

Terminales

Command

(
)
def var
=
move
turn
: left
: right
: around
: north
: south
: east
: west
put
pick
Balloons
Chips
: front
: back
move - dir
run - dirs
move - face
skip

Control structure

Conditional

if

Condition

faan g-p
can-put-p
can-pick-p
can-move-p
not

Repeat

loop

Repeat times

repeat

Function

defun

both

- numbers
(0, 1, 2, 3, 4, 5, 6, 7, 8, 9)
- letters
(a ... z)
- space

No terminales:

Com
↓
Comando

name
↓
name

num
↓
number

ConStr
↓
Control
structure
↓
Cond
↓
condition

Bl

Params
↓
Parameters

I
↓
Instruction

l
↓
Letter

dig
↓
digito

Reglas producción

Both:

l → (a | ... | z)
dig → (0 | ... | 9)
num → (num dig | dig)
name → (dig | l)

num → dig {dig }

Comandos:

com → (defvar Comb 1)
com → (= Comb 1)

com \rightarrow (move Comb 2)
 com \rightarrow (turn (:left | :right | :around))
 com \rightarrow (face Cardinal)
 com \rightarrow (put Comb 32)
 com \rightarrow (pick Comb 32)
 com \rightarrow (move-dir Comb 2 (:front | :right | :left | :back))
 com \rightarrow (run-dirs Ds)
 com \rightarrow (move-face Comb 1 Cardinal)
 com \rightarrow (skip)

Condiciones

cond \rightarrow (facing-p Cardinal)
 cond \rightarrow (can-put-p Comb 23)
 cond \rightarrow (can-pick-p Comb 23)
 cond \rightarrow (can-move-p Cardinal)
 cond \rightarrow (not cond)

Estructuras Control

ConStr \rightarrow (if cond B1 B2)
 ConStr \rightarrow (loop cond B1)
 ConStr \rightarrow (repeat Comb 2 B1)
 ConStr \rightarrow (defun name (Para) B1)

Combinaciones

Comb1 \rightarrow ~~name~~ num

Comb2 \rightarrow ~~(~~ (name | num)

Comb3 \rightarrow ~~(~~ (ballons | Chips)

Comb23 \rightarrow Comb3 Comb2

run - dirs

Ds \rightarrow ((Ds | (:front | :right | :left | :back)))

Face / move-face

Cardinal \rightarrow (:north | :south | :east | :west)

Bl \rightarrow (I)

I \rightarrow {ConStr | com} (ConStr | com)

Para \rightarrow (Para | name)