# Синтаксический анализ, часть II
# Нисходящий LL(1) разбор

Кирилл Юхин

МФТИ, 17 октября 2018 года

# Recap

- Грамматика $\langle N, \Sigma, P, S \rangle$
- Продукции $(P)$
  - Контекстно-зависимая: $(\Sigma \cup N)^* N (\Sigma \cup N)^* \to (\Sigma \cup N)^*$
  - Контекстно-свободная: $N \to (\Sigma \cup N)^*$
- Применение продукции
  $x \underset{G}{\Rightarrow} y \iff \exists u, v, p, q \in (\Sigma \cup N)^* : (x = upv) \wedge (p \to q \in P) \wedge (y = uq$
- Нисходящий и восходящий разборы
- Сентенциальная форма — последовательность символов (терминалов и нетерминалов), выводимых из начального символа.
  Элемент множества $\left\{ w \in (\Sigma \cup N)^* \mid S \overset{*}{\underset{G}{\Rightarrow}} w \right\}$
- Язык, порождаемый грамматикой $L(G) = \left\{ w \in \Sigma^* \mid S \overset{*}{\underset{G}{\Rightarrow}} w \right\}$

# Алгоритм рекурсивного спуска

```
void A() {
  // Select A-production A→X1X2 ... Xk
  for (i = 1; i <= k; ++i) {
    if (Xi - non terminal)
      Xi();
    else if (Xi equal to current token a)
      goto next input token;
    else
      report error;
  }
}
```

# Дерево вывода, сопоставления

- $[r]X \to \gamma$ - продукция $r$ отображает нетерминал $X$ на строку $\gamma$
- $\omega : \gamma$ - строка терминалов $\omega$ сопоставлена со строкой $\gamma$

$$\frac{}{\epsilon : \epsilon}\ P_1$$

$$\frac{\omega_1 : \gamma_1 \qquad \omega_2 : \gamma_2}{\omega_1\omega_2 : \gamma_1\gamma_2}\ P_2$$

$$\frac{}{a : a}\ P_3$$

$$[r]X \to \gamma$$
$$\frac{\omega : \gamma}{\omega : X}\ P_4(r)$$

# Пример дерева вывода

$$\dfrac{\dfrac{}{[\ :\ [}\ P_3 \quad \dfrac{\dfrac{\dfrac{}{\epsilon : \epsilon}\ P_1}{\epsilon : S}\ P_4(\mathsf{emp}) \quad \dfrac{}{]\ :\ ]}\ P_3}{[\ ] : [\ S\ ]}\ P_2^2}{[\ ] : S}\ P_4(\mathsf{pars}) \quad \dfrac{\vdots}{[\ ] : S}\ P_2}{[\ ]\ [\ ] : S\ S}\ P_4(\mathsf{dup}) \quad \dfrac{}{]\ :\ ]}\ P_3}{[\ [\ ]\ [\ ]\ ] : [\ S\ ]}\ P_4(\mathsf{pars})}{[\ [\ ]\ [\ ]\ ] : S}$$

# Рекурсивный спуск

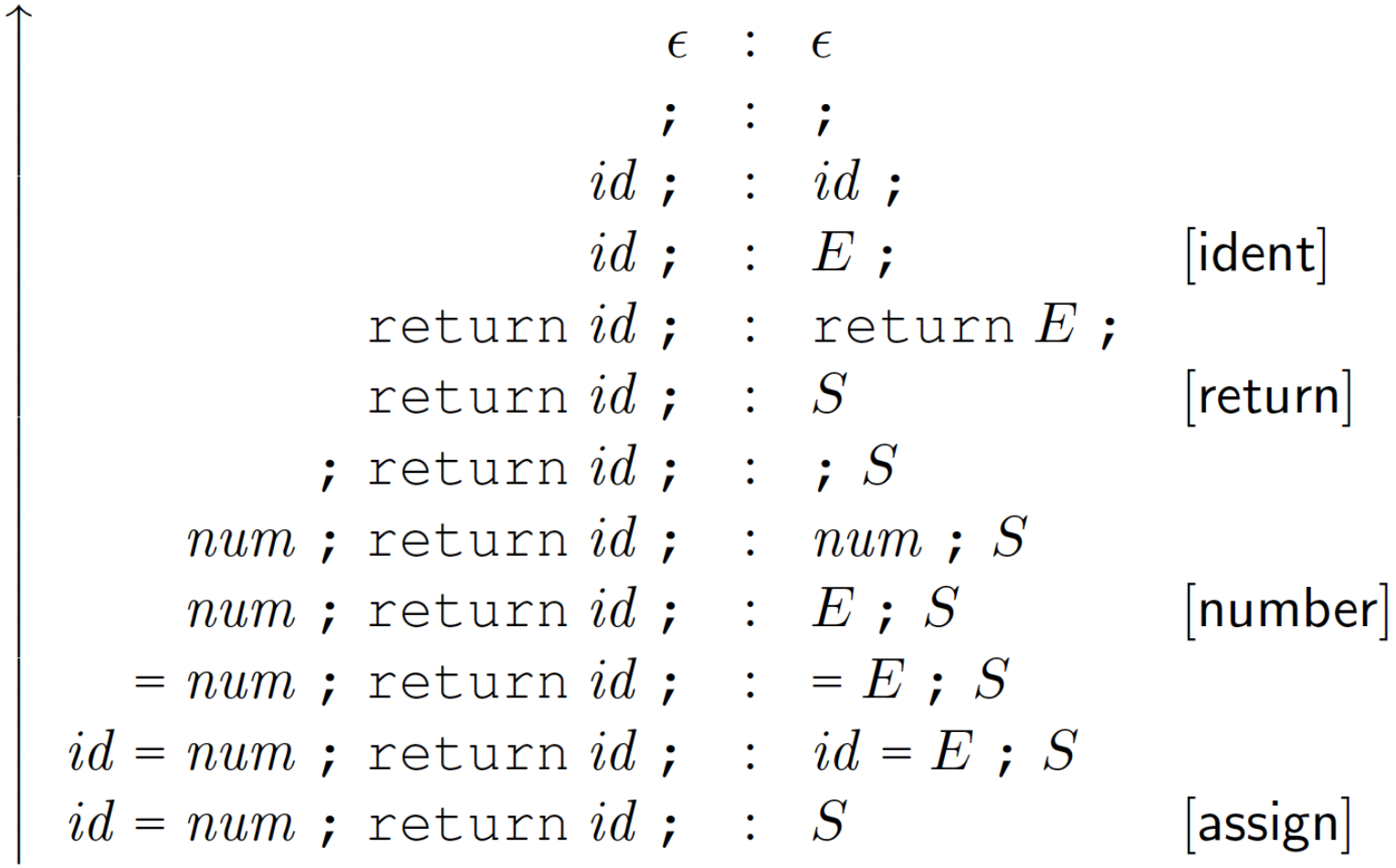
$$\frac{}{\epsilon : \epsilon}\ R_1$$

$$\frac{a\omega : a\gamma}{\omega : \gamma_2}\ R_2$$

$$[r]X \to \beta$$

$$\frac{\omega : \beta\gamma}{\omega : X\gamma}\ R_3(r)$$

# Пример рекурсивного спуска



# Предиктивный анализатор

- Попытаемся построить синтаксический анализатор без откатов.
- Для каждого входного символа $a$ и продукций $A \to \alpha_1 \dots \alpha_n$ мы должны однозначно указать какая из них порождает строку начинающуюся с $a$.
- Управляющие конструкции языков программирования как правило позволяют это сделать:

$$
\begin{aligned}
\mathit{stmt} \to\ & \textbf{if}\ \mathit{expr}\ \textbf{then}\ \mathit{stmt}\ \textbf{else}\ \mathit{stmt} \\
\mid\ & \textbf{while}\ \mathit{expr}\ \textbf{do}\ \mathit{stmt} \\
\mid\ & \textbf{begin}\ \mathit{stmt\_list}\ \textbf{end}
\end{aligned}
$$

# Основные понятия LL(1)

- $first(\beta, a)$ - терминал $a$ может первым в слове, выведенном из $\beta$
- $null(\beta)$ - из $\beta$ можно вывести пустую строку

$$\frac{}{first(a\beta, a)}\ F_1 \qquad \frac{first(X, a)}{first(X\beta, a)}\ F_2 \qquad \frac{null(X) \qquad first(\beta, a)}{first(X\beta, a)}\ F_3$$

$$\frac{[r]X \to \gamma \qquad first(\gamma, a)}{first(X, a)}\ F_4(r)$$

$$\frac{}{null(\epsilon)}\ N_1 \qquad \frac{null(X) \qquad null(\beta)}{null(X, \beta)}\ N_2 \qquad \frac{[r]X \to \gamma \qquad null(\gamma)}{null(X)}\ N_3(r)$$

# Диаграмма переходов

- Можно представить план предиктивного синтаксического анализатора в виде диаграммы переходов.
- Для каждого нетерминала - отдельная диаграмма переходов.
- Дуги двух типов:
  1. Терминалы - поглощение входного токена
  2. Нетерминалы - вызов процедуры нетерминала
- Построение:
  1. Для каждого нетерминала строим начальное и конечное состояние
  2. Для каждой продукции нетерминала $A \to A_1 \dots A_n$ строим путь из начального состояния в конечное через дуги $A_1 \dots A_n$

# Диаграмма переходов



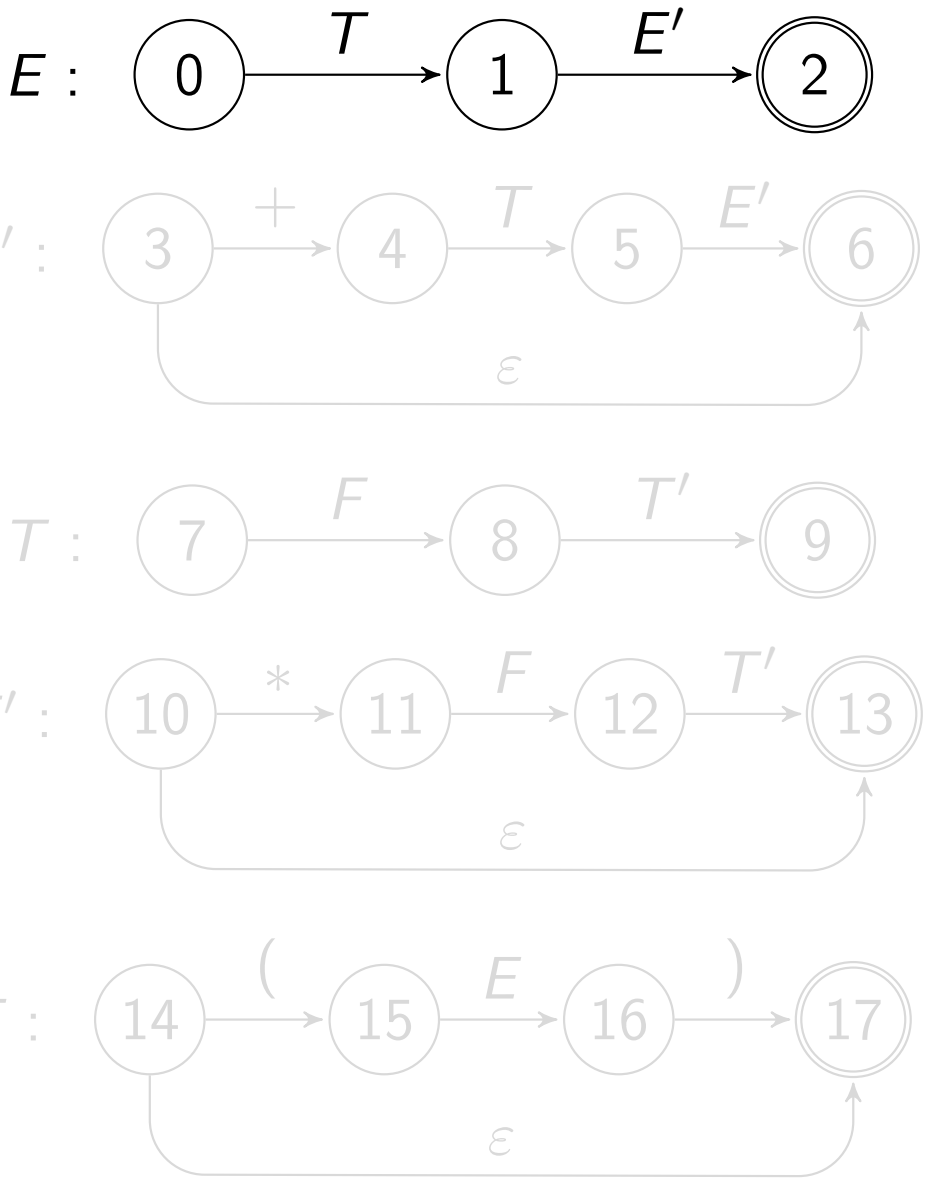

$$E \to E + T \mid T$$
$$T \to T * F \mid F$$
$$F \to (E) \mid \mathbf{id}$$

# Диаграмма переходов




$$E \to TE'$$
$$E' \to +TE' \mid \varepsilon$$
$$T \to FT'$$
$$T' \to *FT' \mid \varepsilon$$
$$F \to (E) \mid \mathbf{id}$$

# Диаграмма переходов




$$E \to TE'$$

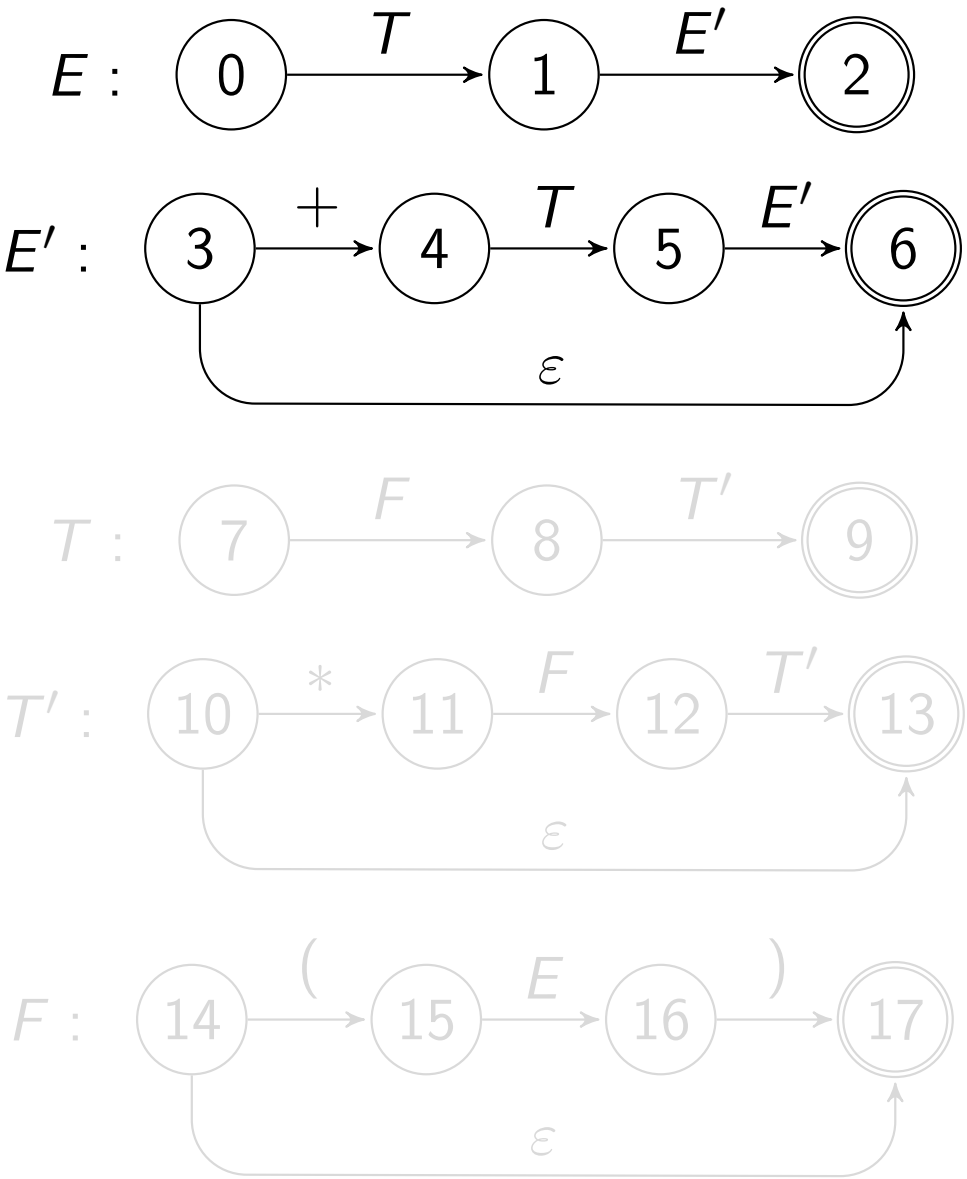
$$E' \to +TE' \mid \varepsilon$$

$$T \to FT'$$

$$T' \to *FT' \mid \varepsilon$$

$$F \to (E) \mid \mathbf{id}$$

# Диаграмма переходов




$$E \to TE'$$

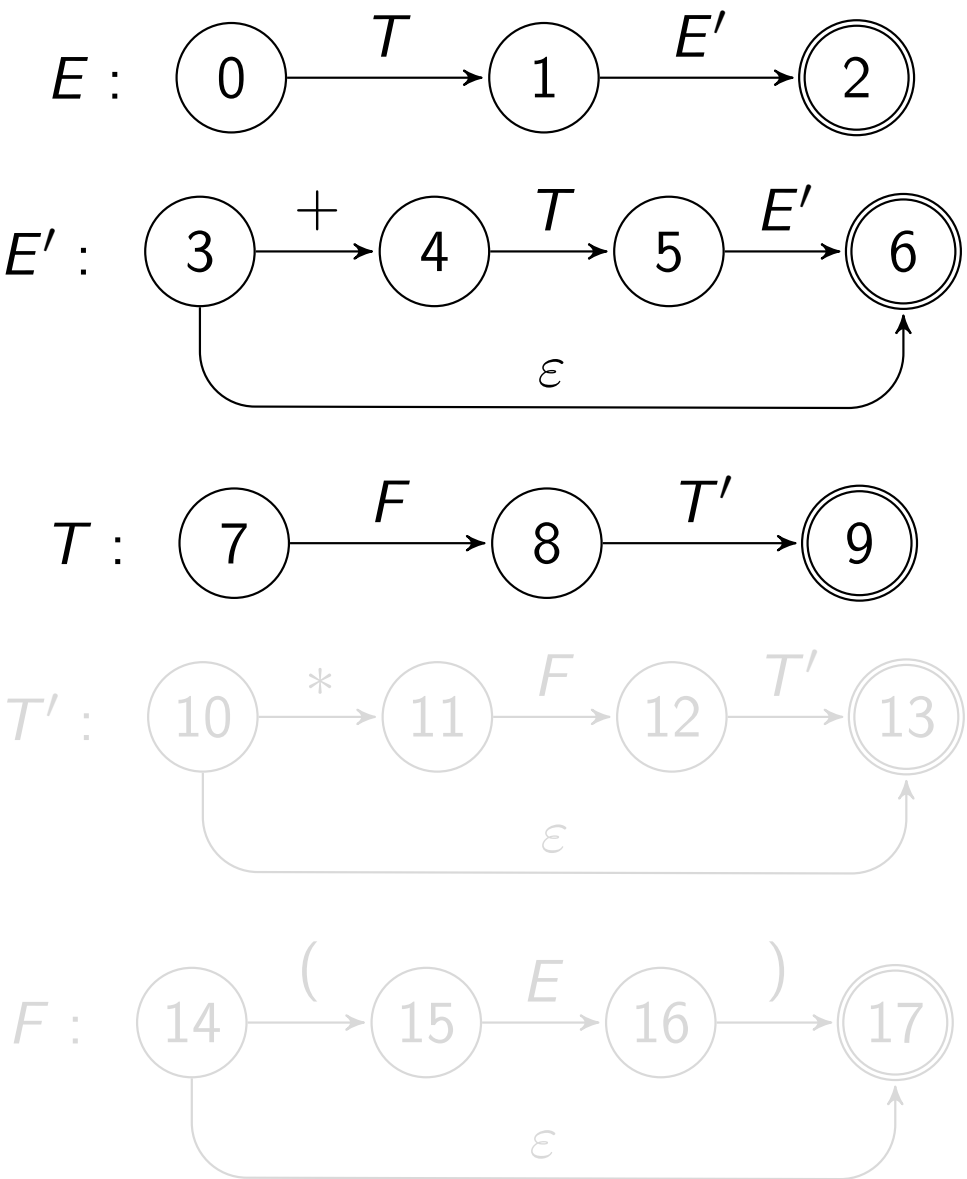
$$E' \to +TE' \mid \varepsilon$$

$$T \to FT'$$

$$T' \to *FT' \mid \varepsilon$$

$$F \to (E) \mid \mathbf{id}$$

# Диаграмма переходов




$$E \to TE'$$

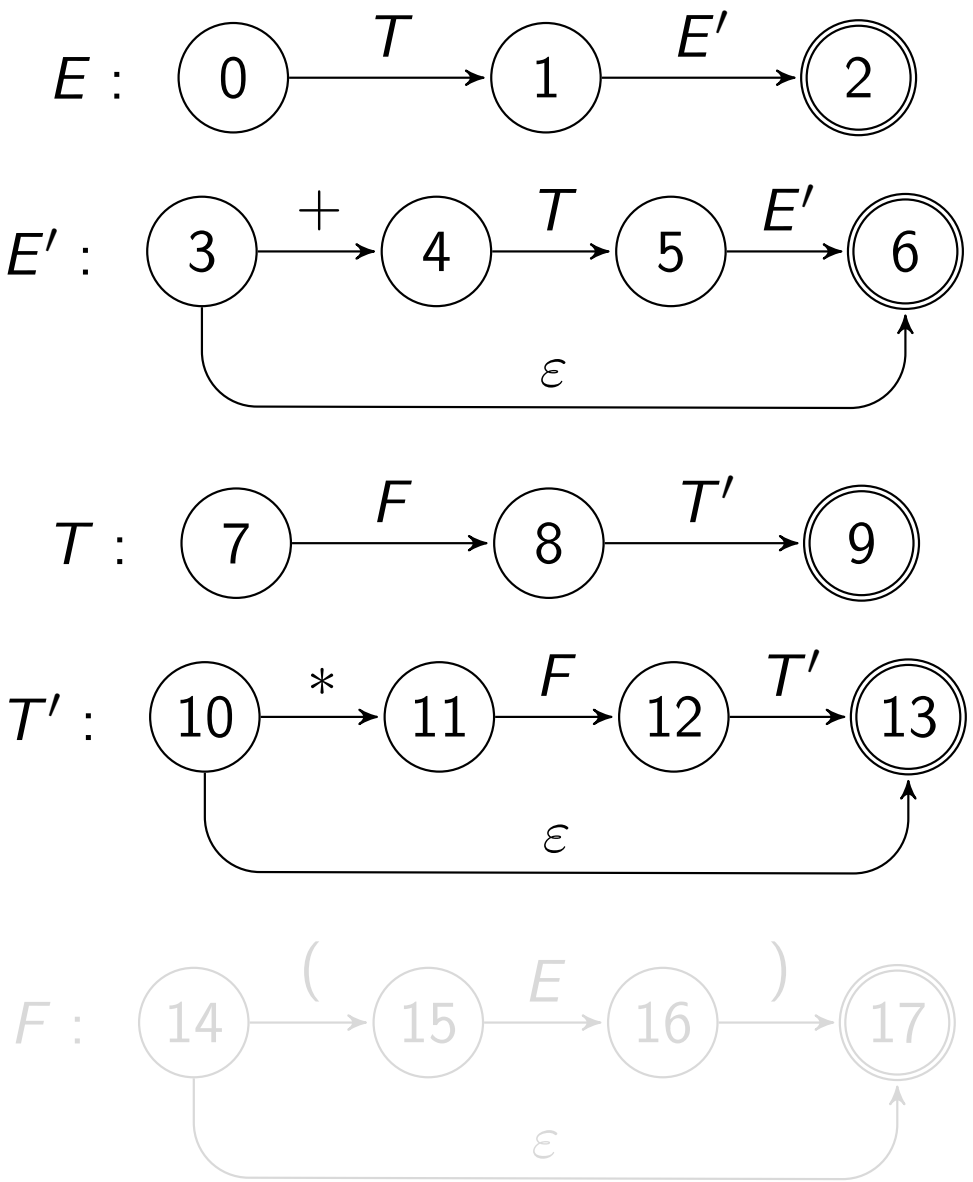
$$E' \to +TE' \mid \varepsilon$$

$$T \to FT'$$

$$T' \to *FT' \mid \varepsilon$$

$$F \to (E) \mid \mathbf{id}$$

# Диаграмма переходов




$E \to TE'$

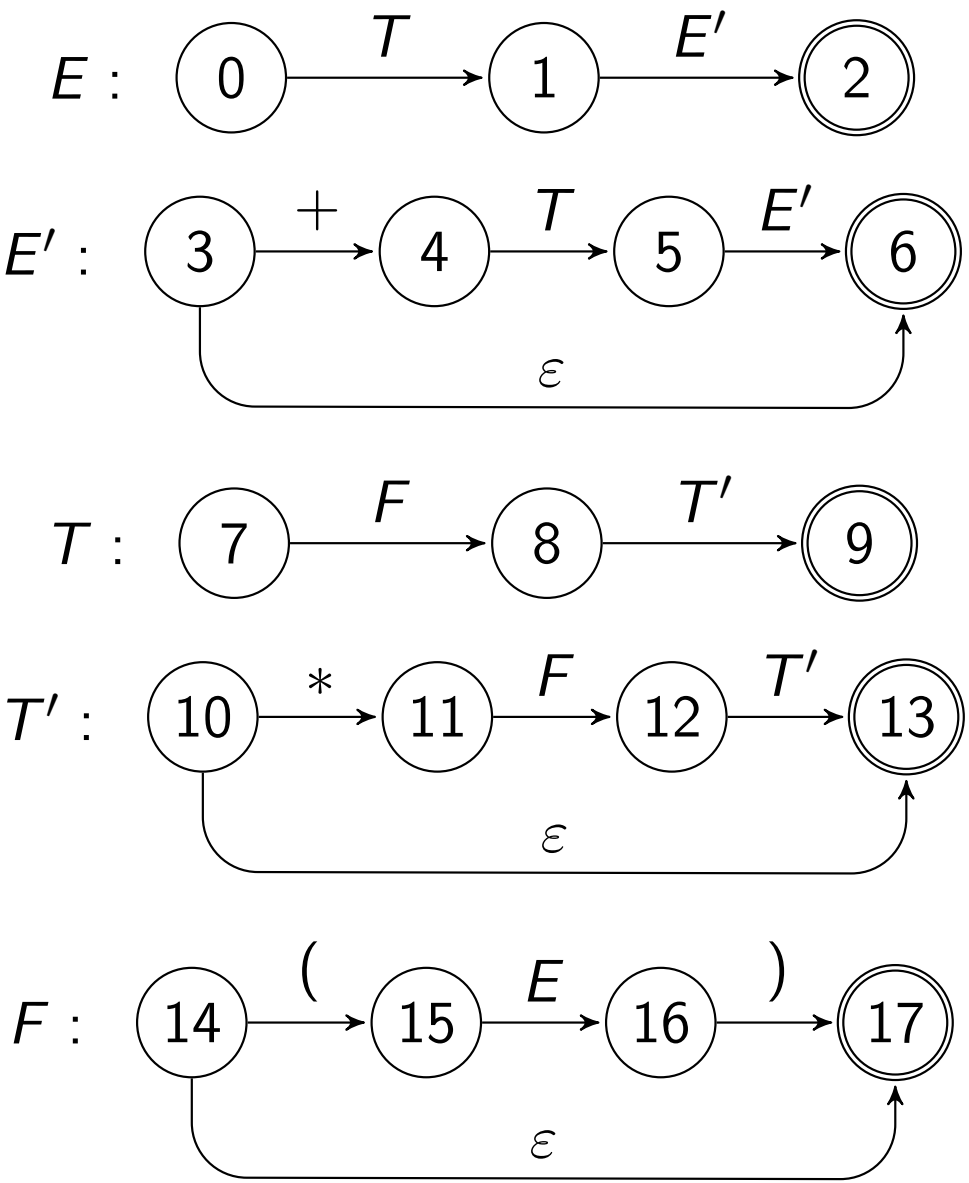
$E' \to +TE' \mid \varepsilon$

$T \to FT'$

$T' \to *FT' \mid \varepsilon$

$F \to (E) \mid \mathbf{id}$

# Диаграмма переходов




$$E \to TE'$$

$$E' \to +TE' \mid \varepsilon$$

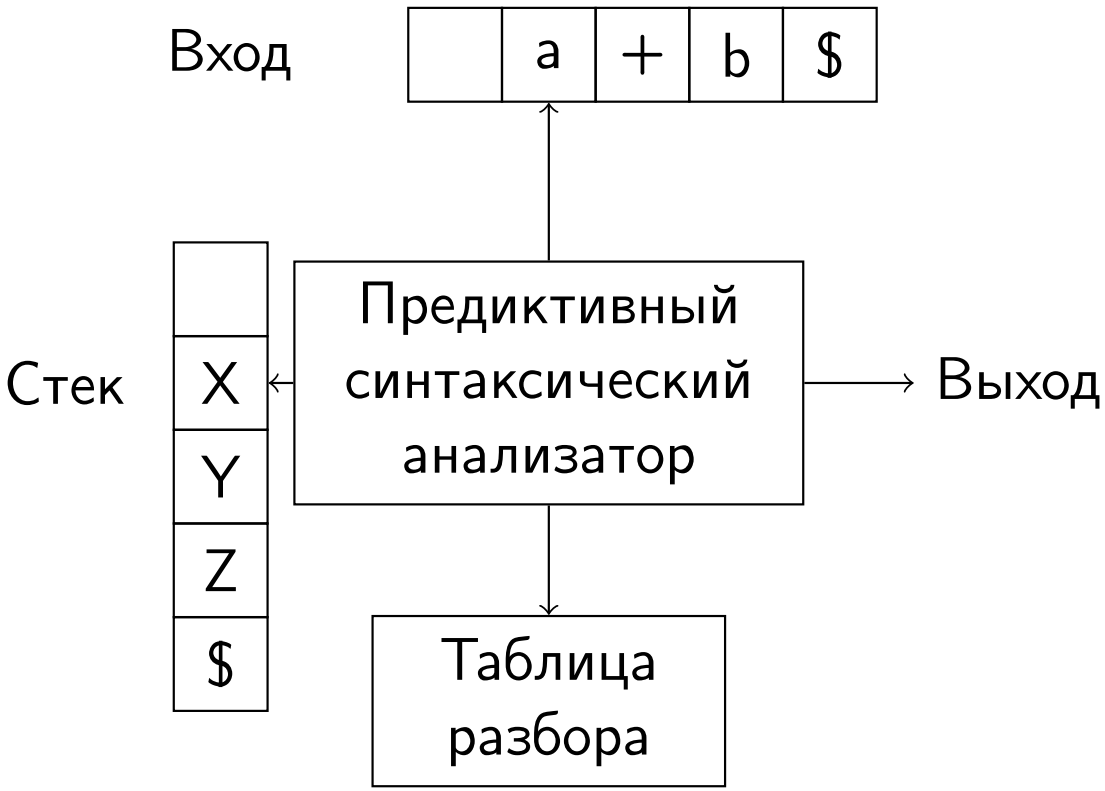
$$T \to FT'$$

$$T' \to *FT' \mid \varepsilon$$

$$F \to (E) \mid \mathbf{id}$$

# Нерекурсивный предиктивный анализ

- Неявное использование стека (в рекурсивном случае) можно заменить на явное.
- Предсказание следующей продукции по текущему нетерминалу и входному символу реализуем в виде поиска в таблице разбора



# Алгоритм

Установить указатель на ip на первый символ w$
**repeat**
    Обозначим X - символ на вершине стека
    a - символ, на который указывает ip
    **if** $X \in \{T, \$\}$ **then**
        **if** $X = a$ **then**
            Снять со стека X и переместить ip
        **else**
            Ошибка
        **end if**
    **else**
        **if** $M[X, a] = X \to Y_1 \dots Y_k$ **then**
            Снять X со стека
            Поместить в стек $Y_k \dots Y_1$
        **else**
            Ошибка
        **end if**
    **end if**
**until** $X = \$$

# Пример таблицы

| Нетерминал | Входной символ | | | | | |
|---|---|---|---|---|---|---|
| | **id** | $+$ | $*$ | $($ | $)$ | \$ |
| $E$ | $E \to TE'$ | | | $E \to TE'$ | | |
| $E'$ | | $E' \to +TE'$ | | | $E' \to \varepsilon$ | $E' \to \varepsilon$ |
| $T$ | $T \to FT'$ | | | $T \to FT'$ | | |
| $T'$ | | $T' \to \varepsilon$ | $T' \to *FT'$ | | $T' \to \varepsilon$ | $T' \to \varepsilon$ |
| $F$ | $F \to \mathbf{id}$ | | | $F \to (E)$ | | |

# Множества FIRST и FOLLOW

**FIRST** Если $\alpha$ - произвольная строка *символов* грамматики, то определим $FIRST(\alpha)$ как множество терминалов, с которых начинаются строки, выводимые из $\alpha$

- Если $\alpha \xrightarrow{*} \varepsilon$, то $\varepsilon \in FIRST(\alpha)$

**FOLLOW** Определим множество $FOLLOW(A)$ для нетерминала $A$, как множество терминалов $a$, которые могут располагаться непосредственно справа от $A$ в некоторой сентенциальной форме[1].

- $FOLLOW(A) = \{a \in T \mid \exists \alpha, \beta : \exists S \xrightarrow{*} \alpha A a \beta\}$
- Если $A$ может оказаться крайним справа символом некоторой сентенциальной формы, то $\$ \in FOLLOW(A)$

[1] Сентенциальная форма — последовательность символов (терминалов и нетерминалов), выводимых из начального символа

# Построение FIRST

- Если $X$ - терминал, то $FIRST(X) = X$.
- Если имеется продукция $X \to \varepsilon$, добавим $\varepsilon$ к $FIRST(X)$.
- Если $X$ - нетерминал и имеется продукция $X \to Y_1 \dots Y_k$, то поместим $a$ в $FIRST(X)$, если для некоторого $i$ $a \in FIRST(Y_i)$ и $\varepsilon \in \{FIRST(Y_1) \cup \dots \cup FIRST(Y_{i-1})\}$, т.е. $Y_1 \dots Y_{i-1} \stackrel{*}{\to} \varepsilon$.
- Если $\varepsilon \in \{Y_1 \cap \dots \cap Y_k\}$, то добавляем $\varepsilon$ в $FIRST(X)$.

# Построение FOLLOW

- Поместим \$ в $FOLLOW(S)$, где $S$ - стартовый символ, а \$ - маркер конца строки.
- Если имеется продукция $A \to \alpha B \beta$, то все элементы $FIRST(\beta)$, кроме $\varepsilon$, помещаются в множество $FOLLOW(B)$.
- Если имеется продукция $A \to \alpha B$ или $A \to \alpha B \beta$, где $FIRST(\beta)$ содержит $\varepsilon$(т.е. $\beta \overset{*}{\to} \varepsilon$), то все элементы из множества $FOLLOW(A)$, помещаются в множество $FOLLOW(B)$.

# Построение таблицы переходов

- Строим FIRST и FOLLOW
- Для каждой продукции грамматики $A \to \alpha$:
  - Для каждого терминала $a$ из $FIRST(\alpha)$ добавляем $A \to \alpha$ в ячейку $M[A, a]$.
  - Если $\varepsilon \in FIRST(\alpha)$, то для каждого терминала $b$ из $FOLLOW(A)$ добавим $A \to \alpha$ в ячейку $M[A, b]$.
  - Если $\varepsilon \in FIRST(\alpha)$, а \$ $\in FOLLOW(A)$, добавим $A \to \alpha$ в ячейку $M[A, \ ]$.
- Пустые ячейки делаем указывающими на ошибку.

# LL(1) грамматики

- Построенная таблица может иметь несколько записей в одной ячейке.
  - Левая рекурсия
  - Неоднозначность
- Грамматика, таблица которой не имеет множественных записей называется LL(1)
  - L - слева-направо
  - L - левое порождение
  - 1 - один символ для предсказания