More Applications

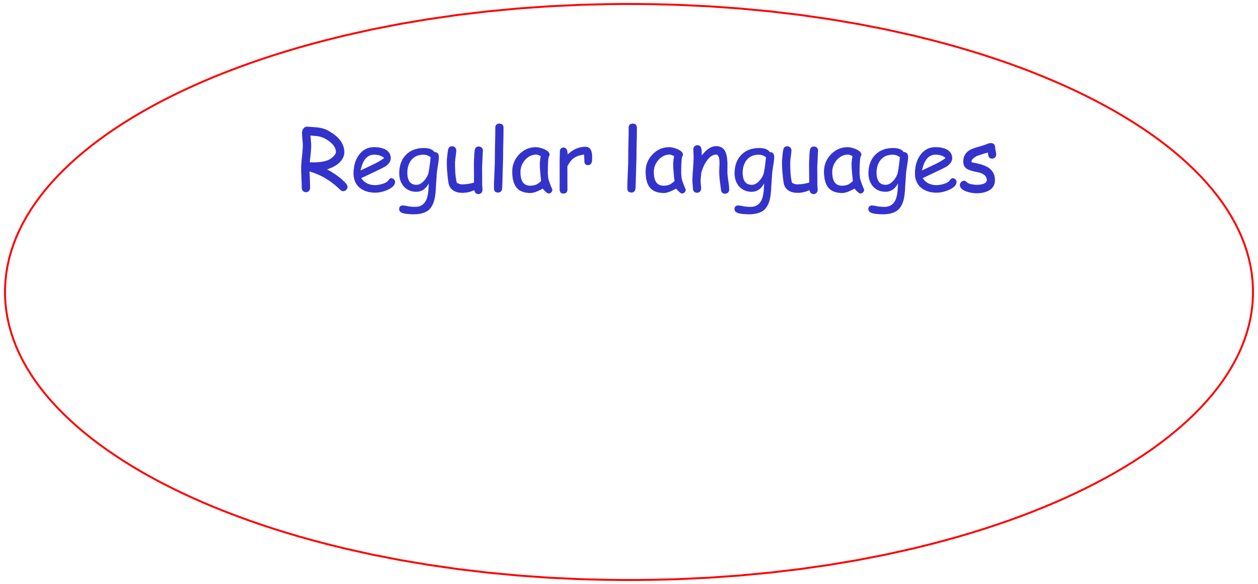
of

the Pumping Lemma

The Pumping Lemma:

- Given a ~~infinite~~ regular language L
- there exists an integer m
- for any string $w \in L$ with length $|w| \geq m$
- we can write $w = x\,y\,z$
- with $|x\,y| \leq m$ and $|y| \geq 1$
- such that: $x\,y^i\,z \in L \qquad i = 0, 1, 2, \ldots$

Non-regular languages $L = \{ww^R : w \in \Sigma^*\}$



Regular languages

Theorem: The language

$$L = \{ww^R : w \in \Sigma^*\} \qquad \Sigma = \{a, b\}$$

is not regular

Proof: Use the Pumping Lemma

$$L = \{ww^R : w \in \Sigma^*\}$$

Assume for contradiction
that L is a regular language

~~Since L is infinite~~
we can apply the Pumping Lemma

$$L = \{ww^R : w \in \Sigma^*\}$$

Let m be the integer in the Pumping Lemma

Pick a string w such that: $w \in L$ and

length $|w| \geq m$

We pick $w = a^m b^m b^m a^m$

Write $a^m b^m b^m a^m = x\ y\ z$

From the Pumping Lemma

it must be that length $|x\ y| \leq m, \quad |y| \geq 1$

$$xyz = \overbrace{\underbrace{a...a}_{x}\underbrace{a...a}_{y}...a}^{m}\underbrace{\overbrace{b...b}^{m}\overbrace{b...b}^{m}\overbrace{a...a}^{m}}_{z}$$

Thus: $y = a^k, \quad k \geq 1$

$x\,y\,z = a^m b^m b^m a^m$ $\qquad y = a^k, \quad k \geq 1$

From the Pumping Lemma: $x\,y^i\,z \in L$

$i = 0, 1, 2, \ldots$

Thus: $x\,y^2\,z \in L$

$$x\,y\,z = a^m b^m b^m a^m \qquad y = a^k, \quad k \geq 1$$

From the Pumping Lemma: $x\,y^2\,z \in L$

$$xy^2z = \overbrace{\underbrace{a...a}_{x}\underbrace{a...a}_{y}\underbrace{a...a}_{y}a...a}^{m+k}\overbrace{b...b}^{m}\overbrace{b...b}^{m}\overbrace{a...a}^{m} \in L$$

$$\underbrace{a...ab...bb...ba...a}_{z}$$

Thus: $a^{m+k}b^m b^m a^m \in L$

$$a^{m+k}b^m b^m a^m \quad \in L \qquad\qquad k \geq 1$$

BUT: $L = \{ww^R : w \in \Sigma^*\}$

$$\Downarrow$$

$$a^{m+k}b^m b^m a^m \quad \notin L$$

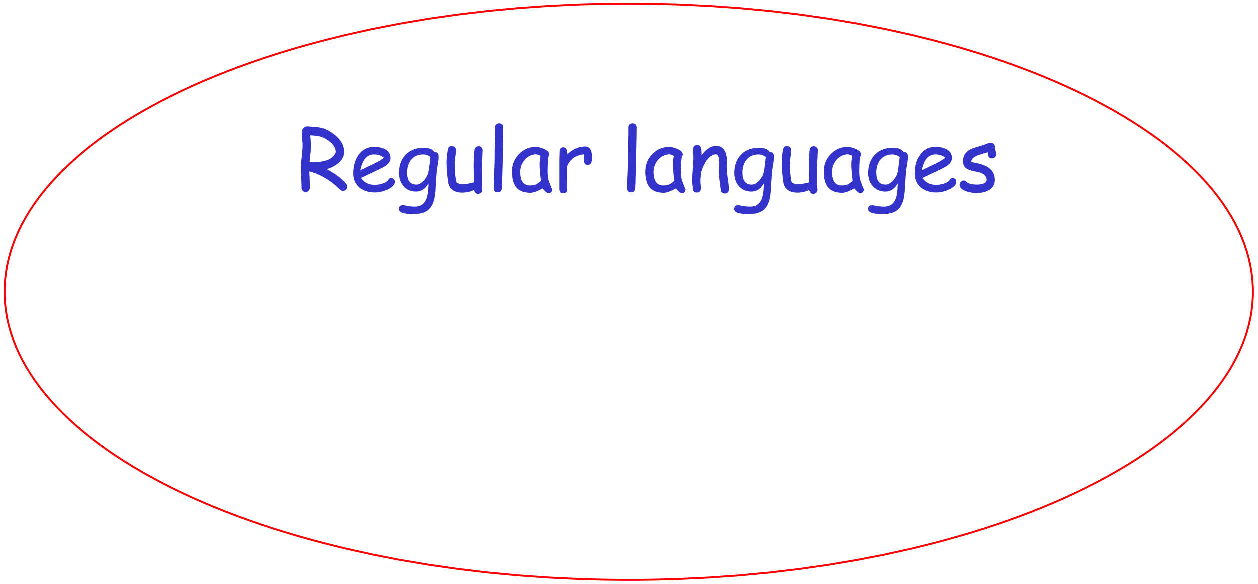
CONTRADICTION!!!

Therefore: Our assumption that L is a regular language is not true

Conclusion: L is not a regular language

Non-regular languages

$$L = \{a^n b^l c^{n+l} : \; n, l \geq 0\}$$



Theorem: The language

$$L = \{a^n b^l c^{n+l} : \ n, l \geq 0\}$$

is not regular

Proof: Use the Pumping Lemma

$$L = \{a^n b^l c^{n+l} : \; n, l \geq 0\}$$

Assume for contradiction
that L is a regular language

~~Since L is infinite~~
we can apply the Pumping Lemma

$$L = \{a^n b^l c^{n+l} : \ n, l \geq 0\}$$

Let m be the integer in the Pumping Lemma

Pick a string w such that: $w \in L$ and

length $|w| \geq m$

We pick $w = a^m b^m c^{2m}$

Write $a^m b^m c^{2m} = x\,y\,z$

From the Pumping Lemma

it must be that length $|x\,y| \le m, \quad |y| \ge 1$

$$xyz = \rlap{$\overbrace{}^{m}$}\underbrace{a\ldots a}_{x}\underbrace{a\ldots a}_{y}\underbrace{a\ldots a\overbrace{b\ldots b}^{m}\overbrace{c\ldots cc\ldots c}^{2m}}_{z}$$

Thus: $y = a^k, \quad k \ge 1$

$$x\ y\ z = a^m b^m c^{2m} \qquad y = a^k, \quad k \geq 1$$

From the Pumping Lemma: $x\ y^i\ z \in L$

$$i = 0, 1, 2, \ldots$$

Thus: $x\ y^0\ z = xz \in L$

$x\ y\ z = a^m b^m c^{2m}$ $\quad$ $y = a^k, \quad k \geq 1$

From the Pumping Lemma: $\quad xz \in L$

$$xz = \underbrace{\overbrace{a...aa...a}^{m-k}}_{x\ \text{(}a...a\text{)}}\overbrace{b...b}^{m}\overbrace{c...cc...c}^{2m} \in L$$

$$xz = \overbrace{\underbrace{a...a}_{x}\underbrace{a...a}_{}}^{m-k}\ \ldots \in L$$

Thus: $\quad a^{m-k} b^m c^{2m} \quad \in L$

$$a^{m-k}b^{m}c^{2m} \in L \qquad\qquad k \geq 1$$

BUT: $L = \{a^{n}b^{l}c^{n+l} : \ n, l \geq 0\}$

$$\Downarrow$$

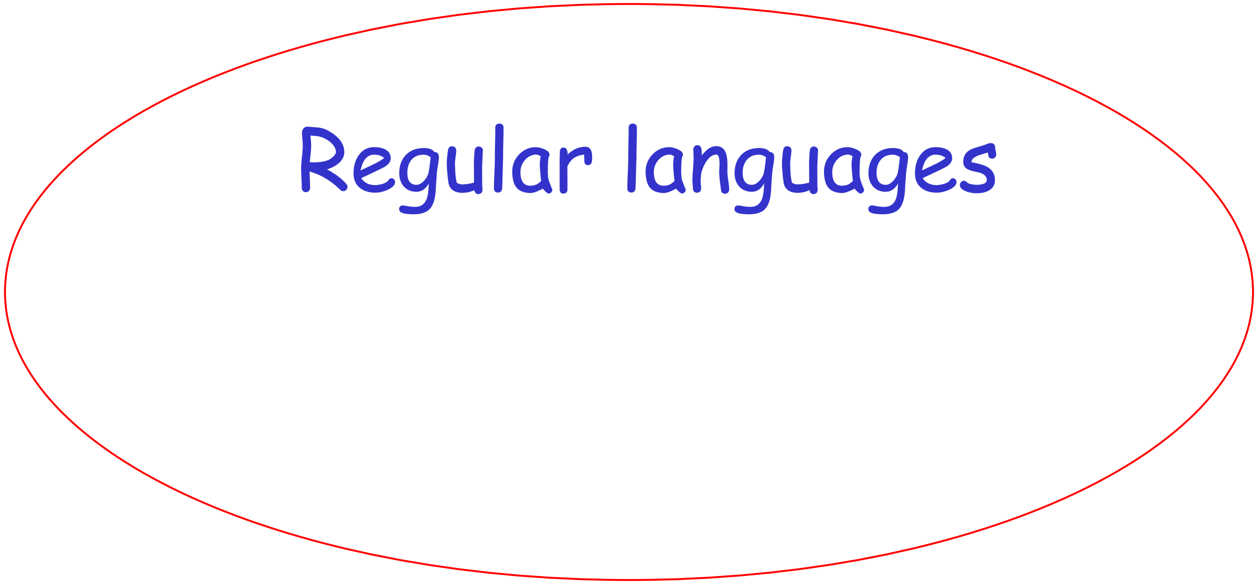
$$a^{m-k}b^{m}c^{2m} \notin L$$

CONTRADICTION!!!

Therefore: Our assumption that L is a regular language is not true

Conclusion: L is not a regular language

Non-regular languages $L=\{a^{n!}:\ n\geq 0\}$



Regular languages

Theorem: The language $L = \{a^{n!} : n \geq 0\}$

is not regular

$$n! = 1 \cdot 2 \cdots (n-1) \cdot n$$

Proof: Use the Pumping Lemma

$$L = \{a^{n!} : \; n \geq 0\}$$

Assume for contradiction
that L is a regular language

~~Since L is infinite~~
we can apply the Pumping Lemma

$$L = \{a^{n!} : \ n \geq 0\}$$

Let m be the integer in the Pumping Lemma

Pick a string w such that: $w \in L$

length $|w| \geq m$

We pick $w = a^{m!}$

Write $a^{m!} = x\ y\ z$

From the Pumping Lemma

it must be that length $|x\ y| \leq m, \quad |y| \geq 1$

$$xyz = a^{m!} = \overbrace{\underbrace{a\ldots a}_{x}\underbrace{a\ldots a}_{y}\mathrlap{\underbrace{}_{z}}a\ldots a}^{m}\overbrace{a\ldots aa\ldots a}^{m!-m}$$

Thus: $y = a^k, \quad 1 \leq k \leq m$

$x\,y\,z = a^{m!}$ $\quad$ $y = a^k, \quad 1 \leq k \leq m$

From the **Pumping Lemma:** $\quad x\,y^i\,z \in L$

$$i = 0, 1, 2, \ldots$$

Thus: $\quad x\,y^2\,z \in L$

$x\,y\,z = a^{m!}$ $\qquad$ $y = a^{k}, \quad 1 \le k \le m$

From the **Pumping Lemma:** $x\,y^{2}\,z \in L$

$$xy^{2}z = \overbrace{\underbrace{a...a}_{x}\underbrace{a...a}_{y}\underbrace{a...a}_{y}\mathrlap{\underbrace{}_{z}}a...a}^{m+k}\overbrace{a...aa...a}^{m!-m} \in L$$

Thus: $a^{m!+k} \in L$

$$a^{m!+k} \in L \qquad 1 \leq k \leq m$$

Since: $L = \{a^{n!}: \ n \geq 0\}$

$$\Downarrow$$

There must exist p such that:

$$m!+k = p!$$

However: $m!+k \le m!+m$ for $m > 1$

$$
\begin{aligned}
m!+k &\le m!+m \\
&\le m!+m! \\
&< m!m + m! \\
&= m!(m+1) \\
&= (m+1)!
\end{aligned}
$$

$$\Downarrow$$

$$m!+k < (m+1)!$$

$$\Downarrow$$

$m!+k \neq p!$ for any p

$$a^{m!+k} \in L \qquad 1 \leq k \leq m$$

BUT: $L = \{a^{n!} : n \geq 0\}$

$$\Downarrow$$

$$a^{m!+k} \notin L$$

CONTRADICTION!!!

Therefore: Our assumption that L is a regular language is not true

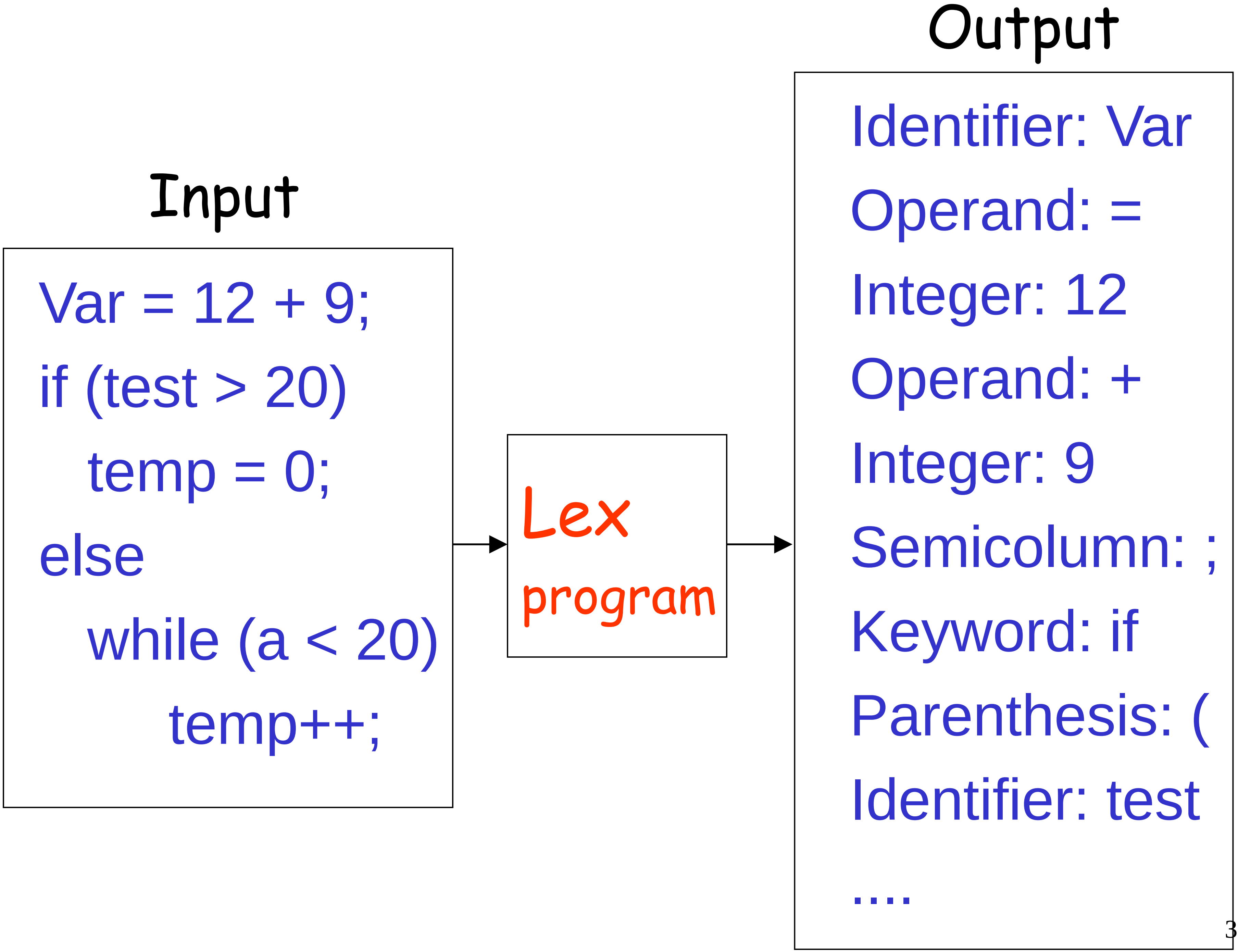
Conclusion: L is not a regular language

Lex

Lex: a lexical analyzer

- A Lex program recognizes strings

- For each kind of string found
  the lex program takes an action



Input

```
Var = 12 + 9;
if (test > 20)
   temp = 0;
else
   while (a < 20)
        temp++;
```

Lex program

Output

```
Identifier: Var
Operand: =
Integer: 12
Operand: +
Integer: 9
Semicolumn: ;
Keyword: if
Parenthesis: (
Identifier: test
....
```

In Lex strings are described with regular expressions

Lex program

Regular expressions

```
"+"
"-"                /* operators */
"="

"if"
"then"             /* keywords */
```

Lex program

Regular expressions

(0|1|2|3|4|5|6|7|8|9)+ /* integers */

(a|b|..|z|A|B|...|Z)+ /* identifiers */

integers

(0|1|2|3|4|5|6|7|8|9)+

[0-9]+

identifiers

(a|b|..|z|A|B|...|Z)+

[a-zA-Z]+

Each regular expression has an associated action (in C code)

Examples:

Regular expression	Action
\n	linenum++;
[0-9]+	prinf("integer");
[a-zA-Z]+	printf("identifier");

Default action: ECHO;

Prints the string identified to the output

A small program

```
%%

[ \t\n]              ;   /*skip spaces*/

[0-9]+               printf(“Integer\n”);

[a-zA-Z]+            printf(“Identifier\n”);
```

Input

```
1234    test

var  566     78

9800
```

Output

```
Integer
Identifier
Identifier
Integer
Integer
Integer
```

Another program

```
%{
int linenum = 1;
%}

%%

[ \t]                  ;    /*skip spaces*/
\n                     linenum++;

[0-9]+                 prinf("Integer\n");

[a-zA-Z]+              printf("Identifier\n");

.                      printf("Error in line: %d\n",
                               linenum);
```

Input

```
1234    test
var  566     78
9800   +
temp
```

Output

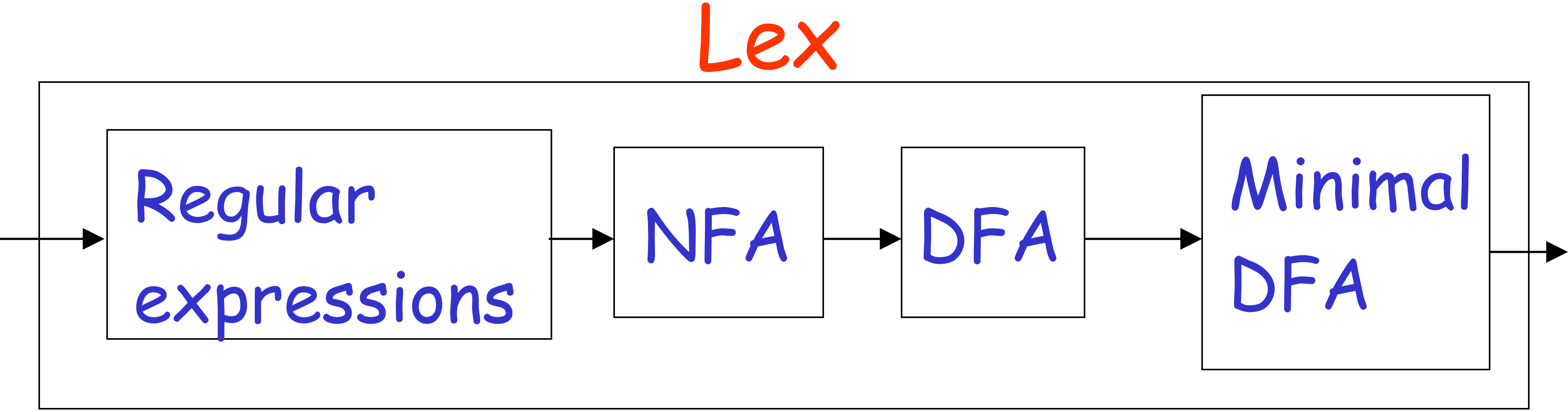
```
Integer
Identifier
Identifier
Integer
Integer
Integer
Error in line: 3
Identifier
```

Lex matches the longest input string

Example: Regular Expressions “if”
“ifend”

Input:	ifend	if	ifn
Matches:	“ifend”	“if”	nomatch

Internal Structure of Lex



Lex

Regular expressions

NFA

DFA

Minimal DFA

The final states of the DFA are associated with actions