Formal Grammar Definition

Tokens

Characters

- $[$: Opening Bracket
- $]$: Closing Bracket
- $:$: Colon
- $;$: Semicolon
- $,$: Comma
- $|$: Pipe

Keywords

- $Robot\_R$
- $vars$
- $procs$
- if
- $then$
- $else$
- $while$
- do
- $repeat$

Commands

- $assignTo$
- $goto$
- $move$
- $turn$
- $face$

- put
- $pick$
- $moveToThe$
- $moveInDir$
- $jumpToThe$
- $jumpInDir$
- nop

Conditions

- $facing$
- $canPut$
- $canPick$
- $canMoveInDir$
- $canJumpInDir$
- $canMoveToThe$
- $canJumpToThe$
- not

Special Words

- $north$
- $south$
- $east$
- $west$
- $front$
- $right$
- $left$
- $back$
- $around$
- $chips$
- $balloons$

Others (see implementation)

- $\langle name \rangle$: Identifiers
- $\langle number \rangle$: Numbers
- $\langle illegal \rangle$: Illegal Character
- $\langle EoF \rangle$: End Of File

Variables

- $\langle Program \rangle$: A Program for the Robot
- $\langle Vars \rangle$: Declaration of Variables
- $\langle Procs \rangle$: Procedure Declaration
- $\langle Proc \rangle$: Procedure Definition
- $\langle Block \rangle$: Block of Instructions
- $\langle Instructions \rangle$: Instructions separated by Semicolons
- $\langle Instruction \rangle$: Instruction Structure
- $\langle Control \rangle$: Control Structure
- $\langle Cond \rangle$: Conditional Structure
- $\langle Loop \rangle$: Loop Structure
- $\langle Repeat \rangle$: RepeatTimes Structure
- $\langle Call \rangle$: Procedure Call
- $\langle Cmd \rangle$: Command
- $\langle Cnd \rangle$: Condition
- $\langle NameList \rangle$: Names separated by Commas
- $\langle ArgsList \rangle$: Arguments separated by Commas
- $\langle Arg \rangle$: Argument Structure

Production Rules

The initial symbol is $\langle Program \rangle$

Terminals (tokens) are shown in $blue$.

Non-Terminals (variables) are shown in $black$.

BNF notation is shown in red:

- $\rightarrow$: represents a production rule.

- $(\ )?$: means optional.
- $(\ )*$: means zero or more.
- $\mid$: means OR.

General

- $\langle Program \rangle \rightarrow Robot\_R\ (\langle Vars \rangle)?\ (\langle Procs \rangle)?\ \langle Block \rangle\ \langle EoF \rangle$
- $\langle Vars \rangle \rightarrow vars\ \langle NameList \rangle$
- $\langle Procs \rangle \rightarrow procs\ \langle Proc \rangle\ (\langle Proc \rangle)*$
- $\langle Proc \rangle \rightarrow \langle name \rangle\ [\ |\ (\langle NameList \rangle\ )?\ |\ \langle Instructions \rangle\ ]$
- $\langle Block \rangle \rightarrow [\ \langle Instructions \rangle\ ]$
- $\langle Instructions \rangle \rightarrow \langle Instruction \rangle\ (\ ;\ \langle Instruction \rangle)*$
- $\langle Instruction \rangle \rightarrow \langle Cmd \rangle \mid \langle Control \rangle \mid \langle Call \rangle$
- $\langle Control \rangle \rightarrow \langle Cond \rangle \mid \langle Loop \rangle \mid \langle Repeat \rangle$
- $\langle Cond \rangle \rightarrow if\ :\ \langle Cnd \rangle\ then\ :\ \langle Block \rangle\ else\ :\ \langle Block \rangle$
- $\langle Loop \rangle \rightarrow while\ :\ \langle Cnd \rangle\ do\ :\ \langle Block \rangle$
- $\langle Repeat \rangle \rightarrow repeat\ :\ \langle Arg \rangle \langle Block \rangle$
- $\langle Call \rangle \rightarrow \langle name \rangle\ :\ (\langle ArgsList \rangle)?$
- $\langle NameList \rangle \rightarrow \langle name \rangle\ (\ ,\ \langle name \rangle)*$
- $\langle ArgsList \rangle \rightarrow \langle Arg \rangle\ (\ ,\ \langle Arg \rangle)*$
- $\langle Arg \rangle \rightarrow \langle name \rangle \mid \langle number \rangle$

Commands

- $\langle Cmd \rangle \rightarrow assignTo\ :\ \langle number \rangle\ ,\ \langle name \rangle$

- $\langle Cmd \rangle \rightarrow goto\ :\ \langle Arg \rangle\ ,\ \langle Arg \rangle$
- $\langle Cmd \rangle \rightarrow move\ :\ \langle Arg \rangle$
- $\langle Cmd \rangle \rightarrow turn\ :\ (\ left \mid right \mid around\ )$
- $\langle Cmd \rangle \rightarrow face\ :\ (\ north \mid south \mid east \mid west\ )$
- $\langle Cmd \rangle \rightarrow put\ :\ \langle Arg \rangle\ ,\ (\ balloons \mid chips\ )$
- $\langle Cmd \rangle \rightarrow pick\ :\ \langle Arg \rangle\ ,\ (\ balloons \mid chips\ )$
- $\langle Cmd \rangle \rightarrow moveToThe\ :\ \langle Arg \rangle\ ,\ (\ front \mid right \mid left \mid back\ )$
- $\langle Cmd \rangle \rightarrow moveInDir\ :\ \langle Arg \rangle\ ,\ (\ north \mid south \mid west \mid east\ )$
- $\langle Cmd \rangle \rightarrow jumpToThe\ :\ \langle Arg \rangle\ ,\ (\ front \mid right \mid left \mid back\ )$
- $\langle Cmd \rangle \rightarrow jumpInDir\ :\ \langle Arg \rangle\ ,\ (\ north \mid south \mid west \mid east\ )$
- $\langle Cmd \rangle \rightarrow nop\ :$

Conditions

- $\langle Cnd \rangle \rightarrow facing\ :\ (\ north \mid south \mid east \mid west\ )$
- $\langle Cnd \rangle \rightarrow canPut\ :\ \langle Arg \rangle\ ,\ (\ balloons \mid chips\ )$
- $\langle Cnd \rangle \rightarrow canPick\ :\ \langle Arg \rangle\ ,\ (\ balloons \mid chips\ )$
- $\langle Cnd \rangle \rightarrow canMoveInDir\ :\ \langle Arg \rangle\ ,\ (\ north \mid south \mid west \mid east\ )$
- $\langle Cnd \rangle \rightarrow canJumpInDir\ :\ \langle Arg \rangle\ ,\ (\ north \mid south \mid west \mid east\ )$
- $\langle Cnd \rangle \rightarrow canMoveToThe\ :\ \langle Arg \rangle\ ,\ (\ front \mid right \mid left \mid back\ )$
- $\langle Cnd \rangle \rightarrow canJumpToThe\ :\ \langle Arg \rangle\ ,\ (\ front \mid right \mid left \mid back\ )$
- $\langle Cnd \rangle \rightarrow not\ :\ \langle Cnd \rangle$