

**Лабораторные работы по курсу  
«Системное программное обеспечение»**

1. Спроектировать грамматику по заданному языку  $L$ .
2. Спроектировать конечный автомат, составить диаграмму переходов КА и реализовать.
3. Определить свойства КА. Построить НДКА. Реализовать преобразование НДКА в ДКА.
4. Устранить из КС-грамматики бесполезные символы и  $\epsilon$ -правила.
5. Устранить из КС-грамматики цепные правила и устранить левую рекурсию.
6. Определить форму КС-грамматики и сделать ее приведение.
7. Спроектировать МП автомат для приведенной КС-грамматики.
8. Реализовать МП автомат для приведенной КС-грамматики.
9. Для LL(1) анализатора построить управляющую таблицу  $M$ .
10. Аналитически написать такты работы LL(1) анализатора для выведенной цепочки.
11. Реализовать управляющую таблицу  $M$  для LL(1) анализатора.
12. Построить замыкание множества ситуаций для пополненной LR(1) грамматики.
13. Определить функцию перехода  $g(x)$ .
14. Построить каноническую форму множества ситуаций.
15. Построить управляющую таблицу для функции перехода  $g(x)$  и действий  $f(u)$ .
16. Реализовать LR(1)-анализатор по управляющей таблице  $(g, f)$  для LR(1) грамматики.

Студент: Бирюков В. В.  
Группа: 08-207

Руководитель: Семёнов А. С.

Оценка:  
Дата:

## Вариант 1

### Лабораторная работа №1

#### Формулировка задания

Составить примеры цепочек (1) и спроектировать грамматику заданного языка (2)

$$L = \{0\omega_1+(01)^* \mid \omega_1 \in \{0,1\}^*\}$$

$$(1) L = \{0+01, 00+0101, 01+, 000+010101, 001+0101, \dots\}$$

$$(2) L_1 = \{0+01, 00+0101, 01+, 000+010101, 001+0101\}$$

$$L_1 \equiv L_1(G) \subset L$$

#### Грамматика

$G = (T, V, S_0, P)$ , где  $T$  – конечное множество терминальных символов,  $V$  – конечное множество нетерминальных символов,  $S_0$  – начальное состояние,  $P$  – множество правил.

$P$ :

1.  $S_0 \rightarrow 0A$
2.  $A \rightarrow 0A$
3.  $A \rightarrow 1A$
4.  $A \rightarrow +B$
5.  $A \rightarrow +$
6.  $B \rightarrow 0C$
7.  $C \rightarrow 1B$
8.  $C \rightarrow 1$

Таким образом, грамматика имеет вид:

$$G = (\{0,1,+\}, \{A,B,C\}, 0A, P)$$

#### Вывод цепочек

$$S_0 \Rightarrow 0A \Rightarrow 0+B \Rightarrow 0+0C \Rightarrow 0+01$$

$$S_0 \Rightarrow 0A \Rightarrow 00A \Rightarrow 00+B \Rightarrow 00+0C \Rightarrow 00+01B \Rightarrow 00+010C \Rightarrow 00+0101$$

$$S_0 \Rightarrow 0A \Rightarrow 01A \Rightarrow 01+$$

$$S_0 \Rightarrow 0A \Rightarrow 00A \Rightarrow 000A \Rightarrow 000+B \Rightarrow 000+0C \Rightarrow 000+01B \Rightarrow 000+010C \Rightarrow 000+0101B \Rightarrow 000+01010C \Rightarrow 000+010101$$

$$S_0 \Rightarrow 0A \Rightarrow 00A \Rightarrow 001A \Rightarrow 001+B \Rightarrow 001+0C \Rightarrow 001+01B \Rightarrow 001+010C \Rightarrow 001+0101$$

#### Регулярность языка

$$(1) \text{ Для регулярного языка } L = \{0\omega_1+(01)^* \mid \omega_1 \in \{0,1\}^*\} \Rightarrow$$

$$(2) \text{ существует целое } (\exists p=2 \geq 1 \text{ такое что}$$

$$(3) \text{ для всех } (\forall w=01+01 \in L(|w| \geq p) \Rightarrow$$

$$(4) \text{ существует } (\exists x=0, y=1, z=+01 \in \Sigma^* \text{ такое что } (w=xyz \Rightarrow$$

1.  $(|y|=|1|\geq 1)$  , цикл у должен быть накачан хотя бы длиной 1 и
2.  $|xy|=|01|\leq p=2$  , цикл должен быть в пределах первых p символов и
3. для всех  $i=1\geq 0, (xy^1z=01+01\in L, )))))))$  , на x и z ограничений не накладывается.

## Лабораторная работа №2

### Формулировка задания

Составить на основе разработанной грамматики конечный автомат, распознающий эквивалентный ей язык.

$$L(KA) \equiv L(G)$$

### Конечный автомат

KA = (Q,  $\Sigma$ ,  $\delta$ ,  $q_0$ , F), где Q – конечное множество состояний,  $\Sigma$  – конечный алфавит входных символов,  $\delta$  – функция перехода, задаваемая отображением  $\delta: Q \times \Sigma \rightarrow Q$ ,  $q_0$  – начальное состояние автомата, F – множество заключительных состояний.

$\delta$ :

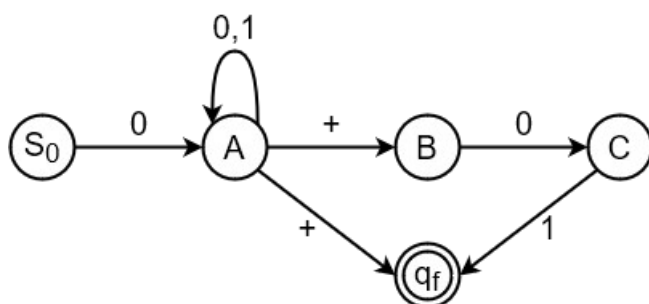
1.  $\delta(S_0, 0) = \{A\}$
2.  $\delta(A, 0) = \{A\}$
3.  $\delta(A, 1) = \{A\}$
4.  $\delta(A, +) = \{B, q_f\}$
5.  $\delta(B, 0) = \{C\}$
6.  $\delta(C, 1) = \{B, q_f\}$

Таким образом конечный автомат имеет вид:

$$KA = (\{S_0, A, B, C, q_f\}, \{0, 1, +\}, \delta, S_0, \{q_f\})$$

KA является недетерминированным.

### Диаграмма переходов конечного автомата



### Конфигурация КА

$$(S_0, 0+01) \vdash^1 (A, +01) \vdash^4 (B, 01) \vdash^5 (C, 1) \vdash^6 (q_f, \epsilon)$$

$$(S_0, 00+0101) \vdash^1 (A, 0+0101) \vdash^2 (A, +0101) \vdash^4 (B, 0101) \vdash^5 (C, 101) \vdash^6 (B, 01) \vdash^5 (C, 1) \vdash^6 (q_f, \epsilon)$$

$$(S_0, 01+) \vdash^1 (A, 1+) \vdash^3 (A, +) \vdash^6 (q_f, \epsilon)$$

$$\begin{aligned}
& (S_0, 000+010101) \vdash^1 (A, 00+010101) \vdash^2 (A, 0+010101) \vdash^2 (A, +010101) \vdash^4 (B, 010101) \vdash^5 \\
& (C, 10101) \vdash^6 (B, 0101) \vdash^5 (C, 101) \vdash^6 (B, 01) \vdash^5 (C, 1) \vdash^6 (q_f, \varepsilon) \\
& (S_0, 001+0101) \vdash^1 (A, 01+0101) \vdash^2 (A, 1+0101) \vdash^3 (A, +0101) \vdash^4 (B, 0101) \vdash^5 (C, 101) \vdash^6 \\
& (B, 01) \vdash^5 (C, 1) \vdash^6 (q_f, \varepsilon)
\end{aligned}$$

$$L(KA) \equiv L(G)$$

## Лабораторная работа №3

### Формулировка задания

Реализовать преобразование НДКА в ДКА. Реализовать конечный автомат по диаграмме переходов.

### Преобразование НДКА в ДКА

#### НДКА

```
myAutomate ndka = new myAutomate(  
    new ArrayList() { "S0", "A", "B", "C", "qf" },  
    new ArrayList() { "1", "0", "+" },  
    new ArrayList() { "qf" },  
    "S0");  
  
ndka.AddRule("S0", "0", "A");  
ndka.AddRule("A", "0", "A");  
ndka.AddRule("A", "1", "A");  
ndka.AddRule("A", "+", "B");  
ndka.AddRule("B", "0", "C");  
ndka.AddRule("C", "1", "B");  
ndka.AddRule("C", "1", "qf");
```

#### ДКА

Automate config:

Q: S0 A C Bqf

Sigma: 1 0 +

Q0: S0

F: Bqf

DeltaList:

: (S0 , 0 ) -> A

: (A , 1 ) -> A

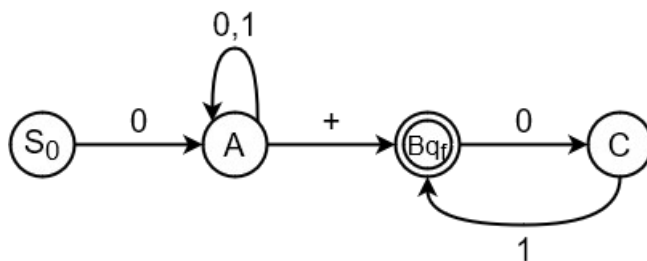
: (A , 0 ) -> A

: (A , + ) -> Bqf

: (C , 1 ) -> Bqf

: (Bqf , 0 ) -> C

### Диаграмма переходов ДКА



## Распознавание цепочек

Enter line to execute :

0+01

Length: 4

i :4

curr: Bqf

chineSymbol belongs to language

Enter line to execute :

00+0101

Length: 7

i :7

curr: Bqf

chineSymbol belongs to language

Enter line to execute :

01+

Length: 3

i :3

curr: Bqf

chineSymbol belongs to language

Enter line to execute :

000+010101

Length: 10

i :10

curr: Bqf

chineSymbol belongs to language

Enter line to execute :

001+0101

Length: 8

i :8

curr: Bqf

chineSymbol belongs to language

## Лабораторная работа №4

### Формулировка задания

Устранить из КС-грамматики бесполезные символы и эпсилон–правила.

$$G = (\{b, c, a, d\}, \{S, F, B, A, C\}, P, S)$$

$$P = \{S \rightarrow Sb, S \rightarrow F, S \rightarrow cFB, A \rightarrow Ab, A \rightarrow c, B \rightarrow cB, F \rightarrow Ca, C \rightarrow d, C \rightarrow \varepsilon\}$$

### Вывод цепочек

$$S \Rightarrow Sb \Rightarrow Sbb \Rightarrow Fbb \Rightarrow Cabb \Rightarrow abb$$

$$S \Rightarrow Sb \Rightarrow Sbb \Rightarrow Sbbb \Rightarrow Fbbb \Rightarrow Cabbb \Rightarrow dabbb$$

$$S \Rightarrow F \Rightarrow Ca \Rightarrow da$$

$$S \Rightarrow F \Rightarrow Ca \Rightarrow a$$

$$S \Rightarrow cFB \Rightarrow cCaB \Rightarrow cdaB \Rightarrow cdacB \Rightarrow cdaccB \Rightarrow cdacccB \Rightarrow cdaccccB$$

### Алгоритм удаления бесполезных символов

#### Алгоритм удаления непроектирующих символов

**Вход:**  $G = (T, V, P, S)$

**Выход:**  $G' = (T, V', P', S)$

$$\text{Шаг 1: } V_p^0 = \emptyset$$

$$\text{Шаг 2: } V_p^1 = \{A, C\}$$

$$\text{Шаг 3: } V_p^2 = \{A, C, F\}$$

$$\text{Шаг 4: } V_p^3 = \{A, C, F, S\}$$

$$\text{Шаг 5: } V_p^4 = \{A, C, F, S\}$$

$$V_p^3 = V_p^4$$

$$V' = V \cap V_p = \{A, C, F, S\}$$

$$P' = \{S \rightarrow Sb, S \rightarrow F, A \rightarrow Ab, A \rightarrow c, F \rightarrow Ca, C \rightarrow d, C \rightarrow \varepsilon\}$$

$$G = (\{b, c, a, d\}, \{S, F, A, C\}, P', S)$$

#### Алгоритм удаления недостижимых символов

**Вход:**  $G = (T, V, P, S)$

**Выход:**  $G = (T', V', P', S)$

$$\text{Шаг 1: } VT_r^0 = \{S\}$$

$$\text{Шаг 2: } VT_r^1 = \{S, b, F\}$$

$$\text{Шаг 3: } VT_r^2 = \{S, b, F, C, a\}$$

$$\text{Шаг 4: } VT_r^3 = \{S, b, F, C, a, d\}$$

$$\text{Шаг 5: } VT_r^4 = \{S, b, F, C, a, d\}$$

$$VT_r^3 = VT_r^4$$

$$VT_r = \{S, b, F, C, a, d\}$$

$$T' = T \cap VT_r = \{a, b, d\}$$

$$V' = V \cap VT_r = \{S, F, C\}$$

$$P' = \{S \rightarrow Sb, S \rightarrow F, F \rightarrow Ca, C \rightarrow d, C \rightarrow \varepsilon\}$$

$$G = (\{a, b, d\}, \{S, F, C\}, P', S)$$

### **Алгоритм построение множества укорачивающих нетерминалов**

**Вход:**  $G = (T, V, P, S)$

**Выход:**  $V_\varepsilon = \{C\}$

Шаг 1:  $V_\varepsilon^0 = \emptyset$

Шаг 2:  $V_\varepsilon^1 = \{C\}$

Шаг 3:  $V_\varepsilon^2 = \{C\}$

### **Алгоритм удаления эпсилон-правил**

**Вход:**  $G = (T, V, P, S)$

**Выход:**  $G' = (T, V', P', S')$

$$S' = S$$

$$V' = V$$

$$P' = \{S \rightarrow Sb, S \rightarrow F, F \rightarrow Ca, F \rightarrow a, C \rightarrow d\}$$

### **Грамматика без бесполезных символов и эпсилон-правил**

$$G = (\{S, F, C\}, \{a, b, d\}, P, S)$$

$$P = \{S \rightarrow Sb, S \rightarrow F, F \rightarrow Ca, F \rightarrow a, C \rightarrow d\}$$

### **Вывод цепочек**

$$S \Rightarrow Sb \Rightarrow Sbb \Rightarrow Fbb \Rightarrow abb$$

$$S \Rightarrow Sb \Rightarrow Sbb \Rightarrow Sbbb \Rightarrow Fbbb \Rightarrow Cabbb \Rightarrow dabbb$$

$$S \Rightarrow F \Rightarrow Ca \Rightarrow da$$

$$S \Rightarrow F \Rightarrow a$$



## **Лабораторная работа №5**

### **Формулировка задания**

Устранить из КС-грамматики цепные правила и левую рекурсию

$$G = (\{S, F, C\}, \{a, b, d\}, P, S)$$

$$P = \{S \rightarrow Sb, S \rightarrow F, F \rightarrow Ca, F \rightarrow a, C \rightarrow d\}$$

### **Алгоритм удаления цепных правил**

**Вход:**  $G = (T, V, P, S)$

**Выход:**  $G' = (T, V, P', S)$

$$V_S = \{S, F\}, V_F = \{F\}, V_C = \{C\}$$

Шаг 1:  $S \rightarrow F$

$$S \rightarrow Ca \mid a$$

Шаг 2: Цепных правил не осталось

$$P' = \{S \rightarrow Sb, S \rightarrow Ca, S \rightarrow a, C \rightarrow d\}$$

$$G = (\{S, C\}, \{a, b, d\}, P', S)$$

### **Алгоритм удаления левой рекурсии**

**Вход:**  $G = (T, V, P, S)$

**Выход:**  $G' = (T, V', P', S)$

Шаг 1:  $V = \{S, C\} = \{A_1, A_2, A_3\}$

Шаг 2:  $i = 1. S \rightarrow Sb \mid Ca \mid a$

$$S \rightarrow Ca \mid a \mid CaS' \mid aS'$$

$$S' \rightarrow b \mid bS'$$

Шаг 3:  $i = 2, j = 1$ . Подходящих правил нет.

Шаг 4:  $i = 2$ . Подходящих правил нет.

Шаг 5:  $i = 3, j = 1$ . Подходящих правил нет.

Шаг 6:  $i = 3, j = 2$ . Подходящих правил нет.

Шаг 7:  $i = 3$ . Подходящих правил нет.

$$V' = \{S, C, S'\}$$

$$P' = \{S \rightarrow CaS', S \rightarrow aS', S' \rightarrow bS', S' \rightarrow b, S \rightarrow Ca, S \rightarrow a, C \rightarrow d\}$$

$$G = (\{S, C, S'\}, \{a, b, d\}, P', S)$$

### **Грамматика без цепных правил и левой рекурсии**

$$G = (\{S, C, S'\}, \{a, b, d\}, P, S)$$

$$P = \{S \rightarrow CaS', S \rightarrow aS', S' \rightarrow bS', S' \rightarrow b, S \rightarrow Ca, S \rightarrow a, C \rightarrow d\}$$

### **Вывод цепочек**

$S \Rightarrow aS' \Rightarrow abS' \Rightarrow abb$

$S \Rightarrow CaS' \Rightarrow daS' \Rightarrow dabS' \Rightarrow dabbS' \Rightarrow dabbbb$

$S \Rightarrow Ca \Rightarrow da$

$S \Rightarrow a$

## **Лабораторная работа №6**

### **Формулировка задания**

Определить форму КС-грамматики и сделать ее приведение.

### **Исходная грамматика**

$G = (\{b, c, a, d\}, \{S, F, B, A, C\}, P, S)$

$P = \{S \rightarrow Sb, S \rightarrow F, S \rightarrow cFB, A \rightarrow Ab, A \rightarrow c, B \rightarrow cB, F \rightarrow Ca, C \rightarrow d, C \rightarrow \epsilon\}$

### **Вывод цепочек**

$S \Rightarrow Sb \Rightarrow Sbb \Rightarrow Fbb \Rightarrow Cabb \Rightarrow abb$

$S \Rightarrow Sb \Rightarrow Sbb \Rightarrow Sbbb \Rightarrow Fbbb \Rightarrow Cabbb \Rightarrow dabbb$

$S \Rightarrow F \Rightarrow Ca \Rightarrow da$

$S \Rightarrow F \Rightarrow Ca \Rightarrow a$

$S \Rightarrow cFB \Rightarrow cCaB \Rightarrow cdaB \Rightarrow cdacB \Rightarrow cdaccB \Rightarrow cdacccB \Rightarrow cdaccccB$

### **Приведённая грамматика**

$G = (\{S, F, C, S'\}, \{a, b, d\}, P, S)$

$P = \{S \rightarrow CaS', S \rightarrow aS', S' \rightarrow bS', S' \rightarrow b, S \rightarrow Ca, S \rightarrow a, C \rightarrow d\}$

Грамматика не в нормальной форме Грейбах, так как имеет правила, начинающиеся с нетерминалов.

Грамматика не в нормальной форме Хомского, так как имеет правила из нетерминалов и терминалов.

### **Вывод цепочек**

$S \Rightarrow aS' \Rightarrow abS' \Rightarrow abb$

$S \Rightarrow CaS' \Rightarrow daS' \Rightarrow dabS' \Rightarrow dabbS' \Rightarrow dabbbb$

$S \Rightarrow Ca \Rightarrow da$

$S \Rightarrow a$

## Лабораторная работа №7

### Формулировка задания

Спроектировать МП-автомат для приведенной КС-грамматики.

### Приведённая грамматика

$G = (\{S, C, S'\}, \{a, b, d\}, P, S)$

$P = \{S \rightarrow CaS', S \rightarrow aS', S \rightarrow Ca, S \rightarrow a, S' \rightarrow bS', S' \rightarrow b, C \rightarrow d\}$

### МП-автомат

$МП = (\{q\}, \{a, b, d\}, \{a, b, d, S, C, S'\}, \delta, q, S, \{q\})$

$\delta$ :

1.  $\delta(q, \varepsilon, S) = \{(q, CaS'), (q, aS'), (q, Ca), (q, a)\}$
2.  $\delta(q, \varepsilon, S') = \{(q, bS'), (q, b)\}$
3.  $\delta(q, \varepsilon, C) = \{(q, d)\}$
4.  $\delta(q, a, a) = \{(q, \varepsilon)\}$
5.  $\delta(q, b, b) = \{(q, \varepsilon)\}$
6.  $\delta(q, d, d) = \{(q, \varepsilon)\}$

### Распознавание цепочек

1.  $(q, abb, S) \vdash^1 (q, abb, aS') \vdash^4 (q, bb, S') \vdash^2 (q, bb, bS') \vdash^5 (q, b, S') \vdash^2 (q, b, b) \vdash^5 (q, \varepsilon, \varepsilon)$
2.  $(q, dabbb, S) \vdash^1 (q, dabbb, CaS') \vdash^3 (q, dabbb, daS') \vdash^6 (q, abbb, aS') \vdash^4 (q, bbb, S') \vdash^2 (q, bbb, bS') \vdash^5 (q, bb, S') \vdash^2 (q, bb, bS') \vdash^5 (q, b, S') \vdash^2 (q, b, b) \vdash^5 (q, \varepsilon, \varepsilon)$
3.  $(q, da, S) \vdash^1 (q, da, Ca) \vdash^3 (q, da, da) \vdash^6 (q, a, a) \vdash^4 (q, \varepsilon, \varepsilon)$
4.  $(q, a, S) \vdash^1 (q, a, a) \vdash^4 (q, \varepsilon, \varepsilon)$

## Лабораторная работа №8

### Формулировка задания

Реализовать МП автомат для приведенной КС-грамматики.

### МП-автомат

```
myMp mp = new myMp(new ArrayList() { "a", "b", "d" },
                    new ArrayList() { "S", "C", "D" },
                    new ArrayList() { "S", "C", "D", "a", "b", "d" },
                    "q",
                    "S",
                    new ArrayList() { "q" });
mp.addDeltaRule("q", "", "S", new ArrayList() { "q" }, new ArrayList() { "a" });
mp.addDeltaRule("q", "", "S", new ArrayList() { "q" }, new ArrayList() { "C", "a",
"D" });
mp.addDeltaRule("q", "", "S", new ArrayList() { "q" }, new ArrayList() { "a", "D" });
mp.addDeltaRule("q", "", "S", new ArrayList() { "q" }, new ArrayList() { "C", "a" });
mp.addDeltaRule("q", "", "D", new ArrayList() { "q" }, new ArrayList() { "b", "D" });
mp.addDeltaRule("q", "", "D", new ArrayList() { "q" }, new ArrayList() { "b" });
mp.addDeltaRule("q", "", "C", new ArrayList() { "q" }, new ArrayList() { "d" });
mp.addDeltaRule("q", "a", "a", new ArrayList() { "q" }, new ArrayList() { "" });
mp.addDeltaRule("q", "b", "b", new ArrayList() { "q" }, new ArrayList() { "" });
mp.addDeltaRule("q", "d", "d", new ArrayList() { "q" }, new ArrayList() { "" });
```

### Распознавание цепочек

Prules:

S -> a  
S -> CaD  
S -> aD  
S -> Ca  
D -> bD  
D -> b  
C -> d

Deltarules :

delta(Q q, T , Z S)	= (Q q, Z CaD)
delta(Q q, T , Z D)	= (Q q, Z b)
delta(Q q, T , Z S)	= (Q q, Z a)
delta(Q q, T , Z S)	= (Q q, Z aD)
delta(Q q, T , Z S)	= (Q q, Z Ca)
delta(Q q, T , Z D)	= (Q q, Z bD)
delta(Q q, T , Z C)	= (Q q, Z d)
delta(Q q, T a, Z a)	= (Q q, Z )
delta(Q q, T b, Z b)	= (Q q, Z )
delta(Q q, T d, Z d)	= (Q q, Z )

Введите строку :

dab

delta(Q q, T , Z S)	= (Q q, Z CaD)
delta(Q q, T , Z C)	= (Q q, Z d)
delta(Q q, T d, Z d)	= (Q q, Z )

```
delta(Q q,T a,Z a)      = (Q q,Z )
delta(Q q,T ,Z D)       = (Q q,Z b)
delta(Q q,T b,Z b)      = (Q q,Z )
True
```

## Лабораторная работа №9

### Формулировка задания

Для LL(1) анализатора построить управляющую таблицу M.

Определить FIRST и FOLLOW. Построить управляющую таблицу построчно.

### Грамматика

$$G = (V, T, P, S)$$

$$T = \{i, =, *, (, )\}$$

$$V = \{S, F, L\}$$

P:

1.  $S \rightarrow F=L$
2.  $S \rightarrow L$
3.  $F \rightarrow (*L)$
4.  $F \rightarrow i$
5.  $L \rightarrow F$

Грамматика не принадлежит классу LL(1), так как для правил  $S \rightarrow F=L \mid L$  :

$$\text{FIRST}(F=L) = \text{FIRST}(L) = \text{FIRST}(F) = \{(, i\}$$

### Преобразование к LL(1)-грамматике

Заметим, что цепное правило  $L \rightarrow F$  является единственным правилом для L, следовательно можно заменить L на F в правых частях других правил и удалить L из грамматики. Получим грамматику:

$$G = (\{S, F\}, \{i, =, *, (, )\}, P, S)$$

P:

1.  $S \rightarrow F=F$
2.  $S \rightarrow F$
3.  $F \rightarrow (*F)$
4.  $F \rightarrow i$

Проведем левую факторизацию по следующему алгоритму:

---

**Алгоритм.** Левая факторизация грамматики

**Вход:** грамматика  $G$ .

**Выход:** эквивалентная левофакторизованная грамматика.

**Метод:** для каждого нетерминала  $A$  находим самый длинный префикс  $\alpha$ , общий для двух или большего числа альтернатив. Если  $\alpha \neq \varepsilon$ , т.е. имеется нетривиальный общий префикс, заменим все продукции  $A \rightarrow \alpha\beta_1 \mid \alpha\beta_2 \mid \dots \mid \alpha\beta_n \mid \gamma$ , где  $\gamma$  представляет все альтернативы, не начинающиеся с  $\alpha$ , продукциями

$$A \rightarrow \alpha A' \mid \gamma$$

$$A' \rightarrow \beta_1 \mid \beta_2 \mid \dots \mid \beta_n$$

Здесь  $A'$  — новый нетерминал. Выполняем это преобразование до тех пор, пока никакие две альтернативы нетерминала не будут иметь общий префикс.

---

Применив алгоритм к правилам  $S \rightarrow F=F \mid F$ , заменяем их на правила  $S \rightarrow FS'$ ,  $S' \rightarrow =F \mid \varepsilon$ . Получим грамматику без неопределённостей:

$$G = (\{S, S', F\}, \{i, =, *, (, )\}, P, S)$$

P:

1.  $S \rightarrow FS'$
2.  $S' \rightarrow =F$
3.  $S' \rightarrow \varepsilon$
4.  $F \rightarrow (*F)$
5.  $F \rightarrow i$

### Построение функции FIRST

$$\text{FIRST}(S) = \text{FIRST}(F) = \{(, i\}$$

$$\text{FIRST}(S') = \{=, \varepsilon\}$$

$$\text{FIRST}(F) = \{(, i\}$$

$$\text{FIRST}(FS') = \text{FIRST}(F) = \{(, i\}$$

$$\text{FIRST}((*F)) = \{( \}$$

### Построение функции FOLLOW

$$\text{FOLLOW}(S) = \{\perp\}$$

$$\text{FOLLOW}(S') = \{\perp\}$$

$$\text{FOLLOW}(F) = \text{FIRST}(S') \setminus \{\varepsilon\} \cup \text{FOLLOW}(S') \cup \{)\} = \{=, \perp, )\}$$

### Построение управляющей таблицы М

М	i	=	*	(	)	ε
S	FS', 1			FS', 1		
S'		=F, 2				ε, 3
F	i, 5			(*F), 4		
i	ВЫБРОС					
=		ВЫБРОС				
*			ВЫБРОС			
(				ВЫБРОС		
)					ВЫБРОС	
⊥						ДОПУСК



## Лабораторная работа №10

### Формулировка задания

Рассмотреть работу алгоритма для цепочки символов, порожденной LL(1) грамматикой.

$$G = (\{S, S', F\}, \{i, =, *, (, )\}, P, S)$$

$$P = \{ S \rightarrow FS', S' \rightarrow =F, S' \rightarrow \varepsilon, F \rightarrow (*F), F \rightarrow i \}$$

### Вывод цепочек

1.  $S \Rightarrow^1 FS' \Rightarrow^4 (*F)S' \Rightarrow^4 ((*F))S' \Rightarrow^5 ((*i))S' \Rightarrow^2 ((*i))=F \Rightarrow^4 ((*i))=(*F) \Rightarrow^5 ((*i))=(*i)$
2.  $S \Rightarrow S \Rightarrow^1 FS' \Rightarrow^5 iS' \Rightarrow^2 i=F \Rightarrow^5 i=i$
3.  $S \Rightarrow^1 FS' \Rightarrow^4 (*F)S' \Rightarrow^4 ((*F))S' \Rightarrow^4 ((*(*F)))S' \Rightarrow^5 ((*(*i)))S' \Rightarrow^6 ((*(*i)))$

### Распознавание цепочек

1.  $(((*i))=(*i), S^\perp, \varepsilon) \vdash$   
 $(((*i))=(*i), FS'^\perp, 1) \vdash$   
 $(((*i))=(*i), (*F)S'^\perp, 14) \vdash (((*i))=(*i), *F)S'^\perp, 14) \vdash ((i))=(*i), F)S'^\perp, 14) \vdash$   
 $((i))=(*i), (*F))S'^\perp, 144) \vdash ((i))=(*i), (*F))S'^\perp, 144) \vdash ((i))=(*i), F))S'^\perp, 144) \vdash$   
 $((i))=(*i), i))S'^\perp, 1445) \vdash ((i))=(*i), ))S'^\perp, 1445) \vdash$   
 $((i))=(*i), )S'^\perp, 1445) \vdash ((i))=(*i), S'^\perp, 1445) \vdash$   
 $((i))=(*i), =F)S'^\perp, 14452) \vdash ((i))=(*i), F)S'^\perp, 14452) \vdash$   
 $((i))=(*i), (*F)S'^\perp, 144524) \vdash ((i))=(*i), *F)S'^\perp, 144524) \vdash ((i))=(*i), F)S'^\perp, 144524) \vdash$   
 $((i))=(*i), i)S'^\perp, 1445245) \vdash ((i))=(*i), )S'^\perp, 1445245) \vdash ((i))=(*i), \varepsilon)S'^\perp, 1445245)$
2.  $(i=i, S^\perp, \varepsilon) \vdash$   
 $(i=i, FS'^\perp, 1) \vdash$   
 $(i=i, iS'^\perp, 15) \vdash ((i))=(*i), S'^\perp, 15) \vdash$   
 $((i))=(*i), =F)S'^\perp, 152) \vdash ((i))=(*i), F)S'^\perp, 152) \vdash$   
 $((i))=(*i), i)S'^\perp, 1525) \vdash ((i))=(*i), \varepsilon)S'^\perp, 1525)$
3.  $(((*(*i))), S^\perp, \varepsilon) \vdash$   
 $(((*(*i))), FS'^\perp, 1) \vdash$   
 $(((*(*i))), (*F)S'^\perp, 14) \vdash (((*(*i))), *F)S'^\perp, 14) \vdash$   
 $(((*(*i))), F)S'^\perp, 14) \vdash$   
 $(((*(*i))), (*F))S'^\perp, 144) \vdash (((*(*i))), *F))S'^\perp, 144) \vdash (((*(*i))), F))S'^\perp, 144) \vdash$   
 $(((*(*i))), (*F)))S'^\perp, 1444) \vdash (((*(*i))), *F)))S'^\perp, 1444) \vdash (((*(*i))), F)))S'^\perp, 1444) \vdash$   
 $(((*(*i))), i)))S'^\perp, 14445) \vdash (((*(*i))), ))S'^\perp, 14445) \vdash (((*(*i))), )S'^\perp, 14445) \vdash$   
 $(((*(*i))), )S'^\perp, 14445) \vdash (((*(*i))), \varepsilon)S'^\perp, 14445) \vdash$   
 $(((*(*i))), \varepsilon)S'^\perp, 144453)$

# Лабораторная работа №11

## Формулировка задания

Реализовать управляющую таблицу М для LL(1) анализатора.

## Грамматика

```
myGrammar grammar = new myGrammar(new ArrayList() { "i", "=", "*", "(", ")", ",", "" },
                                     new ArrayList() { "S", "S'", "F" },
                                     "S");
```

```
grammar.AddRule("S", new ArrayList() { "F", "S'" });
grammar.AddRule("S'", new ArrayList() { "=", "F" });
grammar.AddRule("S'", new ArrayList() { "" });
grammar.AddRule("F", new ArrayList() { "(", "*", "F", ")" });
grammar.AddRule("F", new ArrayList() { "i" });
```

## Построение управляющей таблицы М

Создадим таблицу. Сначала создадим по столбцу для каждого из этих терминалов:  
i, =, \*, (, ), ,

Также создаем строку для Эпсилон

Рассмотрим нетерминал S

Первый символ правила S → FS' - (

Это правило заносим в таблицу на пересечении строки нетерминала S и столбца терминала (

Первый символ правила S → FS' - i

Это правило заносим в таблицу на пересечении строки нетерминала S и столбца терминала i

Рассмотрим нетерминал S'

Первый символ правила S' → =F - =

Это правило заносим в таблицу на пересечении строки нетерминала S' и столбца терминала =

Рассмотрим нетерминал F

Первый символ правила F → (\*F) - (

Это правило заносим в таблицу на пересечении строки нетерминала F и столбца терминала (

Первый символ правила F → i - i

Это правило заносим в таблицу на пересечении строки нетерминала F и столбца терминала i

## Распознавание цепочек

Введите строку:

(\*(i))=(\*i)

Успех. Строка соответствует грамматике.

1445245

Введите строку:

i=i

Успех. Строка соответствует грамматике.

1525

Введите строку:

(\*(\*(i)))

Успех. Строка соответствует грамматике.

144453

## **Лабораторные работы №12-15**

### **Формулировка задания**

Определить пополненную LR(1) грамматику.

Определить активные префиксы. Определить функцию перехода  $g(x)$ . Построить управляющую таблицу для функций перехода и действий.

Определить множество FIRST для LR(1) грамматики. Определить множество LR(0) ситуаций. Построить замыкание CLOSURE(I) множества ситуаций. Определить функцию перехода  $GOTO(I, x)$ . Построить каноническую форму множества ситуаций. Построить диаграмму переходов автомата. Построить управляющую таблицу для функций перехода и действий.

### **Исходная грамматика**

$G = (\{S, F, L\}, \{i, *, =, (, )\}, P, S)$

P:

1.  $S \rightarrow F=L$
2.  $S \rightarrow L$
3.  $F \rightarrow (*L)$
4.  $F \rightarrow i$
5.  $L \rightarrow F$

### **Пополненная грамматика**

$G = (\{S', S, F, L\}, \{i, *, =, (, )\}, P, S)$

P:

0.  $S' \rightarrow S$
1.  $S \rightarrow F=L$
2.  $S \rightarrow L$
3.  $F \rightarrow (*L)$
4.  $F \rightarrow i$
5.  $L \rightarrow F$

## Построение LR(k) анализатора способом использования расширенного магазинного алфавита

### Определение активных префиксов

Символ грамматики	Магазинный символ	Кодируемая цепочка
S	$S_0$	$\perp S$
F	$F_1$	F
	$F_5$	F
L	$L_1$	$F=L$
	$L_2$	L
	$L_3$	$(*L$
i	$i_4$	i
*	$*_3$	$(*$
=	$=_1$	$F=$
(	$(_3$	(
)	$)_3$	$(*L)$

### Функция переходов g(x)

g(x)	i	=	*	(	)	S	F	L
$S_0$								
$F_1$		$=_1$						
$F_5$					$)_3$			
$L_1$								
$L_2$								
$L_3$					$)_3$			

$i_4$		$=_1$			$)_3$			
$*_3$	$i_4$			$(_3$			$F_5$	$L_3$
$=_1$	$i_4$			$(_3$			$F_1$	$L_1$
$(_3$			$*_3$					
$)_3$		$=_1$						
$\perp$	$i_4$			$(_3$		$S_0$	$F_1, F_5$	$L_2$

Автомат переходов получается недетерминированным. Введем новый магазинный символ  $F_x = \{F_1, F_5\}$ .  $F_x$  кодирует цепочку  $F$ .

### Управляющая таблица

M	Функция действий						Функция переходов							
	i	=	*	(	)	$\perp$	i	=	*	(	)	S	F	L
$S_0$						Д								
$F_x$	C, 5	П	C, 5	C, 5	C, 5	C, 5		$=_1$			$)_3$			
$L_1$	C, 1	C, 1	C, 1	C, 1	C, 1	C, 1								
$L_2$	C, 2	C, 2	C, 2	C, 2	C, 2	C, 2								
$L_3$	П	П	П	П	П						$)_3$			
$i_4$	C, 4	C, 4	C, 4	C, 4	C, 4	C, 4		$=_1$			$)_3$			
$*_3$	П	П	П	П	П		$i_4$			$(_3$			$F_x$	$L_3$
$=_1$	П	П	П	П	П		$i_4$			$(_3$			$F_x$	$L_1$
$(_3$	П	П	П	П	П				$*_3$					
$)_3$	C, 3	C, 3	C, 3	C, 3	C, 3	C, 3		$=_1$						
$\perp$	П	П	П	П	П		$i_4$			$(_3$		$S_0$	$F_x$	$L_2$

## Распознавание цепочек

1.  $(\perp, i=i\perp, \varepsilon) \vdash^\Pi (\perp i_4, =i\perp, \varepsilon) \vdash^C (\perp F_x, =i\perp, 4) \vdash^\Pi (\perp F_x=i, i\perp, 4) \vdash^\Pi (\perp F_x=i i_4, \perp, 4) \vdash^C$   
 $(\perp F_x=i F_x, \perp, 44) \vdash^C (\perp F_x=i L_1, \perp, 445) \vdash^C (\perp S_0, \perp, 4451) \vdash^\Delta$
2.  $(\perp, i\perp, \varepsilon) \vdash^\Pi (\perp i_4, \perp, \varepsilon) \vdash^C (\perp F_x, \perp, 4) \vdash^C (\perp L_2, \perp, 45) \vdash^C (\perp S_0, \perp, 452) \vdash^\Delta$
3.  $(\perp, (*i)\perp, \varepsilon) \vdash (\perp_{(3, *i)}\perp, \varepsilon) \vdash (\perp_{(3^*_3, i)}\perp, \varepsilon) \vdash (\perp_{(3^*_3, i)}\perp, \varepsilon) \vdash (\perp_{(3^*_3 i_4)}\perp, \varepsilon) \vdash$   
 $(\perp_{(3^*_3 F_x)}\perp, \varepsilon) \vdash (\perp_{(3^*_3 L_3)}\perp, \varepsilon) \vdash (\perp_{(3^*_3 L_3)}\perp, \varepsilon) \vdash (\perp_{(3^*_3 L_3)_3}, \perp, \varepsilon) \vdash (\perp_{(3^*_3 L_3)_3}, \perp, \varepsilon) \vdash$   
 $(\perp S_0, \perp, \varepsilon) \vdash^\Delta$

## Построение LR(k) анализатора способом грамматических вхождений

### Построение функции FIRST

$$\text{FIRST}(S) = \text{FIRST}(F) \cup \text{FIRST}(L) = \{ (, i \}$$

$$\text{FIRST}(F) = \{ (, i \}$$

$$\text{FIRST}(L) = \text{FIRST}(F) = \{ (, i \}$$

$$\text{FIRST}(F=L) = \text{FIRST}(F) = \{ (, i \}$$

$$\text{FIRST}(( * L )) = \{ ( \}$$

### Построение функции FOLLOW

$$\text{FOLLOW}(S) = \{ \perp \}$$

$$\text{FOLLOW}(F) = \{ = \} \cup \text{FOLLOW}(L) = \{ =, \perp, ) \}$$

$$\text{FOLLOW}(L) = \text{FOLLOW}(S) \cup \{ \} = \{ \perp, ) \}$$

### Замыкание множества ситуаций

$$I = \text{CLOSURE}(\{ S' \rightarrow \cdot S \})$$

$$\text{Шаг 1: } I = \{ S' \rightarrow \cdot S \}$$

$$\text{Шаг 2: } I = I \cup \{ S \rightarrow \cdot F=L, S \rightarrow \cdot L \}$$

$$\text{Шаг 3: } I = I \cup \{ F \rightarrow \cdot (*L), F \rightarrow \cdot i \}$$

$$\text{Шаг 4: } I = I \cup \{ L \rightarrow \cdot F \}$$

$$I = \{ S' \rightarrow \cdot S, S \rightarrow \cdot F=L, S \rightarrow \cdot L, F \rightarrow \cdot (*L), F \rightarrow \cdot i, L \rightarrow \cdot F \}$$

### Функция переходов GOTO

$$I_0 = \text{CLOSURE}(\{ S' \rightarrow \cdot S \}) = \{ S' \rightarrow \cdot S, S \rightarrow \cdot F=L, S \rightarrow \cdot L, F \rightarrow \cdot (*L), F \rightarrow \cdot i, L \rightarrow \cdot F \}$$

$$\text{GOTO}(I_0, S) = \{ S' \rightarrow S \cdot \} = I_1$$

$$\text{GOTO}(I_0, F) = \{ S \rightarrow F \cdot=L, L \rightarrow F \cdot \} = I_2$$

$$\text{GOTO}(I_0, L) = \{ S \rightarrow L \cdot \} = I_3$$

$$\text{GOTO}(I_0, i) = \{ F \rightarrow i \cdot \} = I_4$$

$$\text{GOTO}(I_0, ( ) = \{ F \rightarrow ( \cdot * L ) \} = I_5$$

$$\text{GOTO}(I_2, =) = \{ S \rightarrow F = \cdot L, L \rightarrow \cdot F, F \rightarrow \cdot (*L), F \rightarrow \cdot i \} = I_6$$

$$\text{GOTO}(I_5, *) = \{ F \rightarrow (* \cdot L), L \rightarrow \cdot F, F \rightarrow \cdot (*L), F \rightarrow \cdot i \} = I_7$$

$$\text{GOTO}(I_6, L) = \{ S \rightarrow F = L \cdot \} = I_8$$

$$\text{GOTO}(I_6, F) = \{ L \rightarrow F \cdot \} = I_9$$

$$\text{GOTO}(I_6, ( ) = \{ F \rightarrow ( \cdot * L ) \} = I_5$$

$$\text{GOTO}(I_6, i) = \{ F \rightarrow i \cdot \} = I_4$$

$$\text{GOTO}(I_7, L) = \{ F \rightarrow (* L \cdot ) \} = I_{10}$$

$$\text{GOTO}(I_7, F) = \{ L \rightarrow F \cdot \} = I_9$$



$GOTO(I_7, ( ) = \{F \rightarrow (\cdot * L)\} = I_5$

$GOTO(I_7, i) = \{F \rightarrow i \cdot\} = I_4$

$GOTO(I_{10}, ) = \{F \rightarrow (*L) \cdot\} = I_{11}$

### Каноническая форма множества ситуаций

$C = \{$

$I_0 = \{S' \rightarrow \cdot S, S \rightarrow \cdot F=L, S \rightarrow \cdot L, F \rightarrow \cdot (*L), F \rightarrow \cdot i, L \rightarrow \cdot F\},$

$I_1 = \{S' \rightarrow S \cdot\},$

$I_2 = \{S \rightarrow F \cdot L, L \rightarrow F \cdot\},$

$I_3 = \{S \rightarrow L \cdot\},$

$I_4 = \{F \rightarrow i \cdot\},$

$I_5 = \{F \rightarrow (\cdot * L)\},$

$I_6 = \{S \rightarrow F \cdot L, L \rightarrow \cdot F, F \rightarrow \cdot (*L), F \rightarrow \cdot i\},$

$I_7 = \{F \rightarrow (* \cdot L), L \rightarrow \cdot F, F \rightarrow \cdot (*L), F \rightarrow \cdot i\},$

$I_8 = \{S \rightarrow F=L \cdot\},$

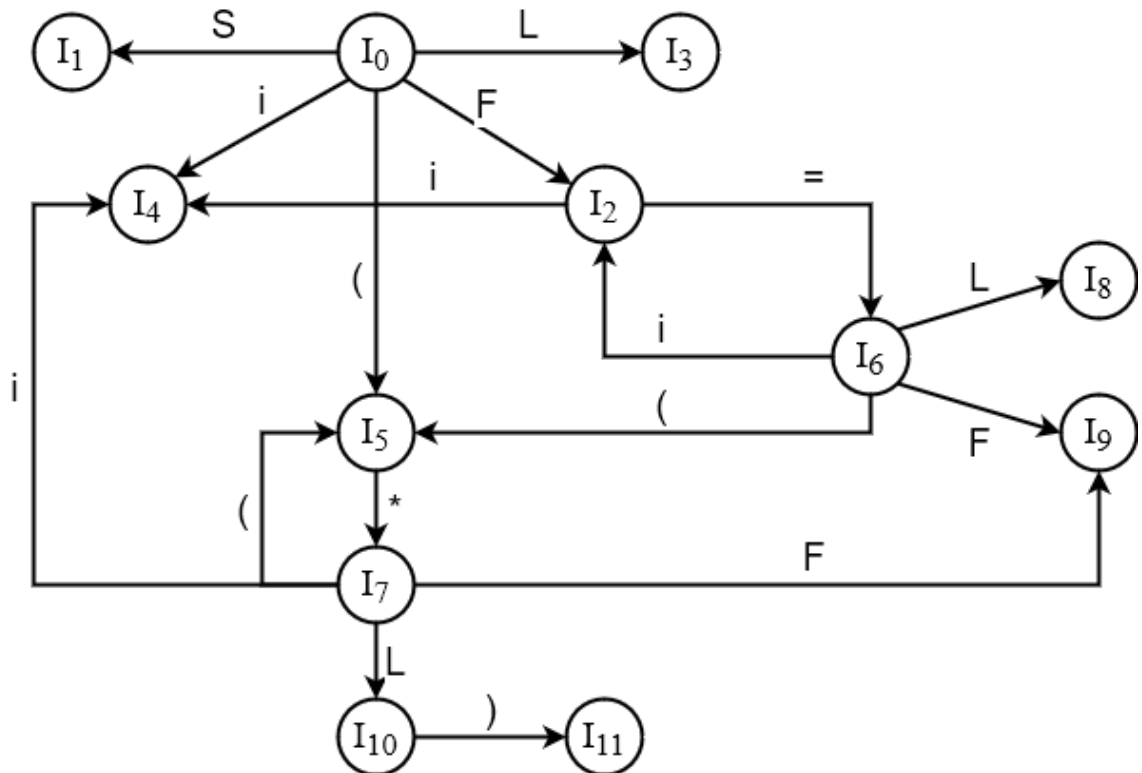
$I_9 = \{L \rightarrow F \cdot\},$

$I_{10} = \{F \rightarrow (*L) \cdot\},$

$I_{11} = \{F \rightarrow (*L) \cdot\}$

$\}$

### Диаграмма переходов автомата



## Управляющая таблица

I	Функция действий						Функция переходов		
	i	=	*	(	)	⊥	S	F	L
0	П, 4			П, 5			1	2	3
1						Д			
2		П, 6			С, 5	С, 5			
3						С, 2			
4		С, 4			С, 4	С, 4			
5			П, 7						
6	П, 4			П, 5				9	8
7	П, 4			П, 5				9	10
8						С, 1			
9					С, 5	С, 5			
10					П, 11				
11		С, 3			С, 3	С, 3			

## Распознавание цепочек

- $(0, (*(*i))=(*i)\perp) \vdash^{\Pi} (0\ 5, (*(*i))=(*i)\perp) \vdash^{\Pi} (0\ 5\ 7, (*i))=(*i)\perp) \vdash^{\Pi} (0\ 5\ 7\ 5, *i))=(*i)\perp) \vdash^{\Pi}$   
 $(0\ 5\ 7\ 5\ 7, i))=(*i)\perp) \vdash^{\Pi} (0\ 5\ 7\ 5\ 7\ 4, ))=(*i)\perp) \vdash^C (0\ 5\ 7\ 5\ 7\ 9, ))=(*i)\perp) \vdash^C$   
 $(0\ 5\ 7\ 5\ 7\ 10, ))=(*i)\perp) \vdash^{\Pi} (0\ 5\ 7\ 5\ 7\ 10\ 11, )=(*i)\perp) \vdash^C (0\ 5\ 7\ 9, )=(*i)\perp) \vdash^C$   
 $(0\ 5\ 7\ 10, )=(*i)\perp) \vdash^{\Pi} (0\ 5\ 7\ 10\ 11, =(*i)\perp) \vdash^C (0\ 2, =(*i)\perp) \vdash^{\Pi} (0\ 2\ 6, (*i)\perp) \vdash^{\Pi}$   
 $(0\ 2\ 6\ 5, *i)\perp) \vdash^{\Pi} (0\ 2\ 6\ 5\ 7, i)\perp) \vdash^{\Pi} (0\ 2\ 6\ 5\ 7\ 4, )\perp) \vdash^C (0\ 2\ 6\ 5\ 7\ 9, )\perp) \vdash^C$   
 $(0\ 2\ 6\ 5\ 7\ 10, )\perp) \vdash^{\Pi} (0\ 2\ 6\ 5\ 7\ 10\ 11, \perp) \vdash^C (0\ 2\ 6\ 9, \perp) \vdash^C (0\ 2\ 6\ 8, \perp) \vdash^C (0\ 1, \perp) \vdash^{\Pi}$
- $(0, i=i\perp) \vdash^{\Pi} (0\ 4, =i\perp) \vdash^C (0\ 2, =i\perp) \vdash^{\Pi} (0\ 2\ 6, i\perp) \vdash^{\Pi} (0\ 2\ 6\ 4, \perp) \vdash^C (0\ 2\ 6\ 9, \perp) \vdash^C$   
 $(0\ 2\ 6\ 8, \perp) \vdash^C (0\ 1, \perp) \vdash^{\Pi}$
- $(0, ((*(*i)))\perp) \vdash^{\Pi} (0\ 5, ((*(*i)))\perp) \vdash^{\Pi} (0\ 5\ 7, ((*i)))\perp) \vdash^{\Pi} (0\ 5\ 7\ 5, ((*i)))\perp) \vdash^{\Pi}$   
 $(0\ 5\ 7\ 5\ 7, (*i)))\perp) \vdash^{\Pi} (0\ 5\ 7\ 5\ 7\ 5, *i)))\perp) \vdash^{\Pi} (0\ 5\ 7\ 5\ 7\ 5\ 7, i)))\perp) \vdash^{\Pi}$

$$\begin{aligned}
& (0\ 5\ 7\ 5\ 7\ 5\ 7\ 4, )))\bot) \vdash^C (0\ 5\ 7\ 5\ 7\ 5\ 7\ 9, )))\bot) \vdash^C (0\ 5\ 7\ 5\ 7\ 5\ 7\ 10, )))\bot) \vdash^\Pi \\
& (0\ 5\ 7\ 5\ 7\ 5\ 7\ 10\ 11, ))\bot) \vdash^C (0\ 5\ 7\ 5\ 7\ 9, ))\bot) \vdash^C (0\ 5\ 7\ 5\ 7\ 10, ))\bot) \vdash^\Pi \\
& (0\ 5\ 7\ 5\ 7\ 10\ 11, )\bot) \vdash^C (0\ 5\ 7\ 9, )\bot) \vdash^C (0\ 5\ 7\ 10, )\bot) \vdash^\Pi (0\ 5\ 7\ 10\ 11, \bot) \vdash^C \\
& (0\ 4, \bot) \vdash^C (0\ 2, \bot) \vdash^C (0\ 3, \bot) \vdash^C (0\ 1, \bot) \vdash^\mathcal{A}
\end{aligned}$$

## Лабораторная работа №16

### Формулировка задания

Реализовать LR(1)-анализатор по управляющей таблице (g,f) для LR(1) грамматики.

### Реализация LR(1)-анализатора

(Этап с пошаговым построением функции GOTO пропущен)

Введите продукции:

```
S F=L
S L
F (*L)
F i
L F
```

Исходная

КС - грамматика :

Алфавит нетерминальных символов: SLF

Алфавит терминальных символов: : i) = (\*

Правила :

```
{ S F=L; S L; F (*L); F i; L F }
```

После удаления е-продукций

Правила:

```
S F=L
S L
F (*L)
F i
L F
```

П S

Терминалы : i) = (\*\$

Нетерминалы: SLFP

-----

Вычислены множества FIRST для символов грамматики и строк

```
First( S ): i(
First( L ): i(
First( F ): (i
First( П ): i(
```

```
First( F=L ): (i
First( L ): i(
First( (*L) ): (
First( i ): i
First( F ): (i
First( S ): i(
```

Создана последовательность С:

```
I0 { П .S,$; S .F=L,$; S .L,$; F .(*L),=; F .i,=; L .F,$; F .
(*L),$; F .i,$ }
I1 { F i.,=; F i.,$ }
I2 { F (. *L),=; F (. *L),$ }
I3 { F (*.L),=; F (*.L),$; L .F,); F .(*L),); F .i,) }
I4 { П S.,$ }
I5 { S L.,$ }
I6 { S F.=L,$; L F.,$ }
I7 { F i.,) }
I8 { S F=.L,$; L .F,$; F .(*L),$; F .i,$ }
I9 { F (. *L),) }
I10 { F (*.L),); L .F,); F .(*L),); F .i,) }
I11 { F (*L.),=; F (*L.),$ }
I12 { L F.,) }
I13 { F i.,$ }
I14 { F (*L).,=; F (*L).,$ }
I15 { F (. *L),$ }
I16 { F (*.L),$; L .F,); F .(*L),); F .i,) }
I17 { S F=L.,$ }
I18 { L F.,$ }
I19 { F (*L.),) }
I20 { F (*L).,) }
I21 { F (*L.),$ }
I22 { F (*L).,$ }
```

Создана ACTION таблица

```
ACTION[14, =] = r F (*L)
ACTION[3, i] = s 7
ACTION[2, *] = s 3
ACTION[1, $] = r F i
ACTION[0, i] = s 1
ACTION[7, )] = r F i
ACTION[6, =] = s 8
ACTION[5, $] = r S L
ACTION[4, $] = a
ACTION[11, )] = s 14
ACTION[10, i] = s 7
ACTION[9, *] = s 10
ACTION[8, (] = s 15
ACTION[15, *] = s 16
ACTION[14, $] = r F (*L)
ACTION[13, $] = r F i
ACTION[12, )] = r L F
ACTION[19, )] = s 20
ACTION[1, =] = r F i
ACTION[17, $] = r S F=L
ACTION[16, i] = s 7
ACTION[22, $] = r F (*L)
ACTION[21, )] = s 22
```

```
ACTION[20, )] = r F (*L)
ACTION[10, (] = s 9
ACTION[0, (] = s 2
ACTION[18, $] = r L F
ACTION[8, i] = s 13
ACTION[6, $] = r L F
ACTION[3, (] = s 9
ACTION[16, (] = s 9
```

Создана GOTO таблица

```
GOTO[8, F] = 18
GOTO[0, S] = 4
GOTO[10, L] = 19
GOTO[16, F] = 12
GOTO[8, L] = 17
GOTO[10, F] = 12
GOTO[0, F] = 6
GOTO[3, L] = 11
GOTO[16, L] = 21
GOTO[0, L] = 5
GOTO[3, F] = 12
```

### Распознавание цепочек

Введите строку:

(\*( \*i ))=( \*i )

Введена строка: (\*( \*i ))=( \*i )\$

Процесс вывода:

```
F->i
L->F
F->(*L)
L->F
F->(*L)
F->i
L->F
F->(*L)
L->F
S->F=L
Строка допущена
```

Введите строку:

i=i

Введена строка: i=i\$

Процесс вывода:

```
F->i
F->i
L->F
S->F=L
Строка допущена
```

Введите строку:

$(*(*(i)))$

Введена строка:  $(*(*(i)))\$$

Процесс вывода:

$F \rightarrow i$

$L \rightarrow F$

$F \rightarrow (*L)$

$L \rightarrow F$

$F \rightarrow (*L)$

$L \rightarrow F$

$F \rightarrow (*L)$

$L \rightarrow F$

$S \rightarrow L$

Строка допущена