



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

# Формальні мови, граматики і автомати

## Лекція 2. Використання скінчених автоматів. Парсери



Проф. Гавриленко Світлана Юріївна  
+380664088551 (Viber)  
+380632864663 (Telegram)  
[Svitlana.Gavrylenko@khpі.edu.ua](mailto:Svitlana.Gavrylenko@khpі.edu.ua)  
Вечірній корпус, 306ВК

# Застосування скінченних автоматів

Скінченні автомати (Finite State Machines) застосовуються в різних областях, включаючи:

- Розробка компіляторів та інтерпретаторів для мов програмування.
- Синтаксичний аналіз текстових даних, наприклад, для пошуку ключових слів або визначення структури документів.
- Розробка мережевих протоколів та обробка мережевих пакетів.
- Розробка алгоритмів Штучного Інтелекту, таких як алгоритми машинного навчання та розпізнавання образів.
- Розробка систем автоматичного керування, наприклад, систем керування виробничими процесами.
- Реалізація ігрових двигунів для комп'ютерних ігор.
- Розробка пристроїв автоматичного керування, наприклад, для керування роботами.

# Парсинг

Термін «парсинг» походить від англійського дієслова to parse, що означає у перекладі з англійської «частинами». Процес є синтаксичним аналізом будь-якого набору пов'язаних один з одним даних. У загальному вигляді парсинг виконується у кілька етапів:

- Сканування вихідного масиву інформації (HTML-коду, тексту, бази даних тощо).
- Відокремлення семантично значущих одиниць за заданими параметрами — наприклад, заголовків, посилань, абзаців, виділених жирним шрифтом фрагментів, пунктів меню.
- Конвертація даних у формат, зручний для вивчення, а також їх систематизація у вигляді таблиць або звітів для подальшого використання.

Об'єктом парсингу може бути будь-яка граматично структурована система: інформація, подана природною або штучною мовами.

Чітко визначити межі лексеми, які в початковому тексті явно не задані та виділили лексеми. Прикладом лексем у мові програмування є: ідентифікатори, строкові, символьні і числові константи, ключові (службові) слова вхідного мови, знаки операцій і роздільники.

# Приклади використання кінцевих автоматів у парсерах

**Компілятори:** Кінцеві автомати використовуються для лексичного аналізу і перевірки синтаксису мов програмування.

**Парсери XML:** Для валідації XML-документів і вилучення даних.

**Протоколи мережевої комунікації:** Для аналізу пакетів даних і виявлення помилок.

**Текстові редактори:** Для підсвічування синтаксису і автодоповнення коду.

# ТЕРМІНОГОЛІЯ

- **Token (умовна окрема одиниця)** – це найменший (атомарний) елемент із визначеним значенням шаблону.
- **Шаблон (Pattern)** - правило, що описує набір рядків.
- **Лексема (lexeme)** - послідовність символів, що відповідає якомусь шаблону.

## Examples

Token	Pattern	Sample Lexeme
while	while	while
relation_op	=   !=   <   >	<
integer	(0-9)*	42
string	Characters between " "	"hello"

# Лексичний аналізатор

Для виділення лексем використовуються лексичні аналізатори. Прикладом лексем у мові програмування є: ідентифікатори, строкові, символьні і числові константи, ключові (службові) слова вхідного мови, знаки операцій і роздільники.

Лексичний аналізатор **складається з окремих автоматів**, кожен з яких розпізнає **одну задану лексему**.

Всі автомати мають однакову структуру і відрізняються тільки внутрішніми станами, що пов'язано з відмінностями розпізнаються лексем.

Для більшості текстів **межі лексем** розпізнаються за заданими символами: **пробіли, знаки операцій, символи коментарів, а також роздільники (кома, крапка і т.д.)**.

Разом із тим такі символи можуть самі бути **лексемами** і необхідно не пропустити їх при розпізнаванні тексту.

# Принцип роботи лексичних аналізаторів

- з вхідного потоку вибирається черговий символ, в залежності від якого запускається той чи інший сканер (символ може бути також проігноровано або визнано помилковим);
- запущений сканер переглядає вхідний потік символів програми на початковій мові, виділяючи символи, що входять до необхідної лексеми, до виявлення чергового символу, який може обмежувати лексему, або до виявлення помилкового символу;
- при успішному розпізнаванні інформація про виділену лексему заноситься в спеціальну таблицю лексем, алгоритм повертається до першого етапу і продовжує розглядати вхідний потік символів з того місця, на якому зупинився сканер;
- при невдалому розпізнаванні видається повідомлення про помилку, а подальші дії залежать від реалізації аналізатора – або його виконання припиняється, або робиться спроба розпізнати наступну лексему (йде повернення до першого етапу алгоритму).

# Приклад 1. Частина 1

## **Постановка проблеми.**

Є список цілих і дійсних чисел, розділених пробілом, наприклад: 0.1045 12.045. 15

Виділити лексеми за рахунок побудови основної таблиці абстрактного автомата і граф-схеми переходів.



# Приклад 1. Визначення станів автомату

- Визначимо входні стани:  $X = \{x_1, x_2, x_3, x_4, x_5\}$ , де  $x_1$  – поява пробілу,  $x_2$  – поява цифри “0”,  $x_3$  – поява цифри ”1,2...9”,  $x_4$  – поява крапки «.»,  $x_5$  – поява забороненого символу (всі інші символи).
- Визначимо вихідні стани  $Y = \{y_0, y_1, y_2, y_3, y_4\}$ , де  $y_0$  – лексема не виділена,  $y_1$  – виділено число 0,  $y_2$  – помилка зчитування;  $y_3$  – виділено дійсне число,  $y_4$  – виділено ціле число. (Можливо вихідний стан  $y_5$  – виділено крапку).
- Визначимо внутрішні стани  $S = \{s_0, s_1, s_2, s_3, s_4\}$ :  $s_0$  – початковий стан,  $s_1$  – сформовано число нуль,  $s_2$  – завершено формування цілої частини дійсного числа,  $s_3$  – формування дробової частини дійсного числа,  $s_4$  – формування цілого числа або цілої частини дійсного числа

# Приклад 1. Процес сканування вхідного рядка. Частина 1

$s_0$  – початковий стан,  $s_1$  – сформовано число нуль,  $s_2$  – завершено формування цілої частини дійсного числа,  $s_3$  – формування дробової частини дійсного числа,  $s_4$  – формування цілого числа або цілої частини дійсного числа.

$y_0$  – лексема не виділена,  $y_1$  – виділено число 0,  $y_2$  – помилка зчитування;  $y_3$  – виділено дійсне число,  $y_4$  – виділено ціле число.

Вхідний символ	Start	1	2	.	0	4	5	пробіл
Множина вхідних символів X	–	x3 Цифра 1..9	x3 Цифра 1..9	x4	x2 Цифра 0	x3 Цифра 1..9	x3 Цифра 1..9	x1
Множина внутрішніх станів автомату S	s0	s4						
Множина вихідних станів Y	–	y0						

# Приклад 1. Процес сканування вхідного рядка. Частина 1

$s_0$  – початковий стан,  $s_1$  – сформовано число нуль,  $s_2$  – завершено формування цілої частини дійсного числа,  $s_3$  – формування дробової частини дійсного числа,  $s_4$  – формування цілого числа або цілої частини дійсного числа.

$y_0$  – лексема не виділена,  $y_1$  – виділено число 0,  $y_2$  – помилка зчитування;  $y_3$  – виділено дійсне число,  $y_4$  – виділено ціле число.

Вхідний символ	Start	1	2	.	0	4	5	пробіл
Множина вхідних символів X	–	x3 Цифра 1..9	x3 Цифра 1..9	x4	x2 Цифра 0	x3 Цифра 1..9	x3 Цифра 1..9	x1
Множина внутрішніх станів автомату S	s0	s4	s4					
Множина вихідних станів Y	–	y0	y0					

# Приклад 1. Процес сканування вхідного рядка. Частина 1

$s_0$  – початковий стан,  $s_1$  – сформовано число нуль,  $s_2$  – завершено формування цілої частини дійсного числа,  $s_3$  – формування дробової частини дійсного числа,  $s_4$  – формування цілого числа або цілої частини дійсного числа.

$y_0$  – лексема не виділена,  $y_1$  – виділено число 0,  $y_2$  – помилка зчитування;  $y_3$  – виділено дійсне число,  $y_4$  – виділено ціле число.

Вхідний символ	Start	1	2	.	0	4	5	пробіл
Множина вхідних символів X	–	x3 Цифра 1..9	x3 Цифра 1..9	x4	x2 Цифра 0	x3 Цифра 1..9	x3 Цифра 1..9	x1
Множина внутрішніх станів автомату S	s0	s4	s4	s2				
Множина вихідних станів Y	–	y0	y0	y0				

# Приклад 1. Процес сканування вхідного рядка. Частина 1

$s_0$  – початковий стан,  $s_1$  – сформовано число нуль,  $s_2$  – завершено формування цілої частини дійсного числа,  $s_3$  – формування дробової частини дійсного числа,  $s_4$  – формування цілого числа або цілої частини дійсного числа.

$y_0$  – лексема не виділена,  $y_1$  – виділено число 0,  $y_2$  – помилка зчитування;  $y_3$  – виділено дійсне число,  $y_4$  – виділено ціле число.

Вхідний символ	Start	1	2	.	0	4	5	пробіл
Множина вхідних символів X	–	x3 Цифра 1..9	x3 Цифра 1..9	x4	x2 Цифра 0	x3 Цифра 1..9	x3 Цифра 1..9	x1
Множина внутрішніх станів автомату S	s0	s4	s4	s2	s3			
Множина вихідних станів Y	–	y0	y0	y0	y0			

# Приклад 1. Процес сканування вхідного рядка. Частина 1

$s_0$  – початковий стан,  $s_1$  – сформовано число нуль,  $s_2$  – завершено формування цілої частини дійсного числа,  $s_3$  – формування дробової частини дійсного числа,  $s_4$  – формування цілого числа або цілої частини дійсного числа.

$y_0$  – лексема не виділена,  $y_1$  – виділено число 0,  $y_2$  – помилка зчитування;  $y_3$  – виділено дійсне число,  $y_4$  – виділено ціле число.

Вхідний символ	Start	1	2	.	0	4	5	пробіл
Множина вхідних символів X	–	x3 Цифра 1..9	x3 Цифра 1..9	x4	x2 Цифра 0	x3 Цифра 1..9	x3 Цифра 1..9	x1
Множина внутрішніх станів автомату S	s0	s4	s4	s2	s3	s3		
Множина вихідних станів Y	–	y0	y0	y0	y0	y0		

# Приклад 1. Процес сканування вхідного рядка. Частина 1

$s_0$  – початковий стан,  $s_1$  – сформовано число нуль,  $s_2$  – завершено формування цілої частини дійсного числа,  $s_3$  – формування дробової частини дійсного числа,  $s_4$  – формування цілого числа або цілої частини дійсного числа.

$y_0$  – лексема не виділена,  $y_1$  – виділено число 0,  $y_2$  – помилка зчитування;  $y_3$  – виділено дійсне число,  $y_4$  – виділено ціле число.

Вхідний символ	Start	1	2	.	0	4	5	пробіл
Множина вхідних символів X	–	x3 Цифра 1..9	x3 Цифра 1..9	x4	x2 Цифра 0	x3 Цифра 1..9	x3 Цифра 1..9	x1
Множина внутрішніх станів автомату S	s0	s4	s4	s2	s3	s3	s3	
Множина вихідних станів Y	–	y0	y0	y0	y0	y0	y0	

# Приклад 1. Процес сканування вхідного рядка. Частина 1

$s_0$  – початковий стан,  $s_1$  – сформовано число нуль,  $s_2$  – завершено формування цілої частини дійсного числа,  $s_3$  – формування дробової частини дійсного числа,  $s_4$  – формування цілого числа або цілої частини дійсного числа.

$y_0$  – лексема не виділена,  $y_1$  – виділено число 0,  $y_2$  – помилка зчитування;  $y_3$  – виділено дійсне число,  $y_4$  – виділено ціле число.

Вхідний символ	Start	1	2	.	0	4	5	пробіл
Множина вхідних символів X	–	x3 Цифра 1..9	x3 Цифра 1..9	x4	x2 Цифра 0	x3 Цифра 1..9	x3 Цифра 1..9	x1
Множина внутрішніх станів автомату S	s0	s4	s4	s2	s3	s3	s3	s0
Множина вихідних станів Y	–	y0	y0	y0	y0	y0	y0	y3



# Приклад 1. Процес сканування вхідного рядка. Частина 3

$s_0$  – початковий стан,  $s_1$  – сформовано число нуль,  $s_2$  – завершено формування цілої частини дійсного числа,  $s_3$  – формування дробової частини дійсного числа,  $s_4$  – формування цілого числа або цілої частини дійсного числа.

$y_0$  – лексема не виділена,  $y_1$  – виділено число 0,  $y_2$  – помилка зчитування;  $y_3$  – виділено дійсне число,  $y_4$  – виділено ціле число.

Вхідний символ	Start	2	2	1	0	4	5	пробіл
Множина вхідних символів X	–	x3	x3	x3	x3	x3	x3	x1
Множина внутрішніх станів автомату S	s0	s4	s4	s4	s4	s4	s4	s0
Множина вихідних станів Y	–	y0	y0	y0	y0	y0	y0	y4

# Приклад 1. Процес сканування вхідного рядка. Частина 2

$s_0$  – початковий стан,  $s_1$  – сформовано число нуль,  $s_2$  – завершено формування цілої частини дійсного числа,  $s_3$  – формування дробової частини дійсного числа,  $s_4$  – формування цілого числа або цілої частини дійсного числа.

$y_0$  – лексема не виділена,  $y_1$  – виділено число 0,  $y_2$  – помилка зчитування;  $y_3$  – виділено дійсне число,  $y_4$  – виділено ціле число. (Після надходження крапки сформовано помилковий стан. Надалі автомат працює некоректно)

Вхідний символ	Start	0	.	1	0	.	5	пробіл
Множина вхідних символів X	–	x2	x4	x3	x2	x3	x3	x1
Множина внутрішніх станів автомату S	s0	s1	s2	s3	s3	s0 Стан помилковий	s4	s0
Множина вихідних станів Y	–	y0	y0	y0	y0	y2	y0	y4

# Приклад 1. Основна таблиця абстрактного автомата

$s_0$  – початковий стан,  $s_1$  – сформовано число нуль,  $s_2$  – завершено формування цілої частини дійсного числа,  $s_3$  – формування дробової частини дійсного числа,  $s_4$  – формування цілого числа або цілої частини дійсного числа.

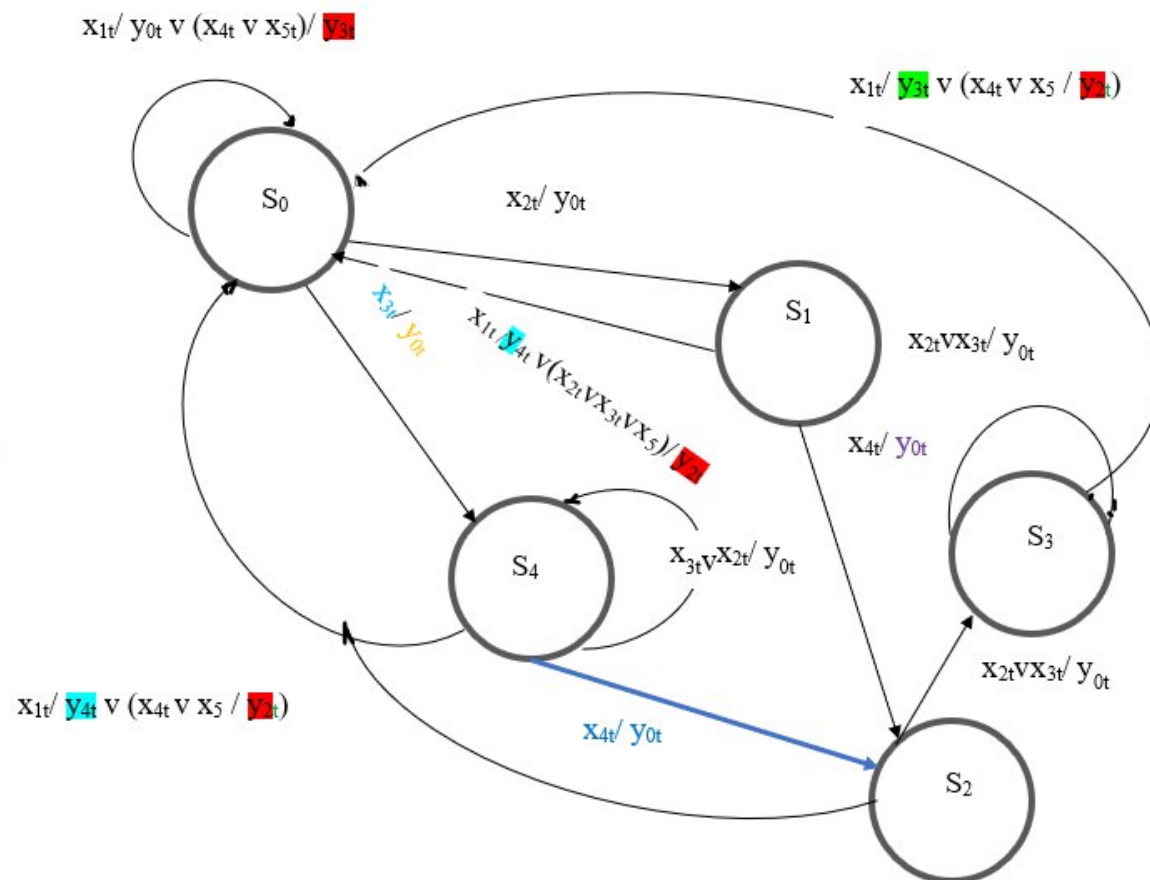
$y_0$  – лексема не виділена,  $y_1$  – виділено число 0,  $y_2$  – помилка зчитування;  $y_3$  – виділено дійсне число,  $y_4$  – виділено ціле число. (можливо інший стан)

0

Finite set of states	Input Symbols				
	$X_{1t}$	$X_{2t}$	$X_{3t}$	$X_{4t}$	$X_{5t}$
	space	0	1..9	.	forbidden
$s_{0\ t-1}$ (start state)	$s_{0t}/y_{0t}$	$s_{1t}/y_{0t}$	$s_{4t}/y_{0t}$	$s_{0t}/y_{2t}$	$s_{0t}/y_{2t}$
$s_{1\ t-1}$ (number =0 )	$s_{0t}/y_{1t}$	$s_{0t}/y_{2t}$	$s_{0t}/y_{2t}$	$s_{2t}/y_{0t}$	$s_{0t}/y_{2t}$
$s_{2\ t-1}$ (0. or NN...N.)	$s_{0t}/y_{2t}$	$s_{3t}/y_{0t}$	$s_{3t}/y_{0t}$	$s_{0t}/y_{2t}$	$s_{0t}/y_{2t}$
$s_{3\ t-1}$ (float number 0.NNN)	$s_{0t}/y_{3t}$	$s_{3t}/y_{0t}$	$s_{3t}/y_{0t}$	$s_{0t}/y_{2t}$	$s_{0t}/y_{2t}$
$s_{4\ t-1}$ (int number or float NN....N)	$s_{0t}/y_{4t}$	$s_{4t}/y_{0t}$	$s_{4t}/y_{0t}$	$s_{2t}/y_{0t}$	$s_{0t}/y_{2t}$

# Приклад 1. Діаграма переходів автомата

Finite set of states	Input Symbols				
	$X_{1t}$	$X_{2t}$	$X_{3t}$	$X_{4t}$	$X_{5t}$
	space	0	1..9	.	stop
$S_{0\ t-1}$ (start state)	$S_{0t}/y_{0t}$	$S_{1t}/y_{0t}$	$S_{4t}/y_{0t}$	$S_{0t}/\text{red}$	$S_{0t}/\text{red}$
$S_{1\ t-1}$ (number=0)	$S_{0t}/\text{yellow}$	$S_{0t}/\text{red}$	$S_{0t}/\text{red}$	$S_{2t}/y_{0t}$	$S_{0t}/\text{red}$
$S_{2\ t-1}$ 1. N.)	$S_{0t}/\text{red}$	$S_{3t}/y_{0t}$	$S_{3t}/y_{0t}$	$S_{0t}/\text{red}$	$S_{0t}/\text{red}$
$S_{3\ t-1}$ (float number)	$S_{0t}/\text{green}$	$S_{3t}/y_{0t}$	$S_{3t}/y_{0t}$	$S_{0t}/\text{red}$	$S_{0t}/\text{red}$
$S_{4\ t-1}$ (int number)	$S_{0t}/\text{cyan}$	$S_{4t}/y_{0t}$	$S_{4t}/y_{3t}$	$S_{2t}/y_{0t}$	$S_{0t}/\text{red}$



# Приклад 2

## Постановка проблеми.

Опис масиву (списку) чисел містить: змінну (у вигляді послідовності літер англійського алфавіту та цифр за умови, що першим символом може бути літера), знак присвоєння «=», дужки «[, ]», десяткові цифри та кому «,». Допускається також, що вираз може містити пробіл.

Наприклад: `ident = [ 1, 2, 34]`

`= ident [, 12, 34]` с точки зору сканера це не помилка (усі лексеми є допустимими)

## Приклад 2. Визначення станів автомату

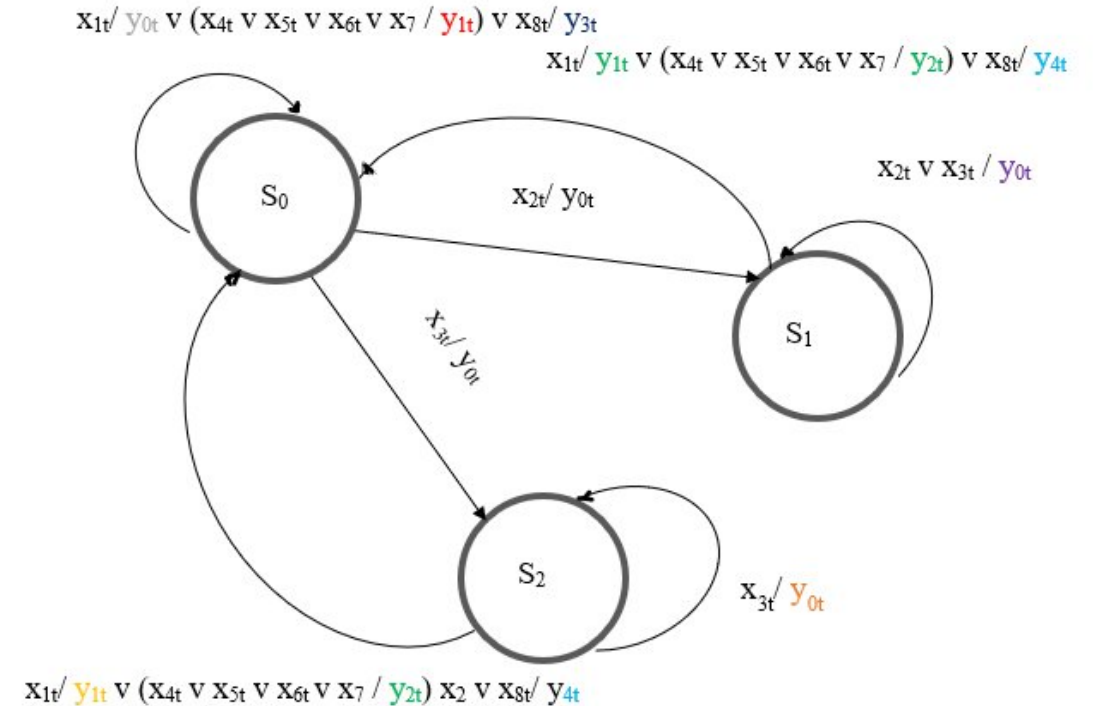
- Визначимо входні стани:  $X = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\}$ , де  $x_1$  — поява пробілу,  $x_2$  — поява будь-якої англійської літери,  $x_3$  — поява цифри,  $x_4$  — знаку присвоєння «=»,  $x_5$  — поява дужки «[»,  $x_6$  — поява коми «,»,  $x_7$  — поява дужки «]»,  $x_8$  — поява забороненого символу (всі інші символи).
- Визначимо вихідні стани  $Y = \{y_0, y_1, y_2, y_3, y_4\}$ , де  $y_0$  — лексема не виділена,  $y_1$  — виділена одна лексема,  $y_2$  — виділено дві лексеми (при появі на вході знаків : «=, ], [, , » які одночасно є межею між лексемами і лексемами),  $y_3$  — помилка зчитування;  $y_4$  — виділена одна лексема і помилка.
- Визначимо внутрішні стани  $S = \{s_0, s_1, s_2\}$ :  $s_0$  — початковий стан,  $s_1$  — формування ідентифікатора,  $s_2$  — формування числа.

# Приклад 2. Таблиця роботи автомату

Внутрішні стани	Вхідні стани							
	Вихідні стани: $y_0$ – лексема не виділена, $y_1$ – виділена одна лексема, $y_2$ – виділено дві лексеми, $y_3$ – помилка зчитування, $y_4$ – виділена одна лексема і помилка							
	$X_{1t}$ пробіл	$X_{2t}$ будь-яка англійськ а літера	$X_{3t}$  цифр	$X_{4t}$ знак присвою- вання	$X_{5t}$ дужка «[»	$X_{6t}$ Кома «,»	$X_{7t}$ Дужка «]»	$X_{8t}$ Заборо- нений символ
$S_{0\ t-1}$ очікування	$S_{0t}/y_{0t}$	$S_{1t}/y_{0t}$	$S_{2t}/y_{0t}$	$S_{0t}/y_{1t}$	$S_{0t}/y_{1t}$	$S_{0t}/y_{1t}$	$S_{0t}/y_{1t}$	$S_{0t}/y_{3t}$
$S_{1\ t-1}$ формування ідентифікатора	$S_{0t}/y_{1t}$	$S_{1t}/y_{0t}$	$S_{1t}/y_{0t}$	$S_{0t}/y_{2t}$	$S_{0t}/y_{2t}$	$S_{0t}/y_{2t}$	$S_{0t}/y_{2t}$	$S_{0t}/y_{4t}$
$S_{2\ t-1}$ формування числа	$S_{0t}/y_{1t}$	$S_{0t}/y_{4t}$	$S_{2t}/y_{0t}$	$S_{0t}/y_{2t}$	$S_{0t}/y_{2t}$	$S_{0t}/y_{2t}$	$S_{0t}/y_{2t}$	$S_{0t}/y_{4t}$

# Приклад 2. Діаграма станів

Внутрішні стани	Вхідні стани							
	Вихідні стани: $y_0$ – лексема не виділена, $y_1$ – виділена одна лексема, $y_2$ – виділено дві лексеми, $y_4$ – виділена одна лексема і помилка.							
	$x_{1t}$ пробіл	$x_{2t}$ будь-яка англійська літера	$x_{3t}$ цифра	$x_{4t}$ знак присвоєння	$x_{5t}$ дужка «[»	$x_{6t}$ Кома «.,»	$x_{7t}$ Дужка «]»	$x_{8t}$ Заборонений символ
$S_{0\ t-1}$ очікування	$S_{0t}/y_{0t}$	$S_{1t}/y_{0t}$	$S_{2t}/y_{0t}$	$S_{0t}/y_{1t}$	$S_{0t}/y_{1t}$	$S_{0t}/y_{1t}$	$S_{0t}/y_{1t}$	$S_{0t}/y_{3t}$
$S_{1\ t-1}$ формування ідентифікатора	$S_{0t}/y_{1t}$	$S_{1t}/y_{0t}$	$S_{1t}/y_{0t}$	$S_{0t}/y_{2t}$	$S_{0t}/y_{2t}$	$S_{0t}/y_{2t}$	$S_{0t}/y_{2t}$	$S_{0t}/y_{4t}$
$S_{2\ t-1}$ формування числа	$S_{0t}/y_{1t}$	$S_{0t}/y_{4t}$	$S_{2t}/y_{0t}$	$S_{0t}/y_{2t}$	$S_{0t}/y_{2t}$	$S_{0t}/y_{2t}$	$S_{0t}/y_{2t}$	$S_{0t}/y_{4t}$





# Приклад 3

**Постановка проблеми.** Виділити лексеми, що представляють собою *цілочисельні* константи в форматі мови C. Відповідно до вимог мови, такі константи можуть бути *десятковими, восьмирічними або шістнадцятирічними*. **Восьмирічною** константою вважається число, що **починається з 0** і містить цифри від 0 до 7; **шістнадцятирічна** константа повинна починатися з послідовності символів **0x** і може містити цифри і букви від *a* до *f*. Решта чисел вважаються десятковими. **Константа** може починатися також з одного із знаків **+** **або** **-**. Для уникнення плутанини і скорочення обсягу інформації в прикладі будемо вважати, що всі допустимі літери є малими.

## Приклад 3. Визначення станів автомату

Визначимо вхідні стани:  $X = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\}$  де  $x_1$  – “+,-“,  $x_2$  – “0”,  $x_3$  – “x”,  $x_4$  – “1...7”,  $x_5$  – “8,9”,  $x_6$  – “a...f”,  $x_7$  – “пробіл або кінець рядка –  $\perp$ ”,  $x_8$  – “інші символи”

Визначимо вихідні стани  $Y = \{y_0, y_1, y_2\}$ , де  $y_0$  – цифра не виділена,  $y_1$  – цифра виділена,  $y_2$  – помилка.

Визначимо внутрішні стани  $S = \{s_0, s_1, s_2, s_3, s_4, s_5\}$ , де  $s_0$  – початковий стан;  $s_1$  – поява знаку,  $s_2$  – формування восьмирічної або шістнадцятирічної цифри числа «0»,  $s_3$  – формування десяткового числа,  $s_4$  – формування шістнадцятирічного числа,  $s_5$  – формування восьмирічного цифра.

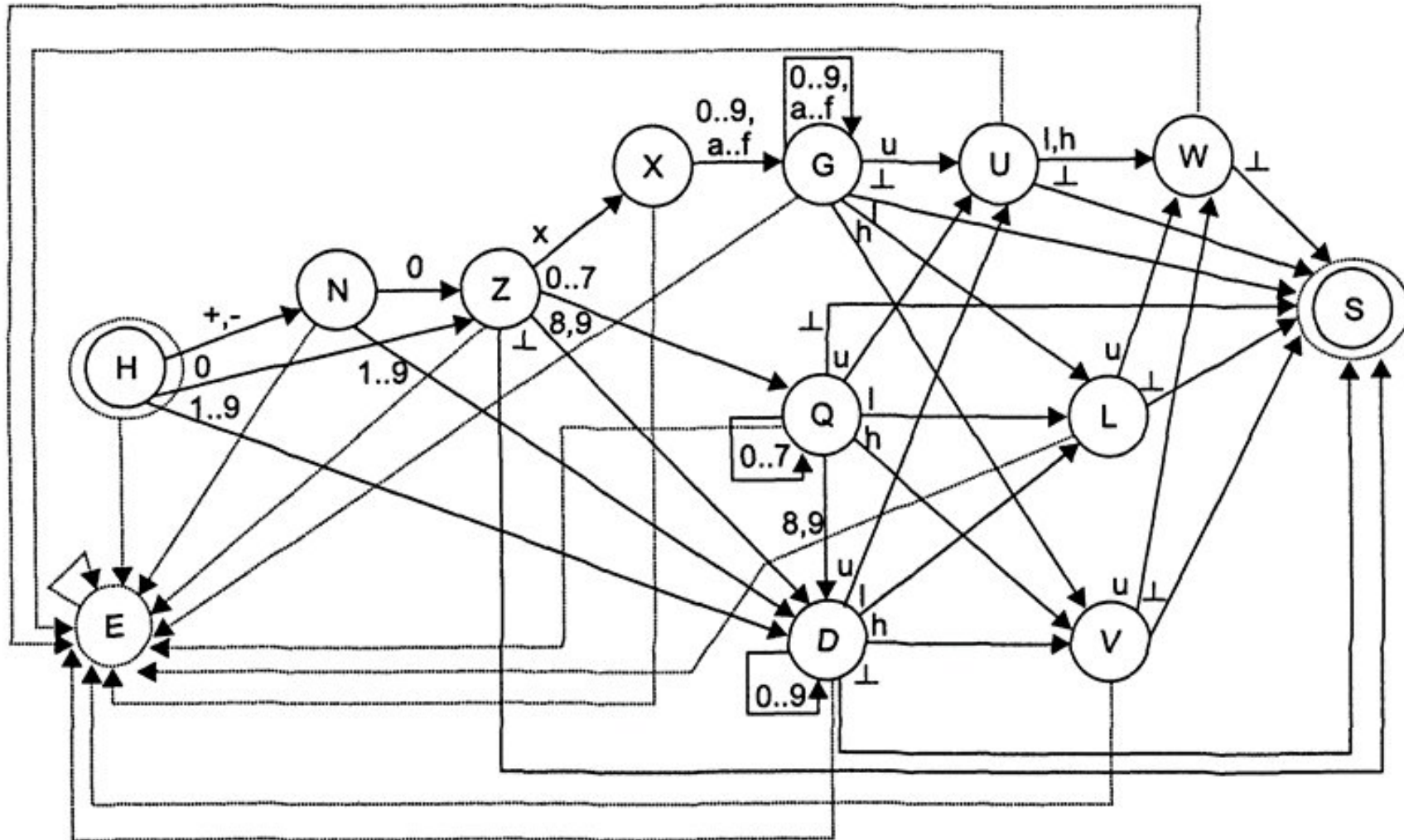
# Приклад 3. Таблиця роботи автомату

Стани	Входи							
	X <sub>1</sub>	X <sub>2</sub>	X <sub>3</sub>	X <sub>4</sub>	X <sub>5</sub>	X <sub>6</sub>	X <sub>7</sub>	X <sub>8</sub>
	+, -	0	x	1...7	8,9	a...f	Пробіл, ⊥	Інші СИМВОЛИ
S <sub>0 t-1</sub> (п. стан)	S <sub>1 t</sub> /Y <sub>0 t</sub>	S <sub>2 t</sub> /Y <sub>0 t</sub>	S <sub>0 t</sub> /Y <sub>2 t</sub>	S <sub>3 t</sub> /Y <sub>0 t</sub>	S <sub>3 t</sub> /Y <sub>0 t</sub>	S <sub>0 t</sub> /Y <sub>2 t</sub>	S <sub>0 t</sub> /Y <sub>0 t</sub>	S <sub>0 t</sub> /Y <sub>2 t</sub>
S <sub>1 t-1</sub> (знак)	S <sub>0 t</sub> /Y <sub>2 t</sub>	S <sub>2 t</sub> /Y <sub>0 t</sub>	S <sub>0 t</sub> /Y <sub>2 t</sub>	S <sub>3 t</sub> /Y <sub>0 t</sub>	S <sub>3 t</sub> /Y <sub>0 t</sub>	S <sub>0 t</sub> /Y <sub>2 t</sub>	S <sub>0 t</sub> /Y <sub>2 t</sub>	S <sub>0 t</sub> /Y <sub>2 t</sub>
S <sub>2 t-1</sub> (A <sub>h,8</sub> , 0)	S <sub>0 t</sub> /Y <sub>2 t</sub>	S <sub>0 t</sub> /Y <sub>2 t</sub>	S <sub>4 t</sub> /Y <sub>0 t</sub>	S <sub>5 t</sub> /Y <sub>0 t</sub>	S <sub>0 t</sub> /Y <sub>2 t</sub>	S <sub>0 t</sub> /Y <sub>2 t</sub>	S <sub>0 t</sub> /Y <sub>1 t</sub>	S <sub>0 t</sub> /Y <sub>2 t</sub>
S <sub>3 t-1</sub> (A <sub>10</sub> )	S <sub>0 t</sub> /Y <sub>2 t</sub>	S <sub>3 t</sub> /Y <sub>0 t</sub>	S <sub>0 t</sub> /Y <sub>2 t</sub>	S <sub>3 t</sub> /Y <sub>0 t</sub>	S <sub>3 t</sub> /Y <sub>0 t</sub>	S <sub>0 t</sub> /Y <sub>2 t</sub>	S <sub>0 t</sub> /Y <sub>1 t</sub>	S <sub>0 t</sub> /Y <sub>2 t</sub>
S <sub>4 t-1</sub> (A <sub>h</sub> )	S <sub>0 t</sub> /Y <sub>2 t</sub>	S <sub>4 t</sub> /Y <sub>0 t</sub>	S <sub>0 t</sub> /Y <sub>2 t</sub>	S <sub>4 t</sub> /Y <sub>0 t</sub>	S <sub>4 t</sub> /Y <sub>0 t</sub>	S <sub>4 t</sub> /Y <sub>0 t</sub>	S <sub>0 t</sub> /Y <sub>1 t</sub>	S <sub>0 t</sub> /Y <sub>2 t</sub>
S <sub>5 t-1</sub> (A <sub>8</sub> )	S <sub>0 t</sub> /Y <sub>2 t</sub>	S <sub>5 t</sub> /Y <sub>0 t</sub>	S <sub>0 t</sub> /Y <sub>2 t</sub>	S <sub>5 t</sub> /Y <sub>0 t</sub>	S <sub>0 t</sub> /Y <sub>2 t</sub>	S <sub>0 t</sub> /Y <sub>2 t</sub>	S <sub>0 t</sub> /Y <sub>1 t</sub>	S <sub>0 t</sub> /Y <sub>2 t</sub>

## Приклад 3. Процес сканування вхідного рядка

Вхідний символ	Start	+	0	x	a	4	5	пробіл
Множина вхідних символів X	–	x1	x2	x3	x6	x4	x4	x7
Множина внутрішніх станів автомату S	s0	s1	s2	s4	s4	s4	s4	s0
Множина вихідних станів Y	–	y0	y0	y0	y0	y0	y0	y1

# Приклад 3. Граф-схема роботи аналогічного автомату зі станом «помилка»



# Приклад 4

**Постановка проблеми.** Виділити лексеми в математичному виразі, який містить змінні (у вигляді послідовності літер англійського алфавіту та цифр за умови, що першим символом може бути літера), математичні знаки («+», «-», «\*», «/»), знак присвоювання «=», дужки «(, )» та знак «;». Допускається також, що математичний вираз може містити пробіл.

Прикладом математичного виразу може бути наступний рядок:

$$a1 = fg2 + (d - cde);$$

При виділенні лексем необхідно врахувати наступні ситуації. Межею лексеми можуть бути математичні знаки («+», «-», «\*», «/»), знак присвоювання «=», дужки «(, )», знак «;» та пробіл, при цьому усі перераховані символи, окрім пробілу, також являються лексемами.

## Приклад 4. Визначення станів автомату

Закодуємо входні стани:  $x_1$  – поява знаку,  $x_2$  – поява будь-якої англійської літери,  $x_3$  – поява цифри,  $x_4$  – поява забороненого символу.  $x_5$  – поява знаку присвоювання «=»,  $x_6$  – поява дужки «(»,  $x_7$  – поява дужки «)»,  $x_8$  – поява знаку «;».

Таким чином, множина входних станів  $X = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\}$ .

Визначимо вихідні стани  $Y = \{y_0, y_1, y_2, y_4\}$ , де  $y_0$  – лексема не виділена,  $y_1$  – виділена одна лексема,  $y_2$  – виділено дві лексеми (при появі на вході знаків : +, -, \*, /, ), (, ; які самі є межею між лексемами і одночасно самі є лексемами),  $y_3$  – помилка.

Визначимо внутрішні стани:  $s_0$  – початковий стан автомату,  $s_1$  – стан формування лексеми ідентифікатор.

# Приклад 4. Таблиця роботи автомату

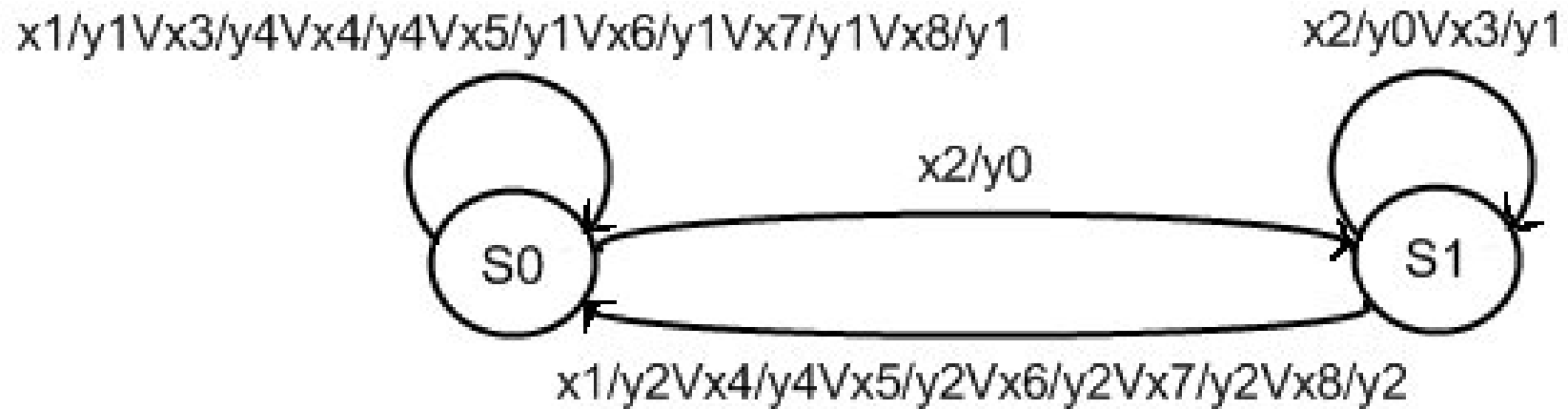
Внутрішні стани	Вхідні стани								
	$X_{1t}$ +,-	$X_{2t}$ a...z	$X_{3t}$ 0...9	$X_{4t}$ ???	$X_{5t}$ =	$X_{6t}$ (	$X_{7t}$ )	$X_{8t}$ ;	
$S_{0\ t-1}$	$S_{0t}/y_{1t}$	$S_{1t}/y_{0t}$	$S_{0t}/y_{3t}$	$S_{0t}/y_{3t}$	$S_{0t}/y_{1t}$	$S_{0t}/y_{1t}$	$S_{0t}/y_{1t}$	$S_{0t}/y_{1t}$	
$S_{1\ t-1}$	$S_{0t}/y_{2t}$	$S_{1t}/y_{0t}$	$S_{1t}/y_{0t}$	$S_{0t}/y_{3t}$	$S_{0t}/y_{2t}$	$S_{0t}/y_{2t}$	$S_{0t}/y_{2t}$	$S_{0t}/y_{2t}$	

Розглянемо роботу автомата. Якщо автомат знаходився в стані  $s_0(t-1)$  і на його вхід було подано знак (стан  $x_1$ ), то він виділить цей знак як лексему, залишиться в стані  $s_0t$  і видасть на виході стан  $y_{1t}$ . Якщо автомат знаходився в стані  $s_0(t-1)$  і на його вхід була подана буква (стан  $x_2$ ), то він перейде в стан  $s_{1t}$ . Поява на вході будь-якої іншої букви або цифри залишить його в стані  $s_{1t}$ , оскільки проходить процес визначення лексеми. Якщо автомат знаходився в стані  $s_{1t}(t-1)$  і на його вхід був поданий знак (стан  $x_1$ ), то він перейде в стан  $s_0t$  і на виході сигнал  $y_2$ , тобто буде виділено дві лексеми: ідентифікатор (змінна) та знак. Якщо автомат знаходився в стані  $s_0(t-1)$ , то поява на вході будь-якої цифри переведе його в стан  $s_{2t}$  – помилка, так як згідно умови змінна не може починатися з цифри.

Якщо на вхід подається стан  $x_4$ , то незалежно від стану автомату перейде до стану  $S_2$ , тобто визначить помилку.

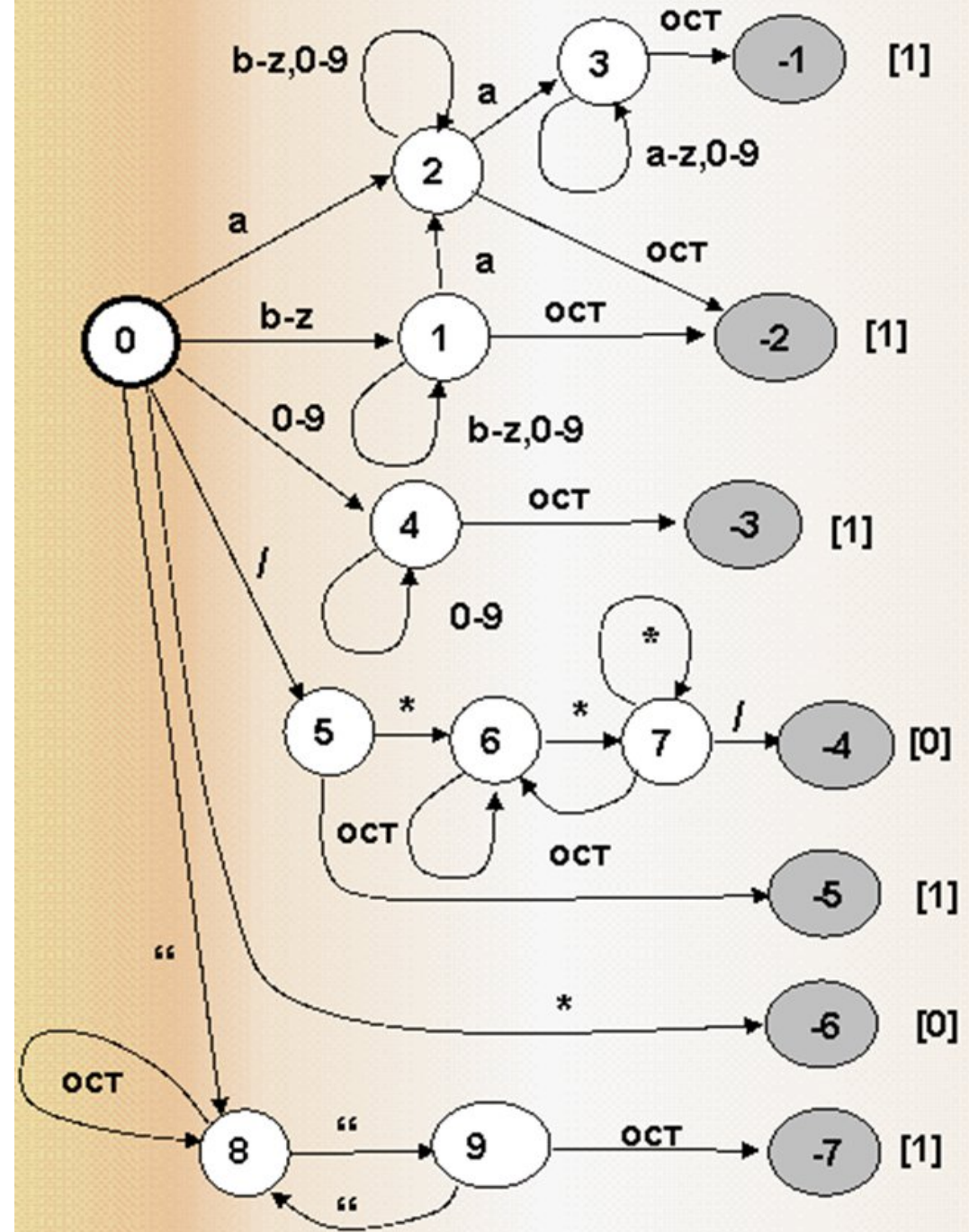


## Приклад 4. Граф-схема лексичного аналізатора



# Приклад 5

Діаграма станів-переходів для десятикових констант, ідентифікаторів (у тому числі, що містять не менше 2 символів *a*), коментарів, окремих символів / та \*, а також рядкових констант (виду "...").



# Регулярні вирази

**Регулярний вираз** — це набір правил для опису текстових рядків у вигляді послідовності звичайних символів і метасимволів (будь-який одиночний символ), який потім в якості зразка використовується в операціях пошуку і заміни тексту.

Метасимвол `[...]` використовується в конструкції `[...]` для подання будь-якого одиночного символу з числа взятих в дужки, тобто він представляє клас символів. Два символи, з'єднані знаком мінус, задають діапазон значень, наприклад `[A-Za-z]` задає всі великі та малі літери англійського алфавіту. Якщо першим символом в дужках є символ `^`, вся конструкція позначає будь-який символ, який не входить в число перерахованих в дужках. Наприклад, `[^0-9]` позначає усі нецифрові символи.

Метасимволи `^` і `$` використовується для завдання прив'язки до певного місця рядка. Метасимвол `^` як перший символ регулярного виразу позначає початок рядка. Метасимвол `$` в якості останнього символу регулярного виразу позначає кінець рядка. Наприклад:

`/^$/` — порожній рядок (початок і кінець, між якими порожньо);

`/^Perl/` — слово Perl на початку рядка;

`/Perl$/` — слово Perl в кінці рядка.

Метасимвол `|` можна розглядати як символ операції, яка задає вибір з кількох варіантів (подібно логічній операції АБО).

# Коефіцієнти, або множники метасимволів








- $r^*$  нуль і більш повторень  $r$ ;
- $r^+$  одне і більш повторень  $r$ ;
- $r?$  нуль або одне повторення  $r$ ;
- $r\{n\}$  рівно  $n$  повторень  $r$ ;
- $r\{n, +\}$   $n$  і більше повторень  $r$ ;
- $r\{n, m\}$  мінімум  $n$ , максимум  $m$  повторень  $r$ .

Наприклад:

- $/.*/$  будь-який рядок;
- $/.+/$  будь-яка непорожній рядок;
- $/[0-9]\{3\}/$  будь-яка послідовність з трьох цифр;
- $^\wedge[+]$  послідовність, що складається з будь-якого числа символів  $[$ .

# Приклади роботи жадібних та лінивих алгоритмів

Наприклад, в рядку "1234567" буде знайдено:

- для зразку  $\wedge d^*/$  або  $[0-9]^*$   максимальний фрагмент "1234567";
- для зразку  $\wedge d^+ /$  або  $[0-9]^+$   максимальний фрагмент "1234567";
- для зразку  $a \wedge d? /$   максимальний фрагмент "1";
- для зразку  $a \wedge d\{2,5\} /$   максимальний фрагмент "12345";
- для зразку  $a \wedge d^*? /$   мінімальний фрагмент "";
- для зразку  $a \wedge d^+? /$   мінімальний фрагмент "1";
- для зразку  $a \wedge d\{2,5\}? /$   мінімальний фрагмент "12".

- $\backslash d$  — клас цифрових символів, однаково, що і  $[0-9]$ .

# Приклади лексем, заданих регулярними виразами

- Ціле число:  $[+, -]? [1-9] [0-9]^*$
- Дійсне число:  $[+, -]? [0 \{1\} | [1-9] + . [0-9]^+$
- Ідентифікатор:  $[A-Za-z\_][A-Za-z\_0-9]^*$
- Ключове слово if: if
- Ключове слово while: while
- Знак операції + : \+
- Знак операції ++ : \++

Дійсно, легко виписати, наприклад, праволінійну граматику для розпізнавання ідентифікаторів:

letter -> 'a' .. 'z' | 'A' .. 'Z' | '\_'

digit -> '0' .. '9'

ident -> letter | letter tail

tail -> letter | digit | letter tail | digit tail

ДЯКУЮ ЗА УВАГУ