Int |
|  | <init> | 🡪 | <value2><ids2> as Double |
|  | <init> | 🡪 | <value3><ids3> as Char |
|  | <init> | 🡪 | <value4><ids4> as String |
|  | <init> | 🡪 | <value5><ids5> as Boolean |
|  | <ids1> | 🡪 | ; id <value1><ids1> |
|  | <ids2> | 🡪 | ; id <value2><ids2> |
|  | <ids3> | 🡪 | ; id <value3><ids3> |
|  | <ids4> | 🡪 | ; id <value4><ids4> |
|  | <ids5> | 🡪 | ; id <value5><ids5> |
|  | <ids1> | 🡪 | Λ |
|  | <ids2> | 🡪 | Λ |
|  | <ids3> | 🡪 | Λ |
|  | <ids4> | 🡪 | Λ |
|  | <ids5> | 🡪 | Λ |
|  | <size> | 🡪 | Numlit<sizes> |
|  | <sizes> | 🡪 | by Numlit |
|  | <sizes> | 🡪 | Λ |
|  | <dtype1> | 🡪 | Int |
|  | <dtype2> | 🡪 | Double |
|  | <dtype3> | 🡪 | Char |
|  | <dtype4> | 🡪 | String |
|  | <dtype5> | 🡪 | Boolean |
|  | <value1> | 🡪 | is <tvalue1> |
|  | <value1> | 🡪 | Λ |
|  | <value2> | 🡪 | is <tvalue2> |
|  | <value2> | 🡪 | Λ |
|  | <value3> | 🡪 | is <tvalue3> |
|  | <value3> | 🡪 | Λ |
|  | <value4> | 🡪 | is <tvalue4> |
|  | <value4> | 🡪 | Λ |
|  | <value5> | 🡪 | is <tvalue5> |
|  | <value5> | 🡪 | Λ |
|  | <tvalue1> | 🡪 | <numvalue><Operations1> |
|  | <numvalue> | 🡪 | Numlit |
|  | <numvalue> | 🡪 | <incdecvar> |
|  | <numvalue> | 🡪 | <objelement> |
|  | <tvalue1> | 🡪 | (<tvalue1>)<Op1> |
|  | <Operations1> | 🡪 | <mathOp><numvalue><Operations1> |
|  | <Operations1> | 🡪 | Λ |
|  | <Op1> | 🡪 | <Operations1> |
|  | <Op1> | 🡪 | <mathOp>(<tvalue1>) |
|  | <Op1> | 🡪 | Λ |
|  | <tvalue2> | 🡪 | <doublevalue><Operations2> |
|  | <doublevalue> | 🡪 | Doublelit |
|  | <doublevalue> | 🡪 | <incdecvar> |
|  | <doublevalue> | 🡪 | <objelement> |
|  | <tvalue2> | 🡪 | (<tvalue2>)<Op2> |
|  | <Operations2> | 🡪 | <mathOp><doublevalue><Operations2> |
|  | <Operations2> | 🡪 | Λ |
|  | <Op2> | 🡪 | <Operations2> |
|  | <Op2> | 🡪 | <mathOp>(<tvalue2>) |
|  | <Op2> | 🡪 | Λ |
|  | <tvalue3> | 🡪 | Charlit |
|  | <tvalue4> | 🡪 | Stringlit |
|  | <tvalue5> | 🡪 | Boollit |
|  | <initvalues> | 🡪 | <taskvar> |
|  | <initvalues> | 🡪 | <tvalue> |
|  | <mathOp> | 🡪 | + |
|  | <mathOp> | 🡪 | - |
|  | <mathOp> | 🡪 | \* |
|  | <mathOp> | 🡪 | / |
|  | <mathOp> | 🡪 | % |
|  | <incdec> | 🡪 | ++ |
|  | <incdec> | 🡪 | -- |
|  | <relop1> | 🡪 | == |
|  | <relop1> | 🡪 | =! |
|  | <relop1> | 🡪 | > |
|  | <relop1> | 🡪 | < |
|  | <relop1> | 🡪 | => |
|  | <relop1> | 🡪 | =< |
|  | <relop2> | 🡪 | == |
|  | <relop2> | 🡪 | =! |
|  | <logop1> | 🡪 | && |
|  | <logop1> | 🡪 | || |
|  | <logop2> | 🡪 | ! |
|  | <logop2> | 🡪 | Λ |
|  | <body> | 🡪 | <statements> |
|  | <statements> | 🡪 | <vardec><varinit><array><arrayinit><io\_statements><taskvar>  <objvar><objinit><condstate><loopstate><option><statements> |
|  | <statements> | 🡪 | Λ |
|  | <vardec> | 🡪 | Var<init>. |
|  | <vardec> | 🡪 | Λ |
|  | <varinit> | 🡪 | id = <varinits>. |
|  | <varinit> | 🡪 | Λ |
|  | <varinits> | 🡪 | <tvalue>. |
|  | <varinits> | 🡪 | <taskvar> |
|  | <io\_statement> | 🡪 | <input><output> |
|  | <input> | 🡪 | Read id. |
|  | <input> | 🡪 | Λ |
|  | <output> | 🡪 | Say (<input\_statement>). |
|  | <output> | 🡪 | Λ |
|  | <input\_statement> | 🡪 | Stringlit<concat> |
|  | <input\_statement> | 🡪 | id<concat> |
|  | <concat> | 🡪 | , <concat\_value><concat> |
|  | <concat> | 🡪 | Λ |
|  | <concat\_value> | 🡪 | Stringlit |
|  | <concat\_value> | 🡪 | Id |
|  | <taskvar> | 🡪 | id(<param>). |
|  | <taskvar> | 🡪 | Λ |
|  | <param> | 🡪 | <tvalue><params> |
|  | <param> | 🡪 | Λ |
|  | <params> | 🡪 | ; <tvalue><params> |
|  | <params> | 🡪 | Λ |
|  | <tvalue> | 🡪 | <tvalue1> |
|  | <tvalue> | 🡪 | <tvalue2> |
|  | <tvalue> | 🡪 | <tvalue3> |
|  | <tvalue> | 🡪 | <tvalue4> |
|  | <tvalue> | 🡪 | <tvalue5> |
|  | <incdecvar> | 🡪 | <incdec>id |
|  | <incdecvar> | 🡪 | id<incdec> |
|  | <objelement> | 🡪 | <incdec><element> |
|  | <objelement> | 🡪 | <element><incdec> |
|  | <element> | 🡪 | id@id<subelement> |
|  | <subelement> | 🡪 | @id<subelement> |
|  | <subelement> | 🡪 | Λ |
|  | <objinit> | 🡪 | <element> = <tvalue>. |
|  | <arrayinit> | 🡪 | id[<size>] = <tvalue>. |
|  | <cond\_loop> | 🡪 | <vardec><varinit><array><arrayinit><io\_statements><taskvar>  <objvar><objinit><condstate><loopstate><option><c><cond\_loop> |
|  | <cond\_loop> | 🡪 | Λ |
|  | <condstate> | 🡪 | Λ |
|  | <condstate> | 🡪 | If <logop2>(<conditions>)<statements><if2 ><if3>EndIf. |
|  | <if2> | 🡪 | OtherwiseIf<conditions><statements><if2> |
|  | <if2> | 🡪 | Λ |
|  | <if3> | 🡪 | Otherwise <statements><c> |
|  | <conditions> | 🡪 | Id <a><b> |
|  | <conditions> | 🡪 | (Id <a>) <b> |
|  | <conditions> | 🡪 | <logop2>Id |
|  | <conditions> | 🡪 | <logop2>Boollit |
|  | <a> | 🡪 | <relop1><cvalue1> |
|  | <a> | 🡪 | <relop2><cvalue2> |
|  | <a> | 🡪 | Λ |
|  | <b> | 🡪 | <logop1><conditions> |
|  | <b> | 🡪 | Λ |
|  | <cvalue1> | 🡪 | Intlit |
|  | <cvalue1> | 🡪 | Doublit |
|  | <cvalue2> | 🡪 | Charlit |
|  | <cvalue2> | 🡪 | Stringlit |
|  | <cvalue2> | 🡪 | <logop2>Boollit |
|  | <option> | 🡪 | <option1> |
|  | <option> | 🡪 | <option2> |
|  | <option> | 🡪 | <option3> |
|  | <option> | 🡪 | Λ |
|  | <option1> | 🡪 | Option id Start. State intlit: <statements> Stop. <state1> End. |
|  | <option2> | 🡪 | Option id Start. State charlit: <statements> Stop. <state2> End. |
|  | <option3> | 🡪 | Option id Start. State stringlit: <statements> Stop. <state3> End. |
|  | <state1> | 🡪 | State intlit: <statements> Stop. <state1> |
|  | <state1> | 🡪 | Λ |
|  | <state2> | 🡪 | State charlit: <statements> Stop. <state2> |
|  | <state1> | 🡪 | Λ |
|  | <state3> | 🡪 | State stringlit: <statements> Stop. <state3> |
|  | <state1> | 🡪 | Λ |
|  | <c> | 🡪 | Skip. |
|  | <c> | 🡪 | Stop. |
|  | <c> | 🡪 | Λ |
|  | <loopstate> | 🡪 | Until (<conditions>)Start <cond\_loop> End. |
|  | <loopstate> | 🡪 | Do < cond\_loop >Until(<conditions>). |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |