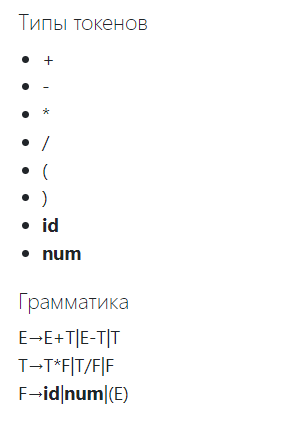
Условие лабораторной:



Начало работы:

1) Добавляем стартовый символ: S=E', теперь грамматика выглядит так:

E'-->E

E-->E+T|E-T|T

T-->T\*F|T/F|F

E-->id|num|(E)

2)Строим все возможные замыкания:

I0 = CLOSURE({E'->.E}) = {E'->.E, E->.E+T, E->.E-T, E->.T, T->.T\*F, T->.T/F, T->.F, F->.id, F->.num, >.(E)}

Точка перед терминальным символом означает, что мы считали все символы до него, значит настала очередь прочитать и его (рассматриваем все правила, которые могут быть у него).

3)Строим все возможные переходы GOTO. Например GOTO(I0, num) означает переходы из состояния I0 по символу num.

* I1 = GOTO(I0,id) = CLOSURE({F-->id.}) = {F-->id.}

Если точка стоит после терминального символа и далее нет нетерминального символа либо точка стоит после терминального символа и далее есть терминальный, то это значит, что мы рассмотрели все варианты.

* I2 = GOTO(I0,num) = CLOSURE({F-->num.}) = {F-->num.}
* I3 = GOTO(I0,"(") = CLOSURE({F-->(.E)}) = {F-->(.E), E-->.E+T, E-->.E-T, E-->.T, T-->.T\*F, T-->.T/F, T-->.F, F-->.id, F-->.num, F-->.(E)}
* I4 = GOTO(I0,E) = CLOSURE({E'-->E., E-->E.+T, E-->E.-T}) = {E'-->E., E-->E.+T, E-->E.-T}
* I5 = GOTO(I0,T) = CLOSURE({E-->T., T-->T.\*F, T-->T./F}) = {E-->T., T-->T.\*F, T-->T./F}
* I6 = GOTO(I0,F) = CLOSURE({T-->F.}) = {T-->F.}
* I1 = GOTO(I3,id) = CLOSURE({F-->id.}) = {F-->id.}
* I2 = GOTO(I3,num) = CLOSURE({F-->num.}) = {F-->num.}
* I3 = GOTO(I0,"(") = CLOSURE({F-->(.E)}) = {F-->(.E)}
* I7 = GOTO(I3,E) = CLOSURE({F-->(E.), E-->E.+T, E-->E.-T}) = {F-->(E.), E-->E.+T, E-->E.-T}
* I5 = GOTO(I3,T) = CLOSURE({E-->T., T-->T.\*F, T-->T./F}) = {E-->T., T-->T.\*F, T-->T./F}
* I6 = GOTO(I3,F) = CLOSURE({T-->F.}) = {T-->F.}
* I8 = GOTO(I4,+) = CLOSURE({E-->E+.T}) = {E-->E+.T, T-->.T\*F, T-->.T/F, T-->.F, F-->.id, F-->.num, F-->.(E)}
* I9 = GOTO(I4,-) = CLOSURE({E-->E-.T}) = {E-->E-.T, T-->.T\*F, T-->.T/F, T-->.F, F-->.id, F-->.num, F-->.(E)}
* I10 = GOTO(I5,\*) = CLOSURE({T-->T\*.F}) = {T-->T\*.F, F-->.id, F-->.num, F-->.(E)}
* I11 = GOTO(I5,/) = CLOSURE({T-->T/.F}) = {T-->T/.F, F-->.id, F-->.num, F-->.(E)}
* I8 = GOTO(I7,+) = CLOSURE({E-->E+.T})
* I9 = GOTO(I7,-) = CLOSURE({E-->E-.T})
* I12 = GOTO(I7,")") = CLOSURE({F-->(E).}) = {F-->(E).}
* I1 = GOTO(I8,id) = CLOSURE({F-->id.})
* I2 = GOTO(I8,num) = CLOSURE({F-->num.})
* I3 = GOTO(I8,"(") = CLOSURE({F-->(.E)})
* I6 = GOTO(I8,F) = CLOSURE({T-->F.})
* I13 = GOTO(I8,T) = CLOSURE({T-->T./F, E-->E+T., T-->T.\*F})
* I1 = GOTO(I9,id) = CLOSURE({F-->id.})
* I2 = GOTO(I9,num) = CLOSURE({F-->num.})
* I3 = GOTO(I9,"(") = CLOSURE({F-->(.E)})
* I14 = GOTO(I9,T) = CLOSURE({E-->E-T., T-->T.\*F, T-->T./F}) = {E-->E-T., T-->T.\*F, T-->T./F}
* I6 = GOTO(I9,F) = CLOSURE({T-->F.})
* I1 = GOTO(I10,id) = CLOSURE({F-->id.})
* I2 = GOTO(I10,num) = CLOSURE({F-->num.})
* I3 = GOTO(I10,"(") = CLOSURE({F-->(.E)})
* I15 = GOTO(I10,F) = CLOSURE({T-->T\*F.}) = {T-->T\*F.}
* I1 = GOTO(I11,id) = CLOSURE({F-->id.})
* I2 = GOTO(I11,num) = CLOSURE({F-->num.})
* I3 = GOTO(I11,"(") = CLOSURE({F-->(.E)})
* I16 = GOTO(I11,F) = CLOSURE({T-->T/F.})
* I11 = GOTO(I13,/) = CLOSURE({T-->T/.F})
* I10 = GOTO(I13,\*) = CLOSURE({T-->T\*.F})
* I10 = GOTO(I14,\*) = CLOSURE({T-->T\*.F})
* I11 = GOTO(I14,/) = CLOSURE({T-->T/.F})

4)Составляем таблицу FIRST и FOLLOW

|  |  |  |
| --- | --- | --- |
|  | FIRST | FOLLOW |
| E | id, num, ( | +, -, ), $ |
| T | id, num, ( | \*, /, $, +, - |
| F | Id, num, ( | +, -, ), $, \*, / |
| E’ | Id, num, ( | $ |

5) Нумеруем правила грамматики:

E’ -> E (0)

E -> E+T (1) | E-T (2) | T (3)

T -> T\*F (4) | T/F (5) | F (6)

F -> id (7) | num (8) | (E) (9)

6) Строим таблицу ACTION GOTO

В таблице ACTION идет работа со всеми терминальными символами. Рассматриваем замыкания, например, у нас записывалось GOTO(I0, id) и это было наше состояние I1, поэтому мы в таблице в I0 по id записываем s1. Таким образом заполняем нашу таблицу.

Далее, мы рассматриваем наши полные замыкания, например, у нас записывалось замыкание {F🡪id.}, где точка стоит уже в самом конце. Мы смотрим, какое у нас правило по этому замыканию, в данном примере – это правило (7). Далее переходим в таблицу FOLLOW по символу F и все символы, которые есть у нас в таблице FOLLOW заполняются правилами r7 в таблице ACTION. Таким образом заполняем нашу таблицу.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | ACTION | | | | | | | | | GOTO | | |
| + | - | \* | / | ( | ) | $ | id | num | E | T | F |
| I0 | ------- | ------- | ------- | ------- | S3 | ------- | ------- | S1 | S2 | I4 | I5 | I6 |
| I1 | R7 | R7 | R7 | R7 | ------- | R7 | R7 | ------- | ------- | ------- | ------- | ------- |
| I2 | R8 | R8 | R8 | R8 | ------- | R8 | R8 | ------- | ------- | ------- | ------- | ------- |
| I3 | ------- | ------- | ------- | ------- | S3 | ------- | ------- | S1 | S2 | I7 | I5 | I6 |
| I4 | S8 | S9 | ------- | ------- | ------- | ------- | R0 | ------- | ------- | ------- | ------- | ------- |
| I5 | R3 | R3 | S10 | S11 | ------- | R3 | R3 | ------- | ------- | ------- | ------- | ------- |
| I6 | R6 | R6 | R6 | R6 | ------- | R6 | R6 | ------- | ------- | ------- | ------- | ------- |
| I7 | S8 | S9 | ------- | ------- | ------- | S12 | ------- | ------- | ------- | ------- | ------- | ------- |
| I8 | ------- | ------- | ------- | ------- | S3 | ------- | ------- | S1 | S2 | ------- | I13 | I6 |
| I9 | ------- | ------- | ------- | ------- | S3 | ------- | ------- | S1 | S2 | ------- | I14 | I6 |
| I10 | ------- | ------- | ------- | ------- | S3 | ------- | ------- | S1 | S2 | ------- | ------- | I15 |
| I11 | ------- | ------- | ------- | ------- | S3 | ------- | ------- | S1 | S2 | ------- | ------- | I16 |
| I12 | R9 | R9 | R9 | R9 | ------- | ------- | R9 | R9 | ------- | ------- | ------- | ------- |
| I13 | R1 | R1 | S10 | S11 | ------- | R1 | R1 | ------- | ------- | ------- | ------- | ------- |
| I14 | R2 | R2 | S10 | S11 | ------- | R2 | R2 | ------- | ------- | ------- | ------- | ------- |
| I15 | R4 | R4 | R4 | R4 | ------- | R4 | R4 | ------- | ------- | ------- | ------- | ------- |
| I16 | R5 | R5 | R5 | R5 | ------- | R5 | R5 | ------- | ------- | ------- | ------- | ------- |

Где в таблице:

Si – выполнить сдвиг (считать символ) и перейти в состояние i

Ri – выполнить сверстку по правилу i (получается, что когда выполнится R0, программа завершится успешно). Затем смотрим, какое правило применили, предыдущее состояние и на пересечении состояния и правила в таблице GOTO смотрим, в какое состояние перейти.

Пример: анализ выражения id+num\*id$

Нам понадобится стек, в который мы будем заносить состояния. Алгоритм рассмотрен по шагам:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Текущее состояние | Считано | Осталось | Стек | Действие |
| I0 | - | id+num\*id$ | 0 | S1 (считываем id и переходим в состояние I1) |
| I1 | id | +num\*id$ | 0,1 | R7 (сверстка по правилу 7 (F -> id), извлекаем из стека 1, GOTO из 0 по F = 6, то есть переходим в состояние I6) |
| I6 | F | +num\*id$ | 0,6 | R6 |
| I5 | T | +num\*id$ | 0,5 | R3 |
| I4 | E | +num\*id$ | 0,4 | S8 |
| I8 | E+ | Num\*id$ | 0,4,8 | S2 |
| I2 | E+num | \*id$ | 0,4,8,2 | R8 |
| I6 | E+F | \*id$ | 0,4,8,6 | R6 |
| I13 | E+T | \*id$ | 0,4,8,13 | S10 |
| I10 | E+T\* | Id$ | 0,4,8,13,10 | S1 |
| I1 | E+T\*id | $ | 0,4,8,13,10,1 | R7 |
| I15 | E+T\*F | $ | 0,4,8,13,10,15 | R4 |
| I13 | E+T | $ | 0,4,8,13 | R1 |
| I4 | E | $ | 0,4 | R0 |
|  | E’ | Получили E’ => строка поддерживается грамматикой | | |

Пояснения к коду в самой программе в виде комментариев