



НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ  
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»  
Кафедра «Комп'ютерної інженерії та програмування»

## ФОРМАЛЬНІ МОВИ, ГРАМАТИКИ І АВТОМАТИ

Лекція 9

### ВИСХІДНІ LR(K) – РОЗПІЗНАВАЧІ

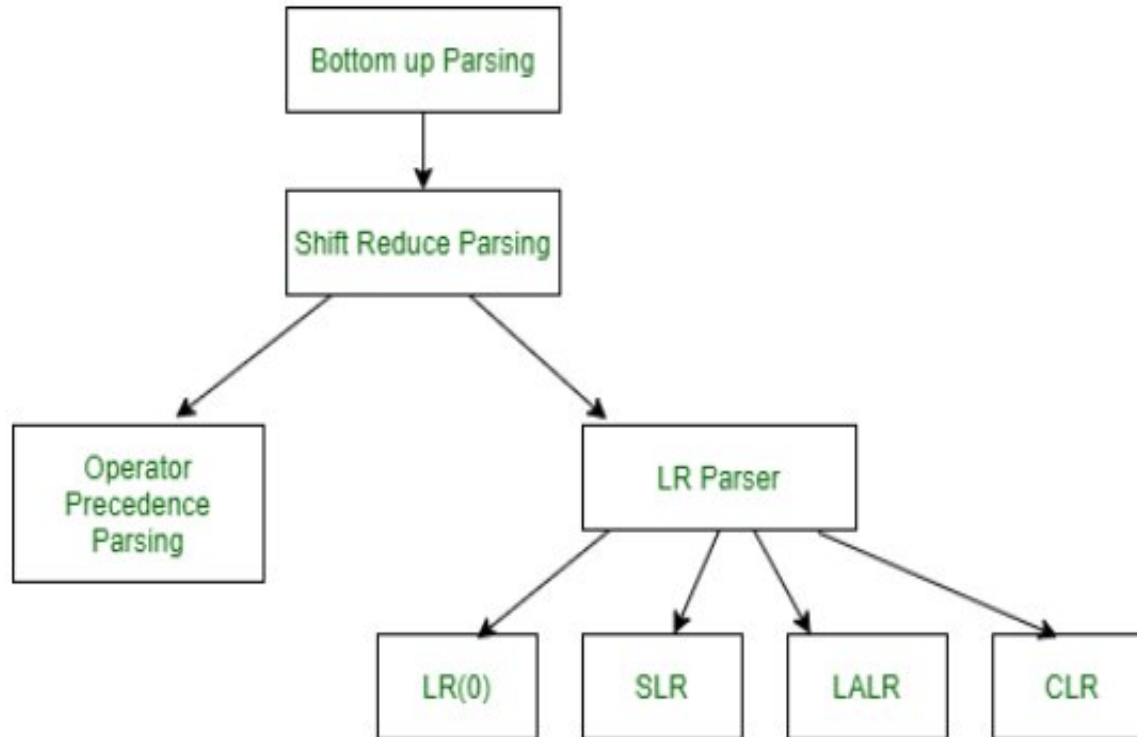


Гавриленко Світлана Юріївна  
+380664088551 (Viber)  
+380632864663 (Telegram)  
Gavrilenko08@gmail.com  
306BK

# LR(k) – граматики

Розпізнавачі можна розділити на дві категорій: спадні і висхідні. В основі роботи висхідних розпізнавачів лежить операція **згортки** – заміна правої частини правила граматки на відповідний лівий нетермінал. Висхідні аналізатори використовують нижні правила раніш тих, що розташовані вище. Детерміновані висхідні розпізнавачі, так само як і спадні, можуть бути побудовані не для всякої КВ-граматики, а тільки для визначених підкласів таких граматик. Найбільш поширеним підкласом КВ-граматик є LR(k)-граматики. Ці граматки забезпечують розпізнавання ланцюжка при перегляді зліва направо, про що свідчить буква L (Left) у назві граматки, і дозволяють виконати **правобічну згортку**, що показує буква R (Right) у назві. Параметр k свідчить про те, що для визначення правила граматки, яке потрібно застосувати для згортання ланцюжка, буде потрібно переглянути не більш ніж  $k$  ще не прочитаних символів вхідного ланцюжка. На практиці найчастіше використовують підкласи LR(k)-граматик: LR(0) та LR(1)-граматики, що дозволяють відносно просто виконувати побудову висхідних розпізнавачів

# Типи of LR-розпізнавачів (Parsers). Частина 1



Типи LR парсерів:

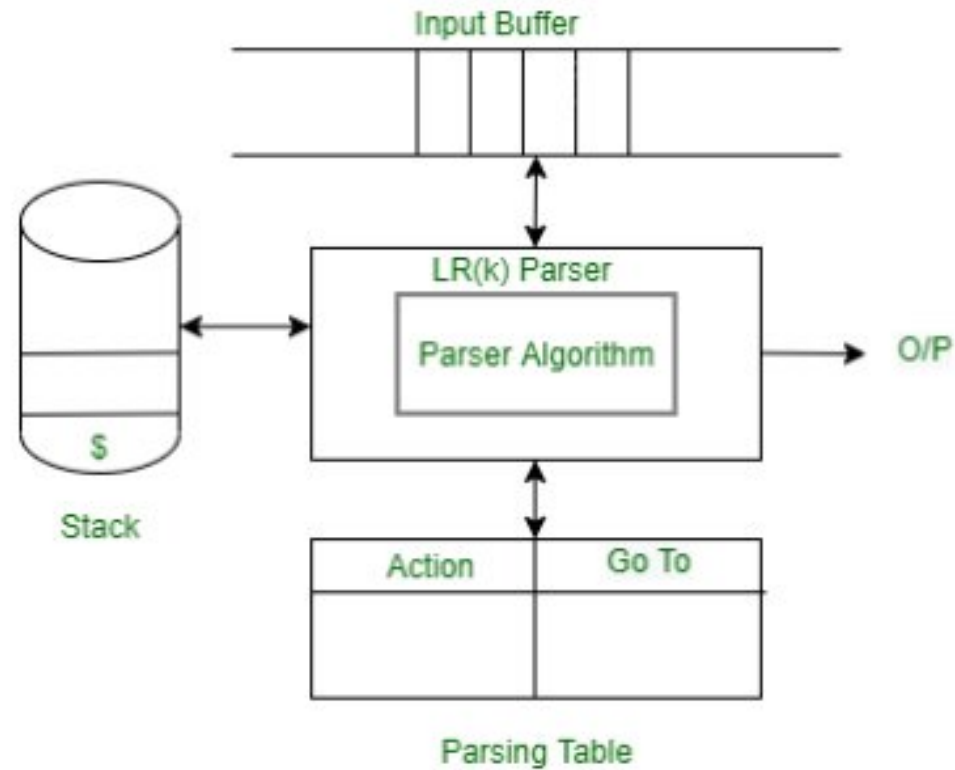
1. LR(0) Parser
2. Simple LR-Parser (SLR)
3. LALR (Look-Ahead LR) Parser.
4. Canonical LR Parser (CLR) or LR(1) parser
5. Operator Precedence parsing

# Типи of LR-розпізнавачів (Parsers). Частина 2

Основні типи LR-парсерів:

- LR(0) парсер: Найпростіший тип LR-парсерів. Не аналізує наступний символ у вхідному потоці при прийнятті рішень. Часто має конфлікти, що обмежує їх застосування.
- SLR (Simple LR) парсер: Є розширенням LR(0) парсерів. Аналізує наступний символ у вхідному потоці при прийнятті рішень. Більш потужний, ніж LR(0) парсер, але все ще має обмеження застосування.
- LALR (Look-Ahead LR) парсер: Поєднує в собі простоту SLR парсерів та потужність LR(0) парсерів. Використовує обмежений аналіз наступних символів у вхідному потоці при прийнятті рішень для розв'язання більшості конфліктів. Широко використовуються на практиці завдяки своїй ефективності та можливостям.
- Канонічний аналізатор LR або LR(1): аналізатор, який використовує таблиці переходів та керуючу таблицю для визначення наступної дії на основі поточного стану та вхідного символу. Це потужний інструмент для розбору контекстно-вільних граматик.

# LR-розпізнавач (parser)



# Граматичні символи та граматичні входження. Частина 1

**Граматичні символи** – це символи повного словника граматики ( $V_T \cup V_A$ ). Кожен граматичний символ може входити в різні правила граматики і, більш того, з'являтися в тому самому правилі кілька разів. При цьому положення символу в правилі граматики може показувати, яку дію потрібно виконати – перенос чи згортку, а також, які граматичні символи можуть за ним впливати. Для зручності подальших міркувань визначимо поняття граматичного входження.

**Граматичне входження** символу граматики задається номером правила і номером позиції, що вказує місце символу в правій частині правила, вважаючи, що самий лівий символ правої частини правила є першим символом. Умовимося позначати граматичні входження символів, що входять у праву частину правила тільки один раз, за допомогою одного індексу, що співпадає з номером правила. Крім того, кожна граMATика містить граматичне входження  **$I_0$** , яке є початковим символом граматики.

# Граматичні символи та граматичні входження. Частина 2

Маємо правила граматики:

$$1. I \rightarrow A b c A$$

$$2. A \rightarrow d$$

Перетворимо граматику використовуючи граматичні входження

$$1. I_0 \rightarrow A_{11} b_{12} c_{13} A_{14}$$

$$2. A \rightarrow d_{21}$$

Спростимо правила граматики таким чином: якщо граматичні входження використовуються в одному правилі граматики один раз, то другий індекс можемо не ставити. Якщо граматичні символи використовуються один раз у всіх правилах граматики, то індекс можемо використовувати або не використовувати.

$$1. I_0 \rightarrow A_{11} b_1 c_1 A_{14} \quad \text{або} \quad I_0 \rightarrow A_{11} b c A_{14}$$

$$2. A \rightarrow d_2 \quad \text{або} \quad A \rightarrow d$$

# Функції ВПЕРШ (Y) та ВПІСЛЯ (Y)

Функція ВПЕРШ(Y) за аналогією з функцією ПЕРШ(Y) визначає множину граматичних входжень, що можуть стояти **на першому місці в ланцюжках**, виведених з Y. Ця множина будується в такий спосіб: у нього входить символ Y і всі символи, що починають проміжні ланцюжки, виведені з Y **без застосування правил, що анулюють**.

Функція ВПІСЛЯ(Y) є аналогом функції СЛІД (Y). Вона визначає множину граматичних входжень, що можуть зустрічатися **безпосередньо після Y** у ланцюжках, виведених з початкового символу граматики. Правило визначення функції ВПІСЛЯ (Y) можна записати так: якщо в правій частині деякого правила після Y безпосередньо слідує Z, то

$$\text{ВПІСЛЯ}(Y) = \text{ВПЕРШ}(Z).$$

При побудові розпізнавачів необхідно враховувати наявність **маркера дна  $h_0$** , для якого сформулюємо ще одне правило визначення цієї функції: якщо Y є маркером дна магазину, то

$$\text{ВПІСЛЯ}(h_0) = \text{ВПЕРШ}(I_0),$$

де  $I_0$  — початкове входження.



# Приклад 9.1. Для заданої граматики $\Gamma_{9.1}$ побудувати функції ВПЕРШ (Y) та ВПІСЛЯ (Y)

$\Gamma_{9.1}$ : 1.  $I \rightarrow a I I b$ , 2.  $I \rightarrow c$

ГраMATика з граматичними входженнями:

1.  $I_0 \Rightarrow a_1 I_{12} I_{13} b_1 c_2$ , 2.  $I \rightarrow c_2$   
 $\text{ВПЕРШ}(a_1) = \{a_1\}$ ,  
 $\text{ВПЕРШ}(I_{12}) = \{I_{12}, a_1, c_2\}$ ,  
 $\text{ВПЕРШ}(I_{13}) = \{I_{13}, a_1, c_2\}$ ,  
 $\text{ВПЕРШ}(b_1) = \{b_1\}$ ,  
 $\text{ВПЕРШ}(c_2) = \{c_2\}$ ,  
 $\text{ВПЕРШ}(I_0) = \{I_0, a_1, c_2\}$ .

$\text{ВПІСЛЯ}(a_1) = \text{ВПЕРШ}(I_{12}) = \{I_{12}, a_1, c_2\}$ ,

$\text{ВПІСЛЯ}(I_{12}) = \text{ВПЕРШ}(I_{13}) = \{I_{13}, a_1, c_2\}$ ,

$\text{ВПІСЛЯ}(I_{13}) = \{b_1\}$ ,

$\text{ВПІСЛЯ}(b_1) = \$$ ,

$\text{ВПІСЛЯ}(c_2) = \$$ ,

$\text{ВПІСЛЯ}(I_0) = \$$ ,

$\text{ВПІСЛЯ}(h_0) = \text{ВПЕРШ}(I_0) = \{I_0, a_1, c_2\}$ .

# Побудова таблиць розпізнавача. Таблиця переходів (GO TO TABLE)

Таблиця переходів будується наступним чином. Кожному граматичному входженню відповідає один рядок таблиці, а кожному граматичному символу – один стовпець. Клітини таблиці заповнюються елементами функцій ВПІСЛЯ таким чином, що елемент  $X_k \in \text{ВПІСЛЯ}(Y_j)$  заноситься в клітку, що знаходиться на перетині рядка  $Y_j$  і стовпця, позначеного граматичним символом  $X$ .

## Побудова таблиці переходів для граматики $\Gamma_{7.1}$

$\text{ВПЕРШ}(a_1) = \{a_1\}$ ,  $\text{ВПЕРШ}(I_{12}) = \{I_{12}, a_1, c_2\}$ ,  $\text{ВПЕРШ}(I_{13}) = \{I_{13}, a_1, c_2\}$ ,  $\text{ВПЕРШ}(b_1) = \{b_1\}$ ,  $\text{ВПЕРШ}(c_2) = \{c_2\}$ ,  $\text{ВПЕРШ}(I_0) = \{I_0, a_1, c_2\}$ .

$\text{ВПІСЛЯ}(a_1) = \{I_{12}, a_1, c_2\}$ ,  $\text{ВПІСЛЯ}(I_{12}) = \{I_{13}, a_1, c_2\}$ ,  $\text{ВПІСЛЯ}(I_{13}) = \{b_1\}$ ,  $\text{ВПІСЛЯ}(b_1) = \{\$ \}$ ,  $\text{ВПІСЛЯ}(c_2) = \{\$ \}$ ,  $\text{ВПІСЛЯ}(I_0) = \{\$ \}$ ,  $\text{ВПІСЛЯ}(h_0) = \{I_0, a_1, c_2\}$ .

Граматичні входження	Граматичні символи			
	$a$	$b$	$c$	$I$
$a_1$	$a_1$		$c_2$	$I_{12}$
$I_{12}$	$a_1$		$c_2$	$I_{13}$
$I_{13}$		$b_1$		
$b_1$				
$c_2$				
$h_0$	$a_1$		$c_2$	$I_0$
$I_0$				

# Керуюча таблиця (Action Table)

У цій таблиці позначимо дію переносу символів із вхідного ланцюжка в магазин символом **П** (перенос), а дії, зв'язані зі згорткою ланцюжків, що відповідають правим частинам правил, позначимо символом **З (№)**, де № – номер використаного правила. Для позначення дій, що здійснюють передачу на вихід результатів роботи розпізнавача, умовимося використовувати початкові букви слів “допустити” (**Д**) і “відкинути” (**В**).

Позначимо рядки таблиці граматичними входженнями, а стовпці – термінальними символами граматики і символом кінця ланцюжка  $\perp_k = \$ = \varepsilon$ .

Підставою для заповнення таблиці є наступні два положення.

1. Операція згортки **З(№)** повинна виконуватися незалежно від вхідного символу завжди, якщо у вершині магазину знаходиться саме праве граматичне входження деякого правила. Для таких граматичних входжень значення функції **ВПІСЛЯ** є порожньою множиною.

2. Якщо у вершині магазину знаходиться граматичне входження, що не є самим правим входженням якого-небудь правила, то варто виконати перенос чергового символу вхідного ланцюжка в магазин.

3. Процес розпізнавання закінчується успішно при виявленні символу  $\perp_k$  на вході і граматичного входження  $I_0$  в магазині. В інших випадках, що залишилися, вхідний ланцюжок повинен бути відкинутий.

# Побудова керуючої таблиці для граматики $\Gamma_{7.1}$

$\Gamma_{9.1}$ : 1.  $I_0 \rightarrow a_1 I_{12} I_{13} b_1$ , 2.  $I \rightarrow c_2$

$\text{ВПЕРШ}(a_1) = \{a_1\}$ ,  $\text{ВПЕРШ}(I_{12}) = \{I_{12}, a_1, c_2\}$ ,  $\text{ВПЕРШ}(I_{13}) = \{I_{13}, a_1, c_2\}$ ,  $\text{ВПЕРШ}(b_1) = \{b_1\}$ ,  
 $\text{ВПЕРШ}(C_2) = \{c_2\}$ ,  $\text{ВПЕРШ}(I_0) = \{I_0, a_1, c_2\}$ .

$\text{ВПІСЛЯ}(a_1) = \{I_{12}, a_1, c_2\}$ ,  $\text{ВПІСЛЯ}(I_{12}) = \{I_{13}, a_1, c_2\}$ ,  $\text{ВПІСЛЯ}(I_{13}) = \{b_1\}$ ,  $\text{ВПІСЛЯ}(b_1) = \$$ ,  
 $\text{ВПІСЛЯ}(C_2) = \$$ ,  $\text{ВПІСЛЯ}(I_0) = \$$ ,  $\text{ВПІСЛЯ}(h_0) = \{I_0, a_1, c_2\}$ .

Граматичні входження	Термінальні символи			
	$a$	$b$	$c$	$\perp_k$ (кінець рядка)
$a_1$	П	П	П	В
$I_{12}$	П	П	П	В
$I_{13}$	П	П	П	В
$b_1$	3(1)	3(1)	3(1)	3(1)
$c_2$	3(2)	3(2)	3(2)	3(2)
$h_0$	П	П	П	В
$I_0$	В	В	В	Д

# Алгоритм роботи висхідного розпізнавача

1. Прочитати черговий символ вхідного ланцюжка  $x$ .
2. Прочитати символ стану, що знаходиться у вершині магазину  $U_{k_I}$ .
3. Прочитати значення елемента керуючої таблиці, що знаходиться в рядку  $U_{k_I}$  і стовпці  $x$ .
4. Якщо прочитане значення є В чи Д, то роботу варто закінчити, оскільки результат отриманий.
5. Якщо прочитане значення визначає операцію **Перенос**, то прочитати в таблиці переходів елемент  $x_{ij}$ , що знаходиться в рядку  $U_{k_I}$  і стовпці  $x$ . Записати символ  $x_{ij}$  у магазин.
6. Якщо прочитане значення визначає операцію  $Z(N_0)$  в нетермінал  $Z$  ( $Z$  - лівий ненермінальний символ правила граматики з номером  $N_0$ ), то прочитати в таблиці переходів елемент  $Z_{ij}$ , що знаходиться в стовпці  $Z$  і рядку, **який відповідає верхньому символу магазину, що не приймає участі у згортці**.

Використовуючи описаний алгоритм, роботу розпізнавача, можна подати у вигляді зміни

# Приклад роботи розпізнавача

$\Gamma_{9,1}$ : 1.  $I_0 \rightarrow a_1 I_{12} I_{13} b_1$ , 2.  $I \rightarrow c_2$

Граматичні входження	Таблиця переходів				Керуюча таблиця			
	Граматичні символи				Термінальні символи			
	$a$	$b$	$c$	$I$	$a$	$b$	$c$	$\perp_K$
$a_1$	$a_1$		$c_2$	$I_{12}$	П	П	П	В
$I_{12}$	$a_1$		$c_2$	$I_{13}$	П	П	П	В
$I_{13}$		$b_1$			П	П	П	В
$b_1$					3(1)	3(1)	3(1)	3(1)
$c_2$					3(2)	3(2)	3(2)	3(2)
$h_0$	$a_1$		$c_2$	$I_0$	П	П	П	В
$I_0$					В	В	В	Д
Вхід	Магазин	Дія		Вхід	Магазин	Дія		
$a accbcb \perp$	$h_0$	П		$bcb \perp$	$h_0 a_1 a_1 I_{12} I_{13}$	П		
$accbcb \perp$	$h_0 a_1$	П		$cb \perp$	$h_0 a_1 a_1 I_{12} I_{13} b_1$	3(1)		
$ccbcb \perp$	$h_0 a_1 a_1$	П		$cb \perp$	$h_0 a_1 I_{12}$	П		
$cbcb \perp$	$h_0 a_1 a_1 c_2$	3(2)		$b \perp$	$h_0 a_1 I_{12} c_2$	3(2)		
$cbcb \perp$	$h_0 a_1 a_1 I_{12}$	П		$b \perp$	$h_0 a_1 I_{12} I_{13}$	П		
$bcb \perp$	$h_0 a_1 a_1 I_{12} c_2$	3(2)		$\perp_K$	$h_0 a_1 I_{12} I_{13} b_1$	3(1)		
				$\perp_K$	$h_0 I_0$	Д		

# Алгоритм побудови висхідного розпізнавача

1. Побудувати правила граматики. Визначити граматичні входження. Визначити для даної граматики функції **ВПЕРШ** ( $Y$ ) та **ВПІСЛЯ** ( $Y$ ).
2. Побудувати детерміновану таблицю переходів, що має по одному стовпцю для кожного граматичного символу і по одному рядку для кожного граматичного входження та маркера дна  $h_0$ . Кожний рядок таблиці  $R_j$  заповнюється елементами, які належать функції **ВПІСЛЯ** ( $R_j$ ). При цьому, якщо  $C_K \notin \text{ВПІСЛЯ} (R_j)$ , то  $C_K$  заноситься в осередок, який знаходиться на перетині рядка  $R_j$  та стовбця  $C$ .
3. Якщо таблиця, побудована на кроці 2, виходить недетермінованою, то потрібно перетворити цю таблицю в детерміновану, розглядаючи її як недетерміновану таблицю переходів кінцевого автомата з початковим станом  $h_0$ .
4. Стани, отримані на кроці 3 (крім стану, що відповідає порожній множині), варто використовувати як магазинні символи. Отримана таблиця переходів може містити переходи в порожню множину. Такі елементи варто розуміти як заборонені і розглядати переходи в них як помилки.
5. Керуючу таблицю заповнюють рядок за рядком відповідно до множини граматичних входжень, що позначають рядки, в такий спосіб:



# Алгоритм побудови висхідного розпізнавача. Продовження

5.1. Якщо рядок позначений початковим входженням  $I_0$ , то в стовпець, що відповідає маркеру кінця рядка  $\perp_k$ , заноситься операція Д, а в усі інші стовбці – операція В.

5.2. Якщо рядок відзначений маркером дна  $h_0$ , або якщо всі граматичні входження, що входять у множину, яка позначає рядок, та не є самими правими символами у своїх правилах, то в стовпець, позначений кінцевим маркером рядка, заноситься операція В, а у всі інші стовпці – операція П.

5.3. Якщо рядок позначений граматичним входженням, що є самим правим входженням у правилі з номером  $k$ , то у всі елементи рядка заноситься операція З ( $k$ ).

5.4. Якщо множина, що позначає рядок після перетворення НКА, містить початкове входження і хоча б одне входження, відмінне від початкового, але не містить жодного самого правого входження, то в стовпець, позначений символом кінця рядка, потрібно помістити операцію Д, а в інші стовпці – П.

Наведена процедура забезпечує побудову розпізнавача, тільки у разі, якщо задана граматика належить до підкласу  **$LR(0)$** , оскільки дії в кожному рядку керуючої таблиці однакові, тобто не залежать від вхідного символу. Якщо ж у процесі побудови виявляється, що хоча б один з пунктів виконати неможливо, то це означає, що для заданої граматики неможливо побудувати  $LR(0)$ -розпізнавач і що вона не є  $LR(0)$ -

# Приклад побудови $LR(0)$ -розпізнавача для граматики $\Gamma_{9.2}$

Розглянемо побудову розпізнавача для наступної граматики  $\Gamma_{9.2}$ :

1.  $E \rightarrow E + T$
2.  $E \rightarrow T$
3.  $T \rightarrow (E)$
4.  $T \rightarrow i$

1.  $E_0 \rightarrow E_1 + T_1$
2.  $E \rightarrow T_2$
3.  $T \rightarrow (E_3)$
4.  $T \rightarrow i$ .

$$\text{ВПЕРШ}(E_1) = \{E_1, T_2, (, i\},$$

$$\text{ВПЕРШ}(T_1) = \{T_1, (, i\},$$

$$\text{ВПЕРШ}(+) = \{+\},$$

$$\text{ВПЕРШ}(T_2) = \{T_2, (, i\},$$

$$\text{ВПЕРШ}(i) = \{i\},$$

$$\text{ВПЕРШ}() = \{(\},$$

$$\text{ВПЕРШ}()) = \{)\},$$

$$\text{ВПЕРШ}(E_3) = \{E_3, E_1, T_2, (, i\},$$

$$\text{ВПЕРШ}(E_0) = \{E_0, E_1, T_2, (, i\},$$

$$\text{ВПІСЛЯ}(E_1) = \{+\},$$

$$\text{ВПІСЛЯ}(+) = \{T_1, (, i\},$$

$$\text{ВПІСЛЯ}(T_1) = \{\$\},$$

$$\text{ВПІСЛЯ}(T_2) = \{\$\},$$

$$\text{ВПІСЛЯ}(i) = \{\$\},$$

$$\text{ВПІСЛЯ}() = \{E_1, E_3, T_2, (, i\}$$

$$\text{ВПІСЛЯ}()) = \{\$\},$$

$$\text{ВПІСЛЯ}(E_0) = \{\$\},$$

$$\text{ВПІСЛЯ}(h_0) = \{E_0, E_1, T_2, (, i\},$$

$$\text{ВПІСЛЯ}(E_3) = \{)\}.$$

## Побудова таблиці переходів для граматики $\Gamma_{9.2}$

ВПІСЛЯ( $E_1$ ) = {+},

ВПІСЛЯ( $T_1$ ) = { $\$$ } ,

ВПІСЛЯ( $T_2$ ) = { $\$$ } ,

ВПІСЛЯ(+)= { $T_1$ , (,  $i$ },

ВПІСЛЯ( $i$ ) = { $\$$ } ,

ВПІСЛЯ(()= { $E_1$ ,  $E_3$ ,  $T_2$ , (,  $i$ },

ВПІСЛЯ()) = { $\$$ } ,

ВПІСЛЯ( $E_0$ ) = { $\$$ } ,

ВПІСЛЯ( $h_0$ )= { $E_0$ ,  $E_1$ ,  $T_2$ , (,  $i$ },

ВПІСЛЯ( $E_3$ ) = {}.

Граматичні входження	Граматичні символи					
	$E$	$T$	+	(	)	$i$
$E_0$						
$E_1$			+			
$T_1$						
$T_2$						
+		$T_1$		(		$i$
$i$						
(	$E_1 E_3$	$T_2$		(		$i$
)						
$h_0$	$E_1 E_0$	$T_2$		(		$i$
$E_3$					)	

Таблиця є недермінованою.

Позначимо множину станів  $(E_0, E_1) = E_x$  і  $(E_1, E_3) = E_y$

# Побудова детермінованої таблиці переходів для граматики $\Gamma_{9.2}$

Граматичні входження	Граматичні символи					
	$E$	$T$	$+$	$($	$)$	$i$
$E_x$			$+$			
$E_y$			$+$		$)$	
$T_1$						
$T_2$						
$+$		$T_1$		$($		$i$
$($	$E_y$	$T_2$		$($		$i$
$)$						
$h_0$	$E_x$	$T_2$		$($		$i$
$i$						

# Побудова керуючої таблиці для граматики $\Gamma_{9.2}$

- 1.  $E_0 \rightarrow E_1 + T_1$
- 2.  $E \rightarrow T_2$
- 3.  $T \rightarrow (E_3)$
- 4.  $T \rightarrow i$

$(E_0, E_1) = E_x$   
 $(E_1, E_3) = E_y$

- 1.  $E_x \rightarrow E_x + T_1$
- 2.  $E \rightarrow T_2$
- 3.  $T \rightarrow (E_y)$
- 4.  $T \rightarrow i$

Граматичні входження	Термінальні символи				
	$\perp_K$	+	(	)	$i$
$E_x$	Д	П	П	П	П
$E_y$	В	П	П	П	П
$T_1$	3 (1)	3 (1)	3 (1)	3 (1)	3 (1)
$T_2$	3 (2)	3 (2)	3 (2)	3 (2)	3 (2)
+	В	П	П	П	П
$i$	3 (4)	3 (4)	3 (4)	3 (4)	3 (4)
(	В	П	П	П	П
)	3 (3)	3 (3)	3 (3)	3 (3)	3 (3)
$h_0$	В	П	П	П	П

$$(E_0, E_1) = E_x$$

$$(E_1, E_3) = E_y$$

1.  $E_x \rightarrow E_x + T_1$
2.  $E \rightarrow T_2$
3.  $T \rightarrow (E_y)$
4.  $T \rightarrow i$

Граматичні входження	Таблиця переходів						Керуюча таблиця				
	Граматичні символи						Термінальні символи				
	$E$	$T$	$+$	$($	$)$	$i$	$\perp_K$	$+$	$($	$)$	$i$
$E_x$			$+$				Д	П	П	П	П
$E_y$			$+$		$)$		В	П	П	П	П
$T_1$							3 (1)	3 (1)	3 (1)	3 (1)	3 (1)
$T_2$							3 (2)	3 (2)	3 (2)	3 (2)	3 (2)
$+$		$T_1$		$($		$i$	В	П	П	П	П
$i$							3 (4)	3 (4)	3 (4)	3 (4)	3 (4)
$($	$E_y$	$T_2$		$($		$i$	В	П	П	П	П
$)$							3 (3)	3 (3)	3 (3)	3 (3)	3 (3)
$h_0$	$E_x$	$T_2$		$($		$i$	В	П	П	П	П

Приклад роботи розпізнавача

Вхід	Магазин	Дія	Вхід	Магазин	Дія
$((i+i)+i)+i$	$h_0$	П	$+i)+i$	$h_0(E_y)$	П
$(i+i)+i)+i$	$h_0($	П	$i)+i$	$h_0(E_y+)$	П
$i+i)+i)+i$	$h_0(($	П	$) + i$	$h_0(E_y+i)$	3(4)
$+i)+i)+i$	$h_0((i$	3(4)	$) + i$	$h_0(E_y+T_1)$	3(1)
$+i)+i)+i$	$h_0((T_2$	3(2)	$) + i$	$h_0(E_y)$	П
$+i)+i)+i$	$h_0((E_y$	П	$+i$	$h_0(E_y)$	3(3)
$i)+i)+i$	$h_0((E_y +$	П	$+i$	$h_0 T_2$	3(2)
$) + i)+i$	$h_0((E_y + i$	3(4)	$+i$	$h_0 E_x$	П
$) + i)+i$	$h_0((E_y + T_1$	3(1)	$i \perp_K$	$h_0 E_x +$	П
$) + i)+i$	$h_0((E_y$	П	$\perp_K$	$h_0 E_x + i$	3(4)
$+i)+i$	$h_0(E_y)$	3(3)	$\perp_K$	$h_0 E_x + T_1$	3(1)
$+i)+i$	$h_0(T_2$	3(2)	$\perp_K$	$h_0 E_x$	Д

Дякую за увагу