

Gramática

Codigo	→	PRprincipal () { Texto
Texto	→	VariablesL Texto Impresión Texto Lectura Texto Condicion Texto CicloM Texto CicloHM CicloH Texto }
VariablesL	→	PRDatoPrimi _ TokId id
id	→	, _ TokId valor id ;
valor	→	= TipoValor e
TipoValor		cadena entero decimal booleano char
Impresión	→	PRImprimir ValImprimir
ValImprimir	→	(Imprimir
Imprimir	→	_ TokId idI cadena idI
idI	→	+ _ TokId idI + cadena idI) ;

Lectura	→	PRLeer ValLeer
ValLeer	→	(_ TokId) ;
Condicion	→	PRSI ValCondicion { CuerpoCond } Cond2
CuerpoCond	→	Texto CuerpoCond TokId = TipoValor ; CuerpoCond e
Cond2	→	PRSINO _SI ValCondicion { CuerpoCond } Cond2 PRSINO { CuerpoCond } e
ValCondicion	→	(SigDif Condicionante CondicionantePlus)
Condicionante	→	Numero OpRel Numero PRBooleana
SigDif	→	! SigDif e
CondicionantePlus	→	OpLogico SigDif Condicionante CondicionantePlus e
CicloW	→	PRHACER ValCondicion { CuerpoCond }
CicloHM	→	PRHACER { CuerpoCond } PR MIENTRAS ValCondicion
CicloF	→	EstructCicloF { CuerpoCond }

EstructCicloF	→	PRDesde desde PRHASTA hasta PRINCREMENTO NumE { CuerpoCond }
desde	→	TokId = NumE
hasta	→	TokId OpRel NumE