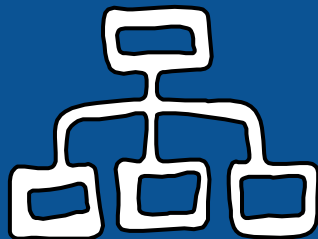


# Алгоритмы и Алгоритмические Языки

Семинар #26:

1. Распознавание слов и выражений.
2. Регулярные выражения.
3. Архитектура лексического анализатора языка выражений.

# Распознавание слов и выражений



# Какие выражения будем моделировать?

**Выражение – это:**

- **Число с плавающей точкой.**
- **Именованная переменная.**
- **Сумма двух других выражений:  $E_1 + E_2$ .**
- **Умножение двух других выражений:  $E_1 * E_2$ .**

**Примеры выражений:**

- 1, 123, 123456.002.
- x, y, length, min\_sum.
- $x + 0.5$ .
- $x * (2 + y)$ .
- $((a + b) + (c + d)) + (e + f) + g$ .

**Как более формально задать язык выражений?**

# Синтаксис языка

**Синтаксис языка** – это набор слов языка и правила построения предложений из слов этого языка.

**Набор слов языка арифметических выражений:**

- `Value(val)` - это число с плавающей точкой.
- `Variable(name)` - это строковый идентификатор.

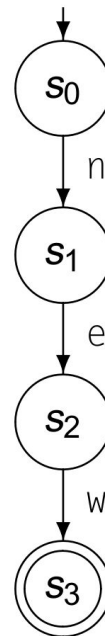
**Правила построения предложений языка арифметических выражений:**

- `OperationAdd( $E_1$ ,  $E_2$ )` – это операция сложения выражений.
- `OperationMul( $E_1$ ,  $E_2$ )` – это операция умножения выражений.

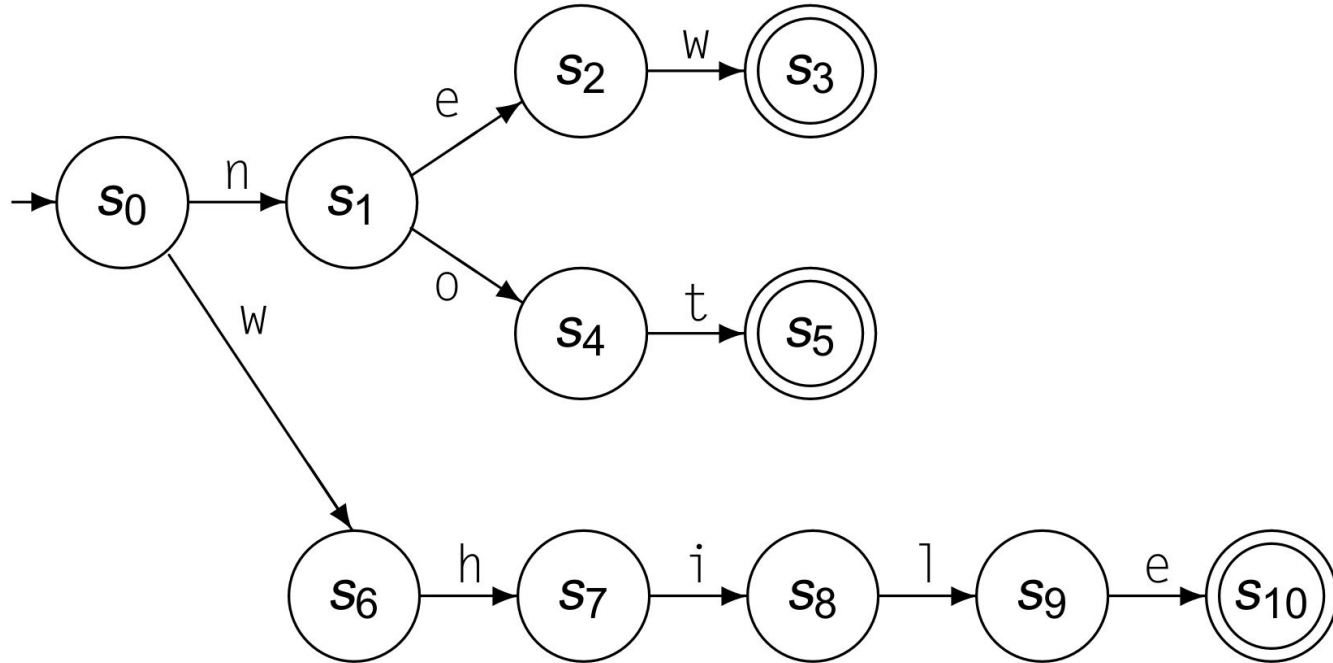
**Но какой последовательности символов соответствует «число с плавающей точкой»?**

# Распознавание фиксированного слова

```
c ← NextChar();  
if (c = 'n')  
  then begin;  
    c ← NextChar();  
    if (c = 'e')  
      then begin;  
        c ← NextChar();  
        if (c = 'w')  
          then report success;  
          else try something else;  
        end;  
      else try something else;  
    end;  
  else try something else;
```



# Распознавание нескольких слов



# Конечный автомат как распознаватель

$$S = \{s_0, s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_e\}$$

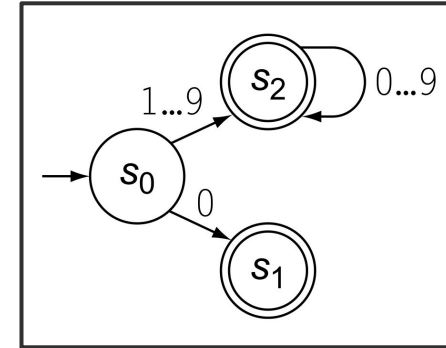
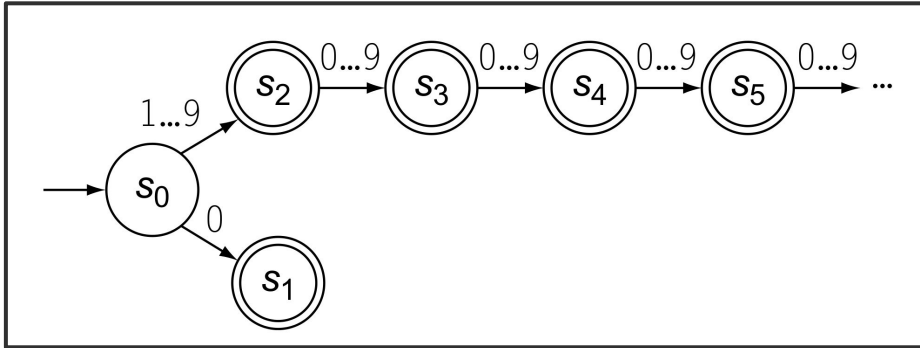
$$\Sigma = \{e, h, i, l, n, o, t, w\}$$

$$\delta = \left\{ \begin{array}{ccccc} s_0 \xrightarrow{n} s_1, & s_0 \xrightarrow{w} s_6, & s_1 \xrightarrow{e} s_2, & s_1 \xrightarrow{o} s_4, & s_2 \xrightarrow{w} s_3, \\ s_4 \xrightarrow{t} s_5, & s_6 \xrightarrow{h} s_7, & s_7 \xrightarrow{i} s_8, & s_8 \xrightarrow{l} s_9, & s_9 \xrightarrow{e} s_{10} \end{array} \right\}$$

$$s_0 = s_0$$

$$S_A = \{s_3, s_5, s_{10}\}$$

# Циклы в распознавателе



```

char ← NextChar();
state ← s0;

while (char ≠ eof and state ≠ se) do
    state ← δ(state, char);
    char ← NextChar();
end;

if (state ∈ SA)
    then report acceptance;
    else report failure;
  
```

$$S = \{s_0, s_1, s_2, s_e\}$$

$$\Sigma = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$$

$$\delta = \left\{ \begin{array}{ll} s_0 \xrightarrow{0} s_1, & s_0 \xrightarrow{1-9} s_2 \\ s_2 \xrightarrow{0-9} s_2, & s_1 \xrightarrow{0-9} s_e \end{array} \right\}$$

$$S_A = \{s_1, s_2\}$$



# Регулярные выражения



# Формализация регулярных выражений

**Регулярное выражение** – способ задания языка (множества строк над алфавитом).

Операции над регулярными выражениями:

- **Объединение языков:**

$$L(r_1 | r_2) = L(r_1) \cup L(r_2)$$

- **Конкатенация:**

$$L(r_1 r_2) = \{xy : x \in L(r_1), y \in L(r_2)\}$$

- **Замыкание Клини (звезда Клини):**

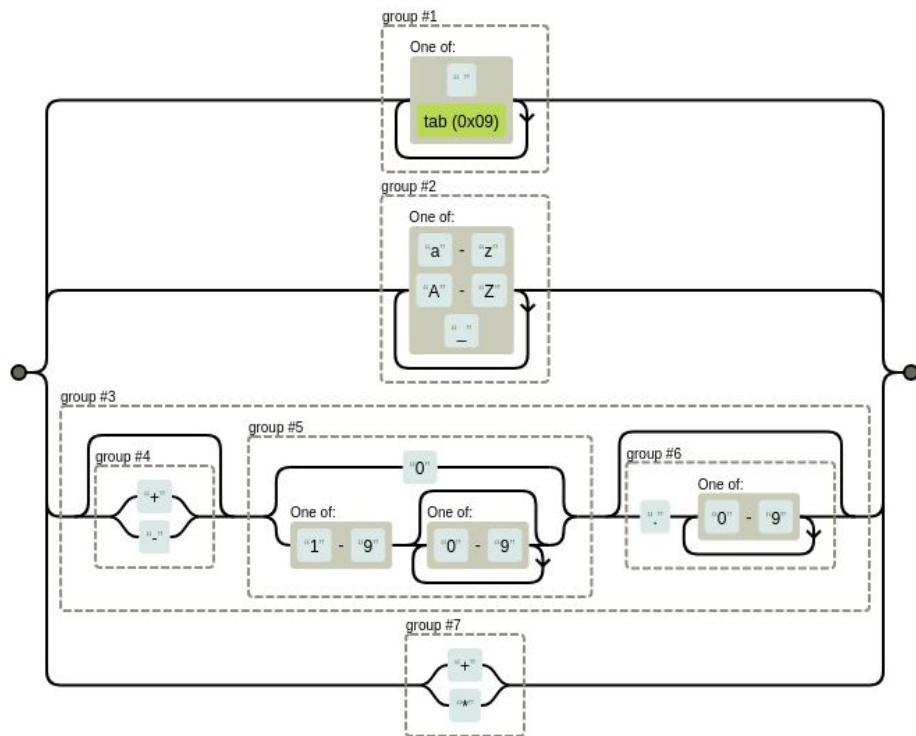
$$L(r^*) = \bigcup_{n \geq 0} L(r)^n$$

- **Плюс Клини:**

$$L(r^+) = \bigcup_{n \geq 1} L(r)^n$$

# Регулярное выражение языка выражений

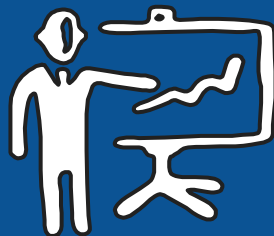
`([ \t]+)|([a-zA-Z_]+)|((\+|-)?(0|[1-9][0-9]*)(\.[0-9]+)?)|(\+|\*)`



# Архитектура лексического анализатора



# Вопросы?



Красивые иконки взяты с сайта [handdrawngoods.com](http://handdrawngoods.com)