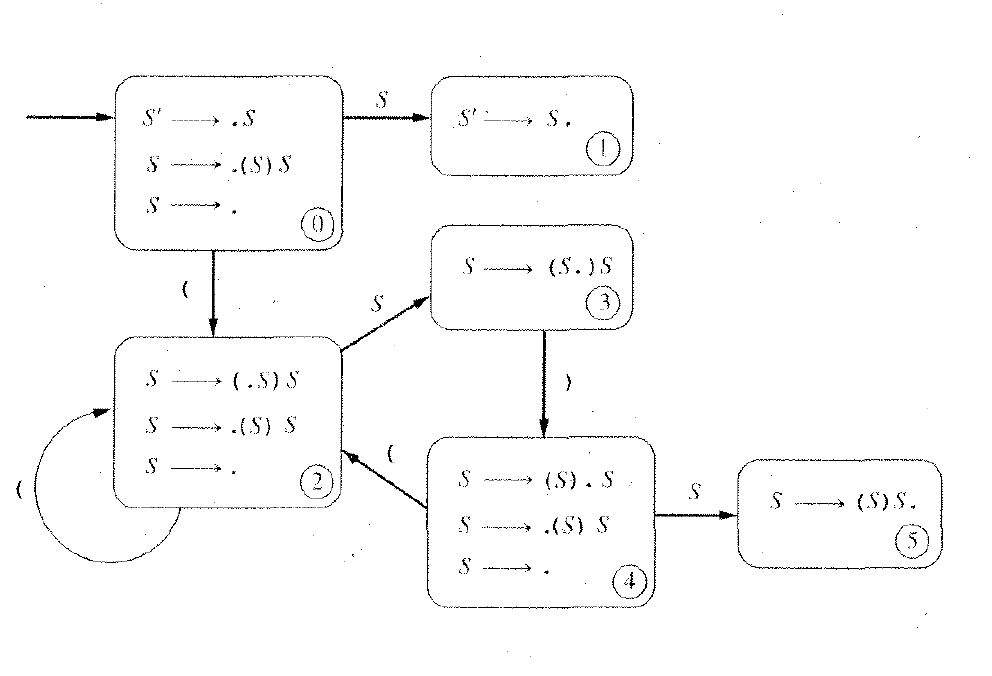
# Taller compiladores segundo parcial

1. Sea la gramática: S -> ( S ) S | e

* + AFD LR correspondientes a esta gramática.
  + Muestre las acciones de análisis sintáctico para la entrada: ( ( ( ) ) ( ( ) ( ) ) )



|  |  |  |  |
| --- | --- | --- | --- |
| Pasos | Pila | Cadena | Acciones |
| 1 | $ 0 | ( ( ( ) ) ( ( ) ( ) ))$ | Desplazar |
| 2 | $ 0(2 | ( ( ) ) ( ( ) ( ) ))$ | Desplazar |
| 3 | $ 0(2(2 | ( ) ) ( ( ) ( ) ))$ | Desplazar |
| 4 | $ 0(2(2(2 | ) ) ( ( ) ( ) ))$ | Reducir S🡪e |
| 5 | $ 0(2(2(2S3 | ) ) ( ( ) ( ) ))$ | Desplazar |
| 6 | $ 0(2(2(2S3)4 | ) ( ( ) ( ) ))$ | Reducir S🡪e |
| 7 | $ 0(2(2(2S3)4S5 | ) ( ( ) ( ) ))$ | Reducir S🡪(S)S |
| 8 | $ 0(2(2S3 | ) ( ( ) ( ) ))$ | Desplazar |
| 9 | $ 0(2(2S3)4 | ( ( ) ( ) ))$ | Desplazar |
| 10 | $ 0(2(2S3)4(2 | ( ) ( ) ))$ | Desplazar |
| 11 | $ 0(2(2S3)4(2(2 | ) ( ) ))$ | Reducir S🡪e |
| 12 | $ 0(2(2S3)4(2(2S3 | ) ( ) ))$ | Desplazar |
| 13 | $ 0(2(2S3)4(2(2S3)4 | ( ) ))$ | Desplazar |
| 14 | $ 0(2(2S3)4(2(2S3)4(2 | ) ))$ | Reducir S🡪e |
| 15 | $ 0(2(2S3)4(2(2S3)4(2S3 | ) ))$ | Desplazar |
| 16 | $ 0(2(2S3)4(2(2S3)4(2S3)4 | ))$ | Reducir S🡪e |
| 17 | $ 0(2(2S3)4(2(2S3)4(2S3)4S5 | ))$ | Reducir S🡪(S)S |
| 18 | $ 0(2(2S3)4(2(2S3)4S5 | ))$ | Reducir S🡪(S)S |
| 19 | $ 0(2(2S3)4(2S3 | ))$ | Desplazar |
| 20 | $ 0(2(2S3)4(2S3)4 | )$ | Reducir S🡪e |
| 21 | $ 0(2(2S3)4(2S3)4S5 | )$ | Reducir S🡪(S)S |
| 22 | $ 0(2(2S3)4S5 | )$ | Reducir S🡪(S)S |
| 23 | $ 0(2S3 | )$ | Desplazar |
| 24 | $ 0(2S3)4 | $ | Reducir S🡪e |
| 25 | $ 0(2S3)4S5 | $ | Reducir S🡪(S)S |
| 26 | $ 0S1 | $ | ACEPTAR |

1. Sea la gramática: E -> E+T | E-T | T T -> T\*F | T/F| F F-> (E) | **id** |num

* **Imagen que contiene texto, pizarrón

  Descripción generada automáticamente**Construya el AFD de elementos LR correspondiente.
* Muestre las acciones de análisis sintáctico para la entrada :

# id+id\*id+(id+id)/(id+id\*id)+id\*id+(id-id)

|  |  |  |  |
| --- | --- | --- | --- |
| Pasos | Pila | Cadena | Acciones |
| 1 | $ E | id+id\*id+(id+id)/(id+id\*id)+id\*id+(id-id)$ | Generar E🡪TE’ |
| 2 | $ TE’ | id+id\*id+(id+id)/(id+id\*id)+id\*id+(id-id)$ | Generar T🡪FT’ |
| 3 | $ FT’E’ | id+id\*id+(id+id)/(id+id\*id)+id\*id+(id-id)$ | Generar F🡪id |
| 4 | $idT’E’ | id+id\*id+(id+id)/(id+id\*id)+id\*id+(id-id)$ | Concordar |
| 5 | $ T’E’ | +id\*id+(id+id)/(id+id\*id)+id\*id+(id-id)$ | Generar T’🡪e |
| 6 | $ E’ | +id\*id+(id+id)/(id+id\*id)+id\*id+(id-id)$ | Generar E’🡪+TE’ |
| 7 | $+TE’ | +id\*id+(id+id)/(id+id\*id)+id\*id+(id-id)$ | Concordar |
| 8 | $ TE’ | id\*id+(id+id)/(id+id\*id)+id\*id+(id-id)$ | Generar T-> FT’ |
| 9 | $ FT’E’ | id\*id+(id+id)/(id+id\*id)+id\*id+(id-id)$ | Generar F🡪id |
| 10 | $id T’E’ | id\*id+(id+id)/(id+id\*id)+id\*id+(id-id)$ | Concordar |
| 11 | $ T’E’ | \*id+(id+id)/(id+id\*id)+id\*id+(id-id)$ | Generar T’-> \*FT’ |
| 12 | $\*FT’E’ | \*id+(id+id)/(id+id\*id)+id\*id+(id-id)$ | Concordar |
| 13 | $ FT’E’ | id+(id+id)/(id+id\*id)+id\*id+(id-id)$ | Generar F🡪id |
| 14 | $ id T’E’ | id+(id+id)/(id+id\*id)+id\*id+(id-id)$ | Concordar |
| 15 | $ T’E’ | +(id+id)/(id+id\*id)+id\*id+(id-id)$ | Generar T’🡪e |
| 16 | $ E’ | +(id+id)/(id+id\*id)+id\*id+(id-id)$ | Generar E’🡪+TE’ |
| 17 | $+TE’ | +(id+id)/(id+id\*id)+id\*id+(id-id)$ | Concordar |
| 18 | $ TE’ | (id+id)/(id+id\*id)+id\*id+(id-id)$ | Generar T🡪FT’ |
| 19 | $ FT’E’ | (id+id)/(id+id\*id)+id\*id+(id-id)$ | Generar F🡪(E) |
| 20 | $(E)T’E’ | (id+id)/(id+id\*id)+id\*id+(id-id)$ | Concordar |
| 21 | $ E)T’E’ | id+id)/(id+id\*id)+id\*id+(id-id)$ | Generar E🡪TE’ |
| 22 | $ TE’) T’E’ | id+id)/(id+id\*id)+id\*id+(id-id)$ | Generar T🡪FT’ |
| 23 | $ FT’E’) T’E’ | id+id)/(id+id\*id)+id\*id+(id-id)$ | Generar F🡪id |
| 24 | $ idT’E’) T’E’ | id+id)/(id+id\*id)+id\*id+(id-id)$ | Concordar |
| 25 | $ T’E’) T’E’ | +id)/(id+id\*id)+id\*id+(id-id)$ | Generar T’🡪e |
| 26 | $ E’) T’E’ | +id)/(id+id\*id)+id\*id+(id-id)$ | Generar E’🡪+TE’ |
| 27 | $ +TE’) T’E’ | +id)/(id+id\*id)+id\*id+(id-id)$ | Concordar |
| 28 | $ TE’) T’E’ | id)/(id+id\*id)+id\*id+(id-id)$ | Generar T🡪FT’ |
| 29 | $ FT’ E’) T’E’ | id)/(id+id\*id)+id\*id+(id-id)$ | Generar F🡪id |
| 30 | $ idT’ E’) T’E’ | id)/(id+id\*id)+id\*id+(id-id)$ | Concordar |
| 31 | $ T’ E’) T’E’ | )/(id+id\*id)+id\*id+(id-id)$ | Generar T’🡪e |
| 32 | $ E’) T’E’ | )/(id+id\*id)+id\*id+(id-id)$ | Generar E’🡪e |
| 33 | $ ) T’E’ | )/(id+id\*id)+id\*id+(id-id)$ | Concordar |
| 34 | $ T’E’ | /(id+id\*id)+id\*id+(id-id)$ | Generar T’-> /FT’ |
| 35 | $ /FT’E’ | /(id+id\*id)+id\*id+(id-id)$ | Concordar |
| 36 | $ FT’E’ | (id+id\*id)+id\*id+(id-id)$ | Generar F🡪(E) |
| 37 | $ (E) T’E’ | (id+id\*id)+id\*id+(id-id)$ | Concordar |
| 38 | $ E) T’E’ | id+id\*id)+id\*id+(id-id)$ | Generar E🡪TE’ |
| 39 | $ TE’) T’E’ | id+id\*id)+id\*id+(id-id)$ | Generar T🡪FT’ |
| 40 | $ FT’E’) T’E’ | id+id\*id)+id\*id+(id-id)$ | Generar F🡪id |
| 41 | $ idT’E’) T’E’ | id+id\*id)+id\*id+(id-id)$ | Concordar |
| 42 | $ T’E’) T’E’ | +id\*id)+id\*id+(id-id)$ | Generar T’🡪e |
| 43 | $ E’) T’E’ | +id\*id)+id\*id+(id-id)$ | Generar E’🡪+TE’ |
| 44 | $ +TE’) T’E’ | +id\*id)+id\*id+(id-id)$ | Concordar |
| 45 | $ TE’) T’E’ | id\*id)+id\*id+(id-id)$ | Generar T🡪FT’ |
| 46 | $ FT’ E’) T’E’ | id\*id)+id\*id+(id-id)$ | Generar F🡪id |
| 47 | $ idT’ E’) T’E’ | id\*id)+id\*id+(id-id)$ | Concordar |
| 48 | $ T’ E’) T’E’ | \*id)+id\*id+(id-id)$ | Generar T’-> \*FT’ |
| 49 | $ \*FT’ E’) T’E’ | \*id)+id\*id+(id-id)$ | Concordar |
| 50 | $ FT’ E’) T’E’ | id)+id\*id+(id-id)$ | Generar F🡪id |
| 51 | $ idT’ E’) T’E’ | )+id\*id+(id-id)$ | Concordar |
| 52 | $ T’ E’) T’E’ | )+id\*id+(id-id)$ | Generar T’🡪e |
| 53 | $ E’) T’E’ | )+id\*id+(id-id)$ | Generar E’🡪e |
| 54 | $ ) T’E’ | )+id\*id+(id-id)$ | Concordar |
| 55 | $ T’E’ | +id\*id+(id-id)$ | Generar T’🡪e |
| 56 | $ E’ | +id\*id+(id-id)$ | Generar E’🡪+TE’ |
| 57 | $ +TE’ | +id\*id+(id-id)$ | Concordar |
| 58 | $ TE’ | id\*id+(id-id)$ | Generar T🡪FT’ |
| 59 | $ FT’ E’ | id\*id+(id-id)$ | Generar F🡪id |
| 60 | $id T’ E’ | id\*id+(id-id)$ | Concordar |
| 61 | $ T’ E’ | \*id+(id-id)$ | Generar T’-> \*FT’ |
| 62 | $ \*FT’ E’ | \*id+(id-id)$ | Concordar |
| 63 | $ FT’ E’ | id+(id-id)$ | Generar F🡪id |
| 64 | $ id T’ E’ | id+(id-id)$ | Concordar |
| 65 | $ T’ E’ | +(id-id)$ | Generar T’🡪e |
| 66 | $ E’ | +(id-id)$ | Generar E’🡪+TE’ |
| 67 | $ +TE’ | +(id-id)$ | Concordar |
| 68 | $ TE’ | (id-id)$ | Generar T🡪FT’ |
| 69 | $ FT’ E’ | (id-id)$ | Generar F🡪(E) |
| 70 | $ (E) T’ E’ | (id-id)$ | Concordar |
| 71 | $ E) T’ E’ | id-id)$ | Generar E🡪TE’ |
| 72 | $ TE’) T’ E’ | id-id)$ | Generar T🡪FT’ |
| 73 | $ FT’ E’) T’ E’ | id-id)$ | Generar F🡪id |
| 74 | $ id T’ E’) T’ E’ | id-id)$ | Concordar |
| 75 | $ T’ E’) T’ E’ | -id)$ | Generar T’🡪e |
| 76 | $ E’) T’ E’ | -id)$ | Generar E’🡪-TE’ |
| 77 | $ -TE’) T’ E’ | -id)$ | Concordar |
| 78 | $ TE’) T’ E’ | id)$ | Generar T🡪FT’ |
| 79 | $ FT’E’) T’ E’ | id)$ | Generar F🡪id |
| 80 | $ id T’E’) T’ E’ | id)$ | Concordar |
| 81 | $ T’E’) T’ E’ | )$ | Generar T’🡪e |
| 82 | $ E’) T’ E’ | )$ | Generar E’🡪e |
| 83 | $ ) T’ E’ | )$ | Concordar |
| 84 | $ T’ E’ | $ | Generar T’🡪e |
| 85 | $ E’ | $ | Generar E’🡪e |
| 86 | $ | $ | ACEPTAR |

1. Considerar La siguiente Gramática:

lexp -> atom | list

atom -> **numero** | **identificador**

list -> ( lexp-seq)

lexp-seq -> lexp lexp –seq’

lexp-seq’ -> lexp lexp-seq’| ε

* Muestre las acciones del analizador sintáctico descendente para la cadena de entrada: (a (b (2)) (c (4)))

|  |  |  |  |
| --- | --- | --- | --- |
| Pasos | Pila | Cadena | Acciones |
| 1 | $ lexp | (a (b (2)) (c (4))) $ | Generar lexp 🡪list |
| 2 | $ list | (a (b (2)) (c (4))) $ | Generar list🡪 ( lexp-seq) |
| 3 | $ ( lexp-seq) | (a (b (2)) (c (4))) $ | Concordar |
| 4 | $ lexp-seq) | a (b (2)) (c (4))) $ | Generar lexp-seq🡪lexp lexp-seq’ |
| 5 | $ lexp lexp-seq’) | a (b (2)) (c (4))) $ | Generar lexp 🡪 atom |
| 6 | $ atom lexp-seq’) | a (b (2)) (c (4))) $ | Generar atom 🡪 **identificador** |
| 7 | $ identificador lexp-seq’) | a (b (2)) (c (4))) $ | Concordar |
| 8 | $ lexp-seq’) | (b (2)) (c (4))) $ | Generar lexp-seq’🡪lexp lexp-seq’ |
| 9 | $ lexp lexp-seq’) | (b (2)) (c (4))) $ | Generar lexp 🡪list |
| 10 | $ list lexp-seq’) | (b (2)) (c (4))) $ | Generar list🡪 ( lexp-seq) |
| 11 | $ ( lexp-seq) lexp-seq’) | (b (2)) (c (4))) $ | Concordar |
| 12 | $ lexp-seq) lexp-seq’) | b (2)) (c (4))) $ | Generar lexp-seq’🡪lexp lexp-seq’ |
| 13 | $ lexp lexp-seq’) lexp-seq’) | b (2)) (c (4))) $ | Generar lexp 🡪 atom |
| 14 | $ atom lexp-seq’) lexp-seq’) | b (2)) (c (4))) $ | Generar atom 🡪 **identificador** |
| 15 | $ identificador lexp-seq’) lexp-seq’) | b (2)) (c (4))) $ | Concordar |
| 16 | $ lexp-seq’) lexp-seq’) | (2)) (c (4))) $ | Generar lexp-seq’🡪lexp lexp-seq’ |
| 17 | $ lexp lexp-seq’) lexp-seq’) | (2)) (c (4))) $ | Generar lexp 🡪list |
| 18 | $ list lexp-seq’) lexp-seq’) | (2)) (c (4))) $ | Generar list🡪 ( lexp-seq) |
| 19 | $ (lexp-seq) lexp-seq’) lexp-seq’) | (2)) (c (4))) $ | Concordar |
| 20 | $ lexp-seq) lexp-seq’) lexp-seq’) | 2)) (c (4))) $ | Generar lexp-seq🡪lexp lexp-seq’ |
| 21 | $ lexp lexp-seq’) lexp-seq’) lexp-seq’) | 2)) (c (4))) $ | Generar lexp 🡪 atom |
| 22 | $ atom lexp-seq’) lexp-seq’) lexp-seq’) | 2)) (c (4))) $ | Generar atom 🡪 **numero** |
| 23 | $ num lexp-seq’) lexp-seq’) lexp-seq’) | 2)) (c (4))) $ | Concordar |
| 24 | $ lexp-seq’) lexp-seq’) lexp-seq’) | )) (c (4))) $ | Generar lexp-seq’🡪 ε |
| 25 | $ ) lexp-seq’) lexp-seq’) | )) (c (4))) $ | Concordar |
| 26 | $ lexp-seq’) lexp-seq’) | ) (c (4))) $ | Generar lexp-seq’🡪 ε |
| 27 | $ ) lexp-seq’) | ) (c (4))) $ | Concordar |
| 28 | $ lexp-seq’) | (c (4))) $ | Generar lexp-seq’🡪lexp lexp-seq’ |
| 29 | $ lexp lexp-seq’) | (c (4))) $ | Generar lexp 🡪list |
| 30 | $ list lexp-seq’) | (c (4))) $ | Generar list🡪 ( lexp-seq) |
| 31 | $ ( lexp-seq) lexp-seq’) | (c (4))) $ | Concordar |
| 32 | $ lexp-seq) lexp-seq’) | c (4))) $ | Generar lexp-seq🡪lexp lexp-seq’ |
| 33 | $ lexp lexp-seq’) lexp-seq’) | c (4))) $ | Generar lexp 🡪 atom |
| 34 | $ atom lexp-seq’) lexp-seq’) | c (4))) $ | Generar atom 🡪 **identificador** |
| 35 | $ identificador lexp-seq’) lexp-seq’) | c (4))) $ | Concordar |
| 36 | $ lexp-seq’) lexp-seq’) | (4))) $ | Generar lexp-seq’🡪lexp lexp-seq’ |
| 37 | $ lexp lexp-seq’) lexp-seq’) | (4))) $ | Generar lexp 🡪list |
| 38 | $ list lexp-seq’) lexp-seq’) | (4))) $ | Generar list🡪 ( lexp-seq) |
| 39 | $ ( lexp-seq) lexp-seq’) lexp-seq’) | (4))) $ | Concordar |
| 40 | $ lexp-seq) lexp-seq’) lexp-seq’) | 4))) $ | Generar lexp-seq🡪lexp lexp-seq’ |
| 41 | $ lexp lexp-seq) lexp-seq’) lexp-seq’) | 4))) $ | Generar lexp 🡪 atom |
| 42 | $ atom lexp-seq’) lexp-seq’) lexp-seq’) | 4))) $ | Generar atom 🡪 **numero** |
| 43 | $ num lexp-seq’) lexp-seq’) lexp-seq’) | 4))) $ | Concordar |
| 44 | $ lexp-seq’) lexp-seq’) lexp-seq’) | ))) $ | Generar lexp-seq’🡪 ε |
| 45 | $ ) lexp-seq’) lexp-seq’) | ))) $ | Concordar |
| 46 | $ lexp-seq’) lexp-seq’) | )) $ | Generar lexp-seq’🡪 ε |
| 47 | $ ) lexp-seq’) | )) $ | Concordar |
| 48 | $ lexp-seq’) | ) $ | Generar lexp-seq’🡪 ε |
| 49 | $ ) | ) $ | Concordar |
| 50 | $ | $ | ACEPTAR |

1. Considerar La siguiente Gramática: E -> E+T | E-T | T

T -> T\*F | T/F| F F-> (E) | **id** | **nu**

* Construya los conjuntos primeros y siguientes.
* Construya la tabla de análisis sintáctico
* Muestre las acciones del analizador sintáctico descendente para la cadena de entrada: 3-5/(3+4\*3)+6+89+(2+4)
* La gramática resultante es LL, justifique su respuesta

Tabla de análisis sintáctico:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | + | - | \* | ( | ) | num | Id | $ |
| E |  |  |  | TE’ |  | TE’ | TE’ |  |
| E’ | +TE | -TE’ |  |  | e |  |  | e |
| T |  |  |  |  |  | FT’ | FT’ |  |
| T’ | E | e | \*FT’ | /FT’ | E |  |  | e |
| F |  |  |  |  |  | num | Id |  |

Analizador sintáctico descendente:

|  |  |  |  |
| --- | --- | --- | --- |
| Pasos | Pila | Cadena | Acciones |
| 1 | $E | 3-5/(3+4\*3)+6+89+(2+4)$ | Generar E->TE’ |
| 2 | $TE’ | 3-5/(3+4\*3)+6+89+(2+4)$ | Generar T -> FT’ |
| 3 | $FT’E’ | 3-5/(3+4\*3)+6+89+(2+4)$ | Generar F-> Num |
| 4 | $NumT’E’ | 3-5/(3+4\*3)+6+89+(2+4)$ | Concordar |
| 5 | $T’E’ | -5/(3+4\*3)+6+89+(2+4)$ | Generar T’->e |
| 6 | $E’ | -5/(3+4\*3)+6+89+(2+4)$ | Generar E’->-TE’ |
| 7 | $-TE’ | -5/(3+4\*3)+6+89+(2+4)$ | Concordar |
| 8 | $TE’ | 5/(3+4\*3)+6+89+(2+4)$ | Generar T-> FT’ |
| 9 | $FT’E’ | 5/(3+4\*3)+6+89+(2+4)$ | Generar F->Num |
| 10 | $NumT’E’ | 5/(3+4\*3)+6+89+(2+4)$ | Concordar |
| 11 | $T’E’ | /(3+4\*3)+6+89+(2+4)$ | Generar T’-> /FT’ |
| 12 | $/FT’E’ | /(3+4\*3)+6+89+(2+4)$ | Concordar |
| 13 | $FT’E’ | (3+4\*3)+6+89+(2+4)$ | Generar T-> (E) |
| 14 | $(E)T’E’ | (3+4\*3)+6+89+(2+4)$ | Concordar |
| 15 | $E)T’E’ | 3+4\*3)+6+89+(2+4)$ | Generar E-> TE’ |
| 16 | $TE’)T’E’ | 3+4\*3)+6+89+(2+4)$ | Generar T->FT’ |
| 17 | $FT’E’)T’E’ | 3+4\*3)+6+89+(2+4)$ | Generar F-> Num |
| 18 | $NumT’E’)T’E’ | 3+4\*3)+6+89+(2+4)$ | Concordar |
| 19 | $T’E’)T’E’ | +4\*3)+6+89+(2+4)$ | Generar T’->e |
| 20 | $E’)T’E’ | +4\*3)+6+89+(2+4)$ | Generar E’-> +TE’ |
| 21 | $+TE’)T’E’ | +4\*3)+6+89+(2+4)$ | Concordar |
| 22 | $TE’)T’E’ | 4\*3)+6+89+(2+4)$ | Generar T->FT’ |
| 23 | $FT’E’)T’E’ | 4\*3)+6+89+(2+4)$ | Generar F->Num |
| 24 | $NumT’E’)T’E’ | 4\*3)+6+89+(2+4)$ | Concordar |
| 25 | $T’E’)T’E’ | \*3)+6+89+(2+4)$ | Generar T’->\*FT’ |
| 26 | $\*FT’E’)T’E’ | \*3)+6+89+(2+4)$ | Concordar |
| 27 | $FT’E’)T’E’ | 3)+6+89+(2+4)$ | Generar F-> Num |
| 28 | $NumT’E’)T’E’ | 3)+6+89+(2+4)$ | Concordar |
| 29 | $T’E’)T’E’ | )+6+89+(2+4)$ | Generar T’->e |
| 30 | $E’)T’E’ | )+6+89+(2+4)$ | Generar E’->e |
| 31 | $)T’E’ | )+6+89+(2+4)$ | Concordar |
| 32 | $T’E’ | +6+89+(2+4)$ | Generar T’-> e |
| 33 | $E’ | +6+89+(2+4)$ | Generar E’-> +TE’ |
| 34 | $+TE’ | +6+89+(2+4)$ | Concordar |
| 35 | $TE’ | 6+89+(2+4)$ | Generar T->FT’ |
| 36 | $FT’E’ | 6+89+(2+4)$ | Generar F-> Num |
| 37 | $NumT’E’ | 6+89+(2+4)$ | Concordar |
| 38 | $T’E’ | +89+(2+4)$ | Generar T’-> e |
| 39 | $E’ | +89+(2+4)$ | Generar E’-> +TE’ |
| 40 | $+TE’ | +89+(2+4)$ | Concordar |
| 41 | $TE’ | 89+(2+4)$ | Generar T->F |
| 42 | $FE’ | 89+(2+4)$ | Genera F-> Num |
| 43 | $NumE’ | 89+(2+4)$ | Concordar |
| 44 | $E’ | +(2+4)$ | Generar E’->+TE’ |
| 45 | $+TE’ | +(2+4)$ | Concordar |
| 46 | $TE’ | (2+4)$ | Generar T->FT’ |
| 47 | $FT’E’ | (2+4)$ | Generar F-> (E) |
| 48 | $(E)T’E’ | (2+4)$ | Concordar |
| 49 | $E)T’E’ | 2+4)$ | Generar E->TE’ |
| 50 | $TE’)E’ | 2+4)$ | Generar T-> FT’ |
| 51 | $FT’E’)E’ | 2+4)$ | Generar F->Num |
| 52 | $NumT’E’)E’ | 2+4)$ | Concordar |
| 53 | $T’E’)E’ | +4)$ | Generar T’->e |
| 54 | $E’)E’ | +4)$ | Generar E’-> +TE’ |
| 55 | $+TE’)E’ | +4)$ | Concordar |
| 56 | $TE’)E’ | 4)$ | Generar T-> FT’ |
| 57 | $FT’E’)E’ | 4)$ | Generar F->Num |
| 58 | $NumT’E’)E’ | 4)$ | Concordar |
| 59 | $T’E’)E’ | )$ | Generar T’-> e |
| 60 | $E’)E’ | )$ | Generar E’->e |
| 61 | $)E’ | )$ | Concordar |
| 62 | $E’ | $ | Generar E’->e |
| 63 | $ | $ | Aceptar |

La gramática resultante es LL porque tiene como máximo una producción en cada entrada

1. Dada la gramática: S I | otro

I  **if** S | **if** S **else** S

Construir el autómata finito de elementos LR

Imagen que contiene blanco, con baldosas, azulejo, parado

Descripción generada automáticamente