



Práctica 2

Procesadores de Lenguajes

Borja Aday Guadalupe Luis
Diego Enrique de Miguel López
Grupo 13

1. Gramáticas del lenguaje.

Gramática	Gramática acondicionada LL(1)
Programa -> PDeclaraciones PInstrucciones PDeclaraciones -> LDecs && PDeclaraciones -> ε LDecs -> LDecs ; Dec LDecs -> Dec Dec -> var Tipo identificador Dec -> type Tipo identificador Dec -> proc identificador ParForm Bloque ParForm -> (LParams) LParams -> LParams , Param LParams -> Param LParams -> ε Param -> Tipo Referencia identificador Referencia -> & Referencia -> ε Bloque -> { Programa } Bloque -> { } Tipo -> int Tipo -> real Tipo -> bool Tipo -> string Tipo -> identificador Tipo -> array [numEnt] of Tipo Tipo -> record { LCampos } LCampos -> LCampos ; Campo LCampos -> Campo Campo -> Tipo identificador Tipo -> pointer Tipo PInstrucciones -> LIns LIns -> LIns ; Ins LIns -> Ins Ins -> E0 = E0 Ins -> if E0 then LInsV endif Ins -> if E0 then LInsV else LInsV endif Ins -> while E0 do LInsV endwhile	Programa -> PDeclaraciones PInstrucciones PDeclaraciones -> LDecs && PDeclaraciones -> ε LDecs -> Dec RLDecs RLDecs -> ; Dec RLDecs RLDecs -> ε Dec -> var Tipo identificador Dec -> type Tipo identificador Dec -> proc identificador ParForm Bloque ParForm -> (LParams) LParams -> Param RLParams RLParams -> , Param RLParams RLParams -> ε LParams -> ε Param -> Tipo Referencia identificador Referencia -> & Referencia -> ε Bloque -> { RBloque } RBloque -> Programa } RBloque -> } Tipo -> int Tipo -> real Tipo -> bool Tipo -> string Tipo -> identificador Tipo -> array [numEnt] of Tipo Tipo -> record { LCampos } LCampos -> Campo RLCampos RLCampos -> ; Campo RLCampos RLCampos -> ε Campo -> Tipo identificador Tipo -> pointer Tipo PInstrucciones -> LIns LIns -> Ins RLIns RLIns -> ; Ins RLIns RLIns -> ε Ins -> E0 = E0 Ins -> InsIfThen RInsIfThen InsIfThen -> if E0 then LInsV RInsIfThen -> endif RInsIfThen -> else LInsV endif Ins -> while E0 do LInsV endwhile

LinsV -> LinsV ; Ins
LinsV -> Ins
LinsV -> ϵ

Ins -> **read** E0
Ins -> **write** E0
Ins -> **nl**
Ins -> **new** E0
Ins -> **delete** E0
Ins -> **call identificador** (LParamsReales)

LParamsReales -> LParamsReales , E0
LParamsReales -> E0
LParamsReales -> ϵ

Ins -> Bloque

E0 -> E1 + E0
E0 -> E1 - E1
E0 -> E1

E1 -> E1 OpN1 E2
E1 -> E2

E2 -> E2 OpN2 E3
E2 -> E3

E3 -> E4 OpN3 E4
E3 -> E4

E4 -> - E5
E4 -> **not** E4
E4 -> E5

E5 -> E5 OpN5
E5 -> E6

E6 -> * E6
E6 -> E7

E7 -> **identificador**
E7 -> **numEnt**
E7 -> **numReal**
E7 -> **true**
E7 -> **false**
E7 -> **cadena**
E7 -> **null**
E7 -> (E0)

OpN1 -> **and**
OpN1 -> **or**

LinsV -> Ins RLinsV
RLinsV -> ; Ins RLinsV
RLinsV -> ϵ
LinsV -> ϵ

Ins -> **read** E0
Ins -> **write** E0
Ins -> **nl**
Ins -> **new** E0
Ins -> **delete** E0
Ins -> **call identificador** (LParamsReales)

LParamsReales -> E0 RLParamsReales
RLParamsReales -> , E0 RLParamsReales
RLParamsReales -> ϵ
LParamsReales -> ϵ

Ins -> Bloque

E0 -> E1 RE0
RE0 -> + E0
RE0 -> - E1
RE0 -> ϵ

E1 -> E2 RE1
RE1 -> OpN1 E2 RE1
RE1 -> ϵ

E2 -> E3 RE2
RE2 -> OpN2 E3 RE2
RE2 -> ϵ

E3 -> E4 RE3
RE3 -> OpN3 E4
RE3 -> ϵ

E4 -> - E5
E4 -> **not** E4
E4 -> E5

E5 -> E6 RE5
RE5 -> OpN5 RE5
RE5 -> ϵ

E6 -> * E6
E6 -> E7

E7 -> **identificador**
E7 -> **numEnt**
E7 -> **numReal**
E7 -> **true**
E7 -> **false**
E7 -> **cadena**
E7 -> **null**
E7 -> (E0)

OpN1 -> **and**
OpN1 -> **or**

OpN2 -> <
OpN2 -> >
OpN2 -> <=
OpN2 -> >=
OpN2 -> ==
OpN2 -> !=

OpN3 -> *
OpN3 -> /
OpN3 -> %

OpN5 -> [E0]
OpN5 -> . **identificador**
OpN5 -> -> **identificador**

OpN2 -> <
OpN2 -> >
OpN2 -> <=
OpN2 -> >=
OpN2 -> ==
OpN2 -> !=

OpN3 -> *
OpN3 -> /
OpN3 -> %

OpN5 -> [E0]
OpN5 -> . **identificador**
OpN5 -> -> **identificador**