Дано: **G** – КС-грамматика

1: *E* → *E*+*T*

2: E → *E-T*

3: *E* → *T*

4: *T* → *a*

5: *T* → *i*

6: *T* → (*E*)

7: *T* → *i*(*E*)

Аксиома **–** символ*Е*;

N = {*E*, *T*}

∑ = {+, -, *a*, *i*, (, ) }

А где варианты я не понял, вот.

1. Все символы полезны и достижимы, требуется устранить непосредственную левую рекурсию. Левая факторизация не требуется. Эквивалентная грамматика:

1: E → *TR*

2: R → *λ*

3: R → +*TR*

4: R → *-TR*

5: T → *a*

6: T → *i*

7: T → *(E)*

8: T → *i(E)*

2) Рекуррентная система соотношений для вычисления функции FIRST1(X);

Система определяющих уравнений:

Запишем рекуррентную систему равенств:

*Ei+1 = T i k Ri | Ei*

*Ri+1 = λ | + k T i k Ri | - k Ti k Ri | Ri*

*T i+1 = a | i | (k Ei k)| ik(k Ei k) | Ti*

*E0 = R0 = T0 = ∅*

3) Значения FIRST1(X) для всех нетерминалов

|  |  |  |  |
| --- | --- | --- | --- |
| i | *E i* | *R i* | *T i* |
| 0 | *∅* | *∅* | *∅* |
| 1 | *∅* | *λ* | *a | i* |
| 2 | *a | i* | *λ | + | -* | *a | i* |
| 3 | *a | i* | *λ | + | -* | *a | i | ( | i(* |
| 4 | *a | i | ( | i(* | *λ | + | -* | *a | i | ( | i(* |
| 5 | *a | i | ( | i(* | *λ | + | -* | *a | i | ( | i(* |

Предел последовательности определяется:

*(FIRST1(E), FIRST1(R), FIRST1(T))* = *(a | i | ( | i(, λ|+|-, a | i | ( | i()*

*E = FIRST1(E) = a | i | ( | i(*

*R = FIRST1(R) = λ|+|-*

*T = FIRST1(T) = a | i | ( | i(*

1. Значения *FOLLOW1(X)* для всех нетерминалов

*FOLLOW1(E) = λ|)*

*FOLLOW1(R) = λ|)*

*FOLLOW1(T) = λ|+|-|)*

1. Проверить выполнимость характеристического свойства SLL(1)-грамматик

Для этого вычислим для каждого нетерминала A N и его альтернативы p: A → γ выражение *qp = FIRSTk (γFOLLOW1 (A))* – управляющее множество аванцепочек альтернативы A → γ:

1: E → q1, *TR*  q1  = *a | i | ( | i(*

2: R → q2, *λ* q2  = *λ|)*

3: R → q3, *+TR* q3  = *+*

4: R → q4, *-TR* q4  = *-*

5: T → q5, *a* q5  = *a*

6: T → q6, *i* q6  = *i*

7: T → q7, *(E)* q7  = *(*

8: T → q8, *i(E)* q8  = *i(*

Грамматика является SLL(1)-грамматикой и, следовательно, LL(1)-грамматикой.

6) Построить управляющую таблицу SLL(1)-анализатора, если грамматика является SLL(1)-грамматикой;

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | a | i | ( | + | - | ) | *λ* |
| E | 1 | 1 | 1 |  |  |  |  |
| R |  |  |  | 3 | 4 | 2 | 2 |
| T | 5 | 6 | 7 |  |  |  |  |
| a | Сдвиг |  |  |  |  |  |  |
| i |  | Сдвиг |  |  |  |  |  |
| ( |  |  | Сдвиг |  |  |  |  |
| + |  |  |  | Сдвиг |  |  |  |
| - |  |  |  |  | Сдвиг |  |  |
| ) |  |  |  |  |  | Сдвиг |  |
| *λ* |  |  |  |  |  |  | Допуск |

7) Применить SLL(1)-анализатор к цепочке x  L(G); предъявить последовательность конфигураций, приводящей к допуску; проиллюстрировать корректность левого вывода, построив синтаксическое дерево и приписав номера правил внутренним узлам дерева

x = a + (a – i)

C0 = (a + (a-i), E, *λ*) 1

C1 = (a + (a-i), TR, 1) 5

C2 = (a + (a-i), aR, 15) сдвиг

C3 = (a+ (a-i), R, 15) 3

C4 = (a+ (a-i), +TR, 153) сдвиг

C5 = (a+ (a-i), TR, 153) 7

C6 = (a+ (a-i), (E)R, 1537) сдвиг

C7 = (a+ ( a-i), E)R, 1537) 1

C8 = (a+ ( a-i), TR)R, 15371) 5

C9 = (a+ ( a-i), aR)R, 153715) сдвиг

C10 = (a+ (a-i), R)R, 153715) 4

C11 = (a+ (a-i), -TR)R, 1537154) сдвиг

C12 = (a+ (a-i), TR)R, 1537154) 6

C13 = (a+ (a-i), iR)R, 15371546) сдвиг

C14 = (a+ (a-i), R)R, 15371546) 2

C15 = (a+ (a-i), )R, 153715462) сдвиг

C16 = (a+ (a-i), R, 153715462) 2

C17 = (a+ (a-i), , 1537154622) допуск

Тогда слово x = a + (a-i) L(G), левый вывод: 1537154622, синтаксическое дерево:

E

1

T

5

*a*

R

3

+

R

2

λ

T

7

E

1

R

4

(

)

T

5

*a*

-

R

2

λ

T

6

*i*