بسم الله الرحمن الرحیم

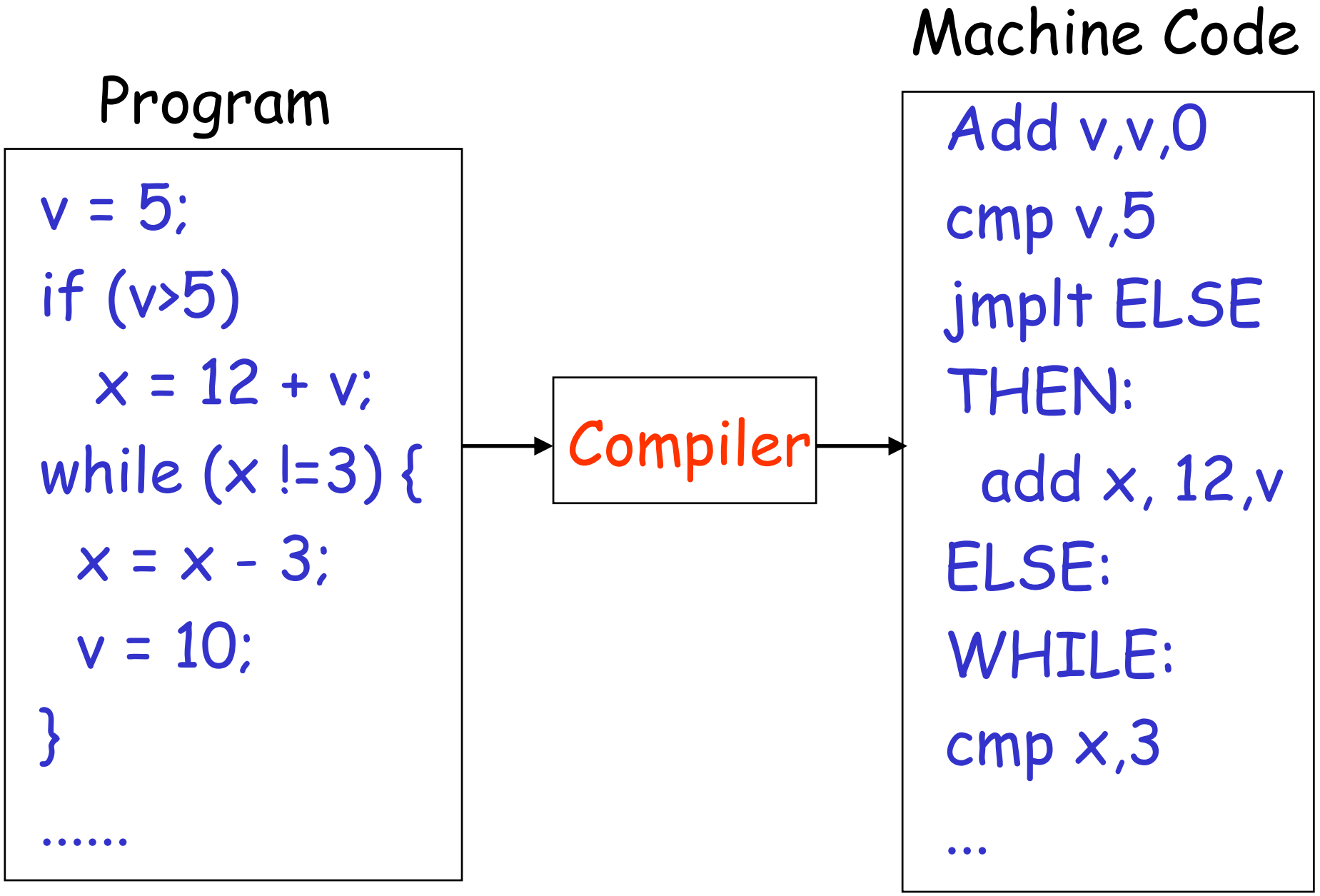
فصل پنجم

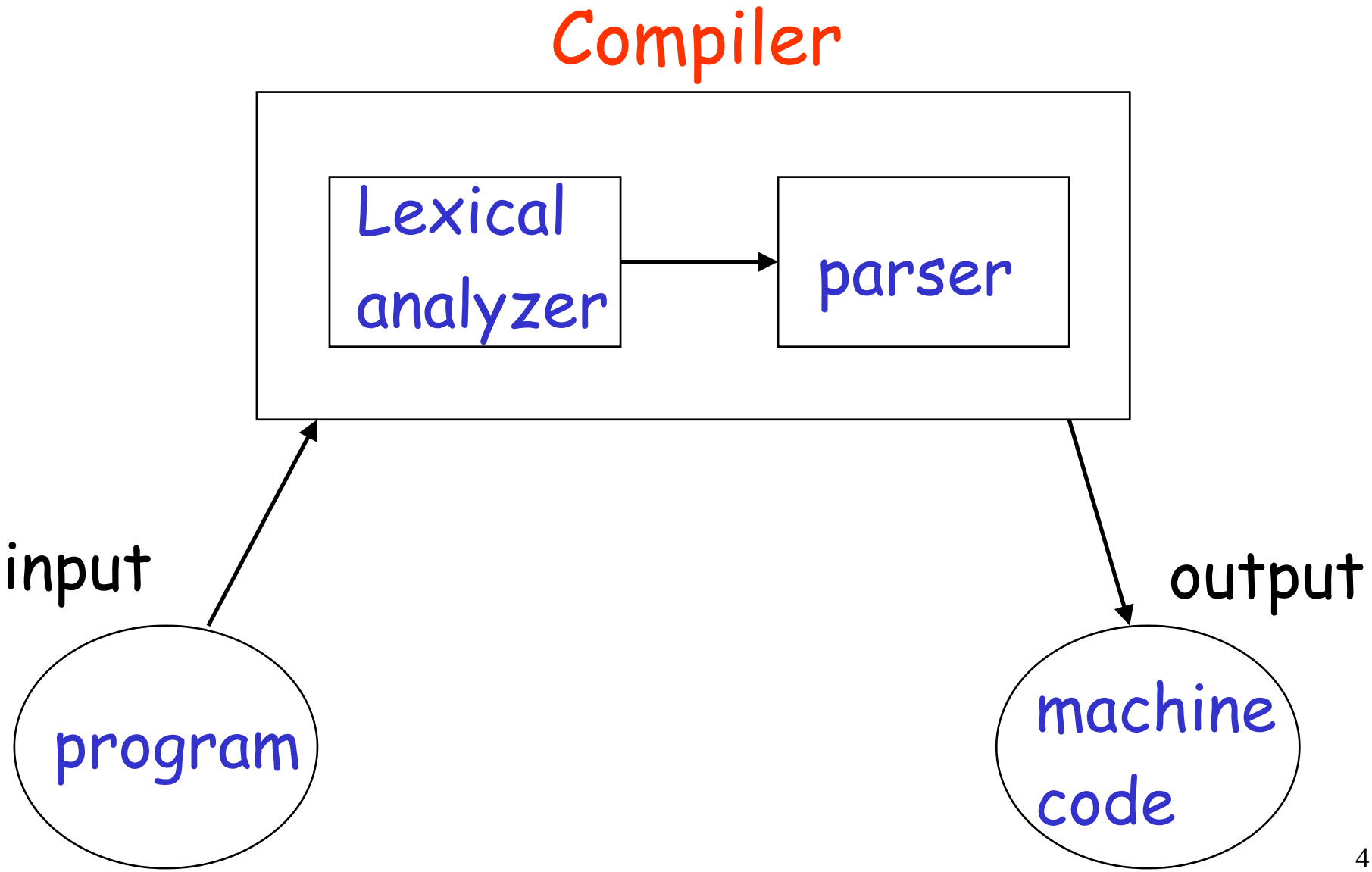
زبان‌های مستقل از متن (۲)

Context-Free Languages (2)

کاظم فولادی
kazim@fouladi.ir
دانشکده‌ی مهندسی برق و کامپیوتر
دانشگاه تهران

Compilers


Program
v = 5;
if (v>5)
x = 12 + v;
while (x !=3) {
x = x - 3;
v = 10;
}
.....
Compiler
Machine Code
Add v,v,0
cmp v,5
jmplt ELSE
THEN:
add x, 12,v
ELSE:
WHILE:
cmp x,3
...


Compiler
Lexical analyzer
parser
input
output
program
machine code

A parser knows the grammar of the programming language

Parser

PROGRAM → STMT_LIST

STMT_LIST → STMT; STMT_LIST | STMT;

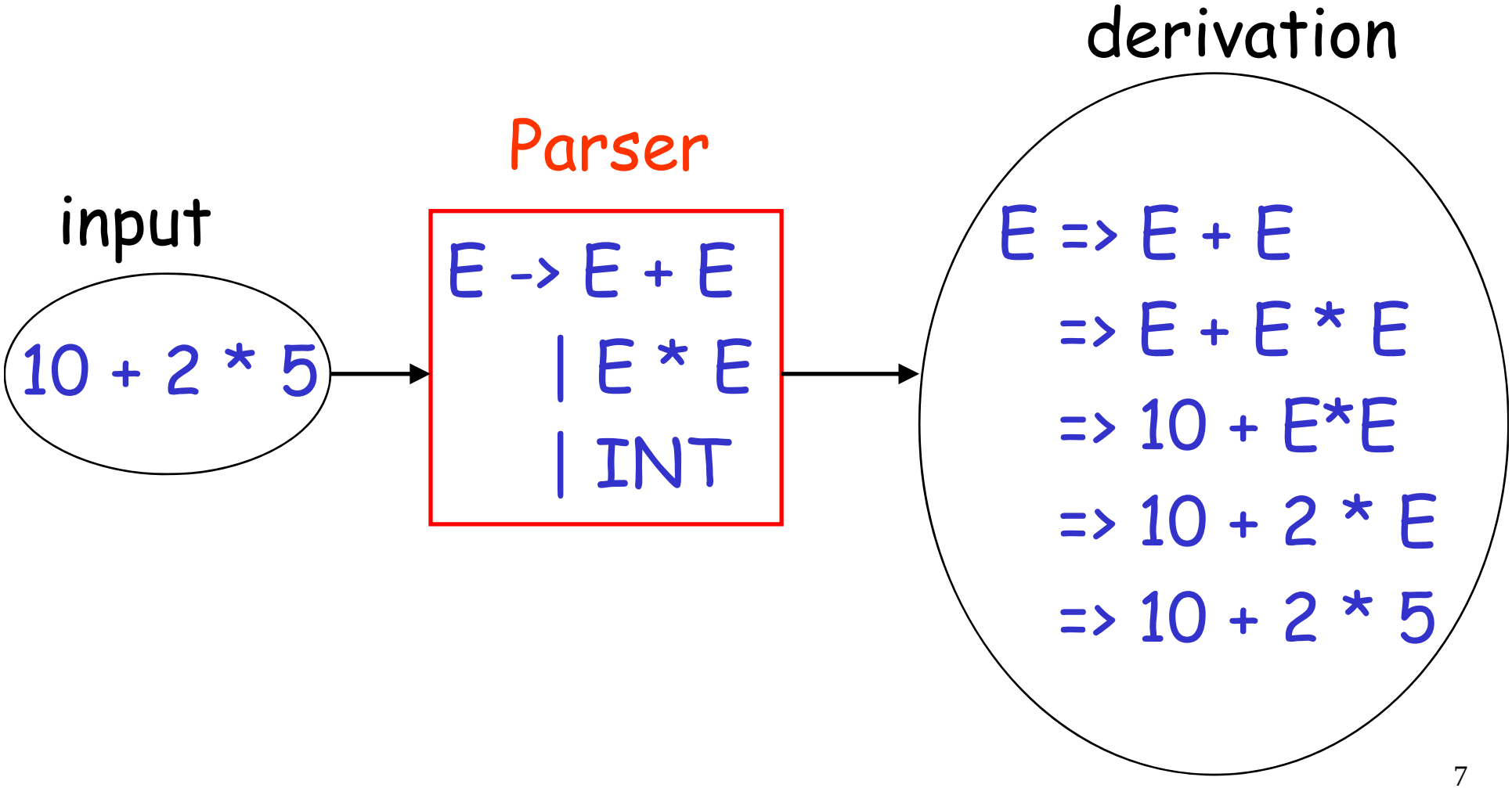
STMT → EXPR | IF_STMT | WHILE_STMT
| { STMT_LIST }

EXPR → EXPR + EXPR | EXPR - EXPR | ID

IF_STMT → if (EXPR) then STMT
| if (EXPR) then STMT else STMT

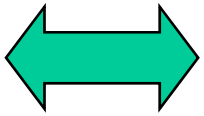
WHILE_STMT → while (EXPR) do STMT

The parser finds the derivation of a particular input

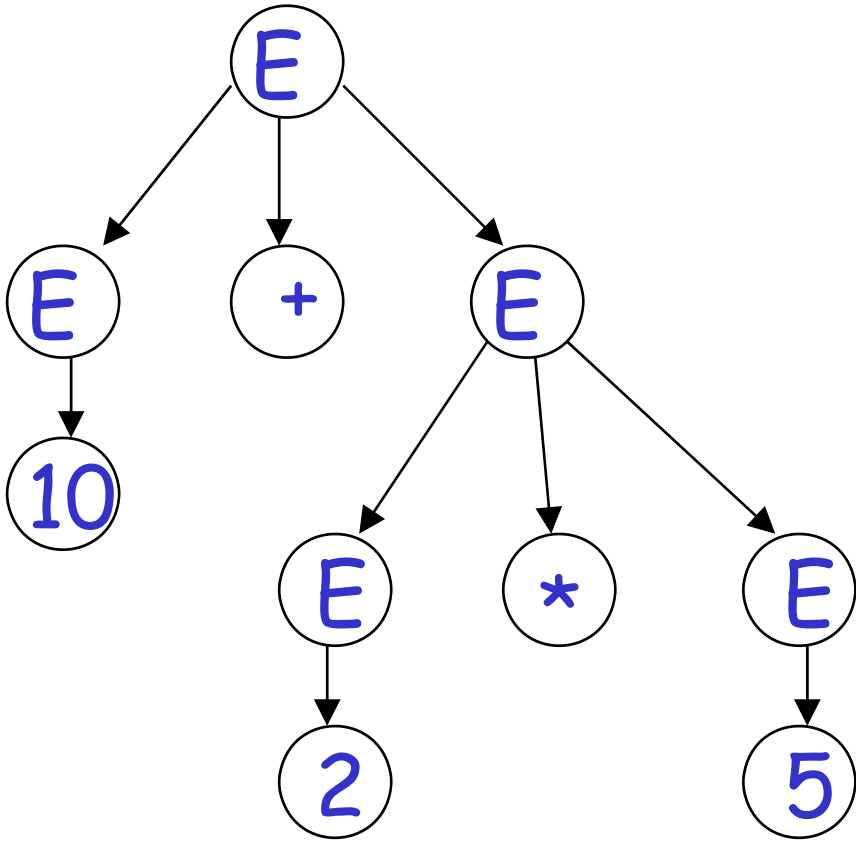


derivation

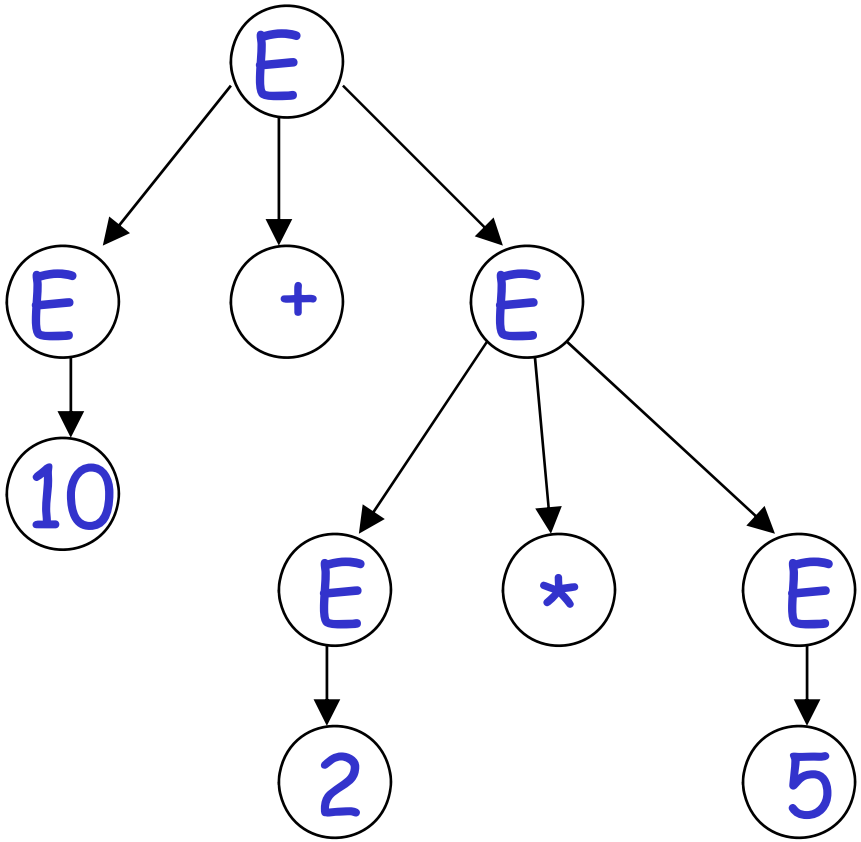
E => E + E
=> E + E * E
=> 10 + E*E
=> 10 + 2 * E
=> 10 + 2 * 5



derivation tree



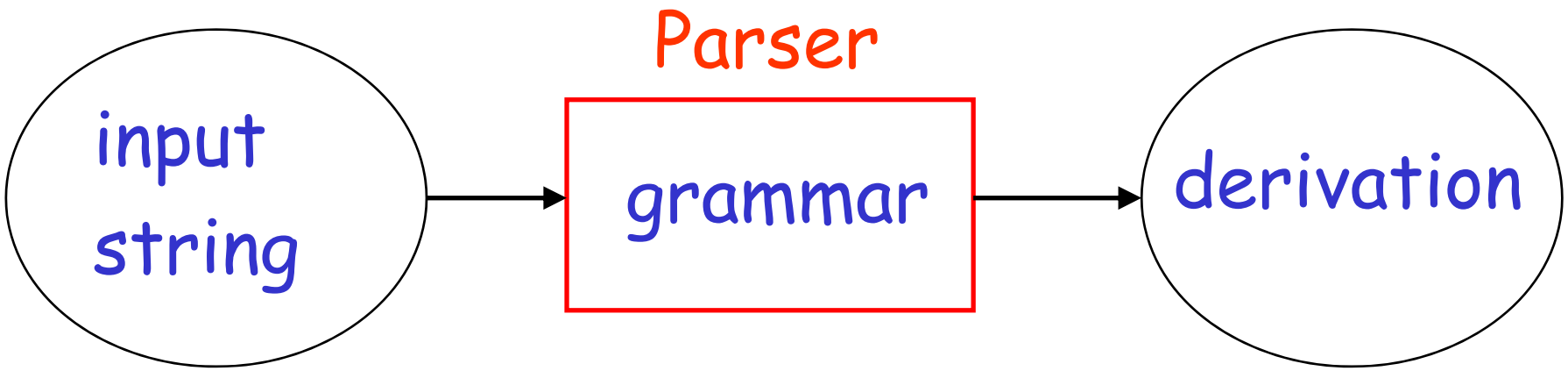
derivation tree



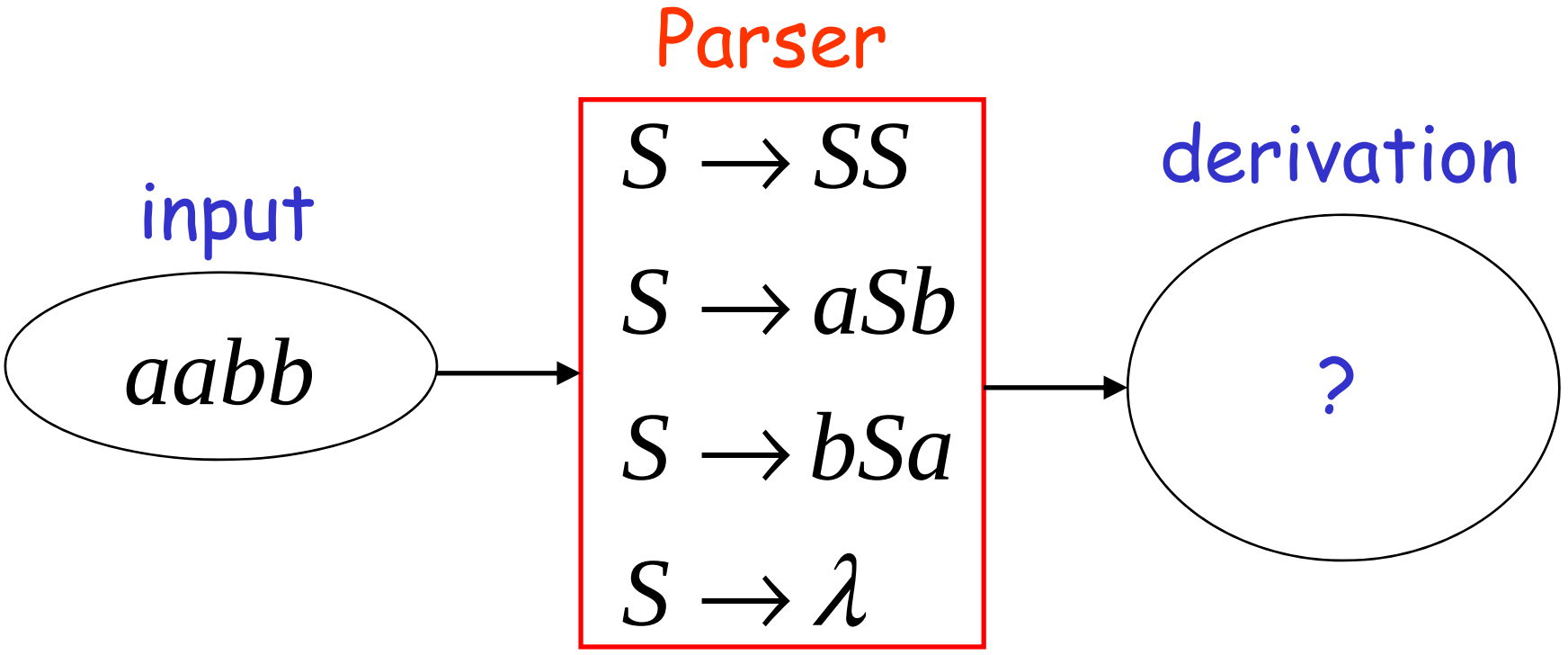
machine code

```
mult a, 2, 5
add b, 10, a
```

Parsing


Parser
input string
grammar
derivation

Example:


Parser
input
aabb
$S \to SS$
$S \to aSb$
$S \to bSa$
$S \to \lambda$
derivation
?

Exhaustive Search

$$S \to SS \mid aSb \mid bSa \mid \lambda$$

Phase 1:

$$S \Rightarrow SS$$

$$S \Rightarrow aSb$$

$$S \Rightarrow bSa$$

$$S \Rightarrow \lambda$$

Find derivation of

$$aabb$$

All possible derivations of length 1

$S \Rightarrow SS$

$S \Rightarrow aSb$

~~$S \Rightarrow bSa$~~

~~$S \Rightarrow \lambda$~~

$aabb$

Phase 2 $S \to SS \mid aSb \mid bSa \mid \lambda$

Phase 1

$S \Rightarrow SS$

$S \Rightarrow SS \Rightarrow SSS$

$S \Rightarrow SS \Rightarrow aSbS$

~~$S \Rightarrow SS \Rightarrow bSaS$~~

$S \Rightarrow SS \Rightarrow S$

$aabb$

$S \Rightarrow aSb$

$S \Rightarrow aSb \Rightarrow aSSb$

$S \Rightarrow aSb \Rightarrow aaSbb$

~~$S \Rightarrow aSb \Rightarrow abSab$~~

~~$S \Rightarrow aSb \Rightarrow ab$~~

$$S \to SS \mid aSb \mid bSa \mid \lambda$$

Phase 2

$$S \Rightarrow SS \Rightarrow SSS$$

$$S \Rightarrow SS \Rightarrow aSbS$$

$$S \Rightarrow SS \Rightarrow S$$

$$aabb$$

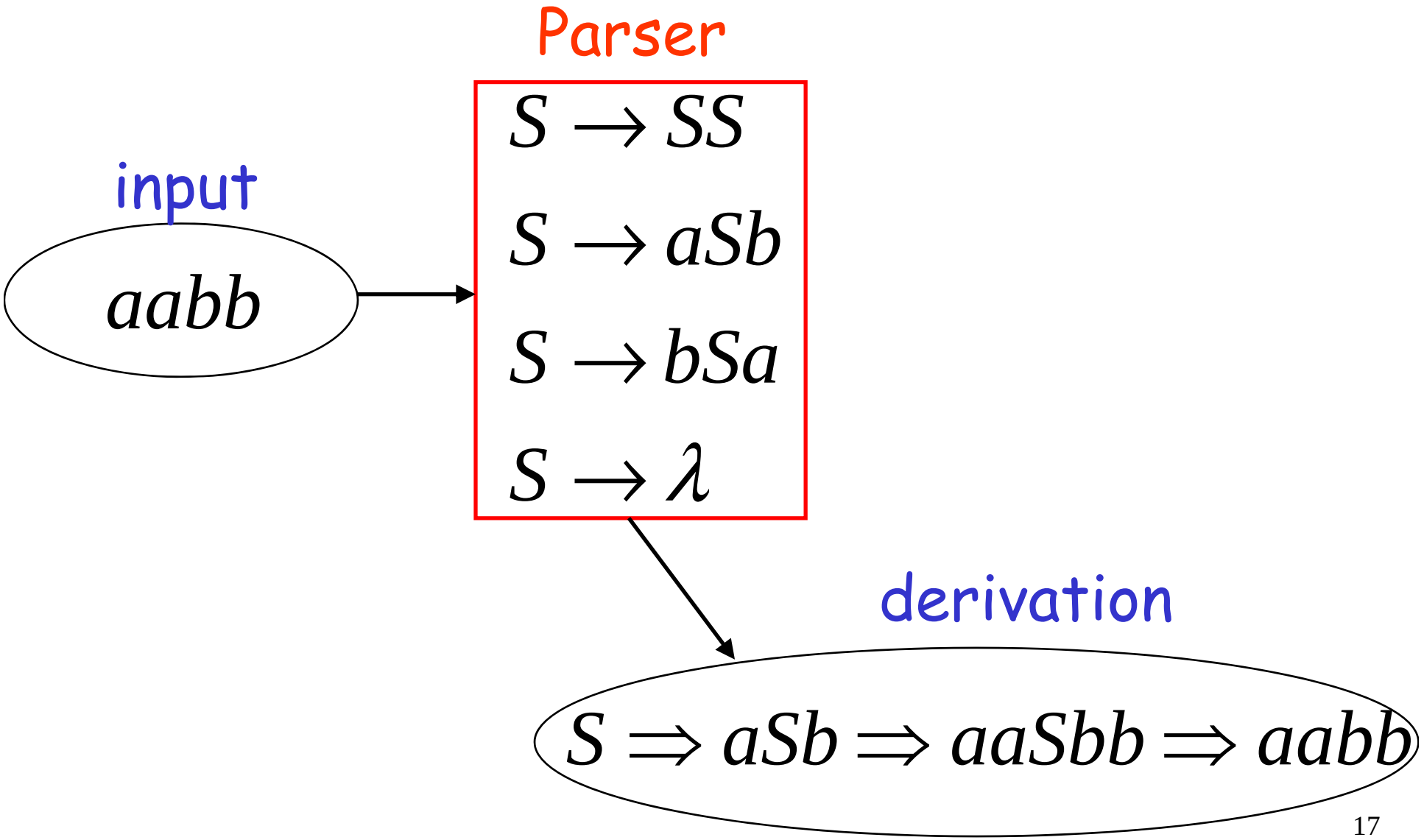
$$S \Rightarrow aSb \Rightarrow aSSb$$

$$S \Rightarrow aSb \Rightarrow aaSbb$$

Phase 3

$$S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aabb$$

Final result of exhaustive search (top-down parsing)



Time complexity of exhaustive search

Suppose there are no productions of the form

$$A \to \lambda$$

$$A \to B$$

Number of phases for string w : $2|w|$

For grammar with k rules

Time for phase 1: k

k possible derivations

Time for phase 2: k^2

k^2 possible derivations

Time for phase $2|w|$: $k^{2|w|}$

$k^{2|w|}$ possible derivations

Total time needed for string w:

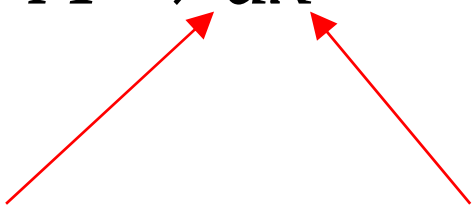
$$k + k^2 + \cdots + k^{2|w|}$$

phase 1 phase 2 phase 2|w|

Extremely bad!!!

There exist faster algorithms for specialized grammars

S-grammar: $A \to ax$



symbol

string of variables

Pair (A, a) appears once

S-grammar example:

$$S \to aS$$

$$S \to bSS$$

$$S \to c$$

Each string has a unique derivation

$$S \Rightarrow aS \Rightarrow abSS \Rightarrow abcS \Rightarrow abcc$$

For S-grammars:

In the exhaustive search parsing
there is only one choice in each phase

Time for a phase: 1

Total time for parsing string w: $|w|$

For general context-free grammars:

There exists a parsing algorithm that parses a string $|w|$ in time $|w|^3$