

Presented By

Veena Mumtaz



Topic:

Parsing



dreamstime.com

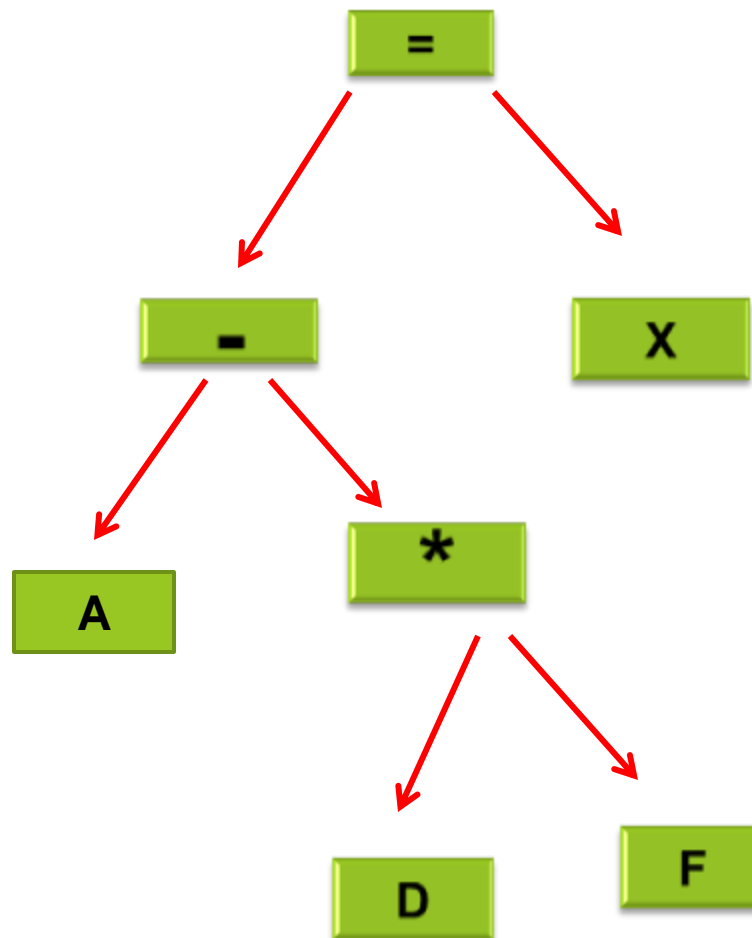
What is “PARSING”

- The term *parsing* comes from Latin *pars* meaning part **Parsing or syntactic analysis** is the process of analyzing a string of symbols, either in natural language or in computer languages according to the rules of a formal grammar.
- So **Parsing** means determining the syntactic structure of an expression



Example of parsing

□ $X = (A - (D * F))$



Types of Parser

- The task of the parser is essentially to determine if and how the input can be derived from the start symbol of the grammar. This can be done in essentially two ways

Top-down

1) **Top-down parser**

2) **Bottom-up parser**



Bottom-up

Top-down parsers

- Top-down parsing expands a parse tree from the start symbol to the leaves
 - ▣ Always expand the leftmost non-terminal

E

id

+

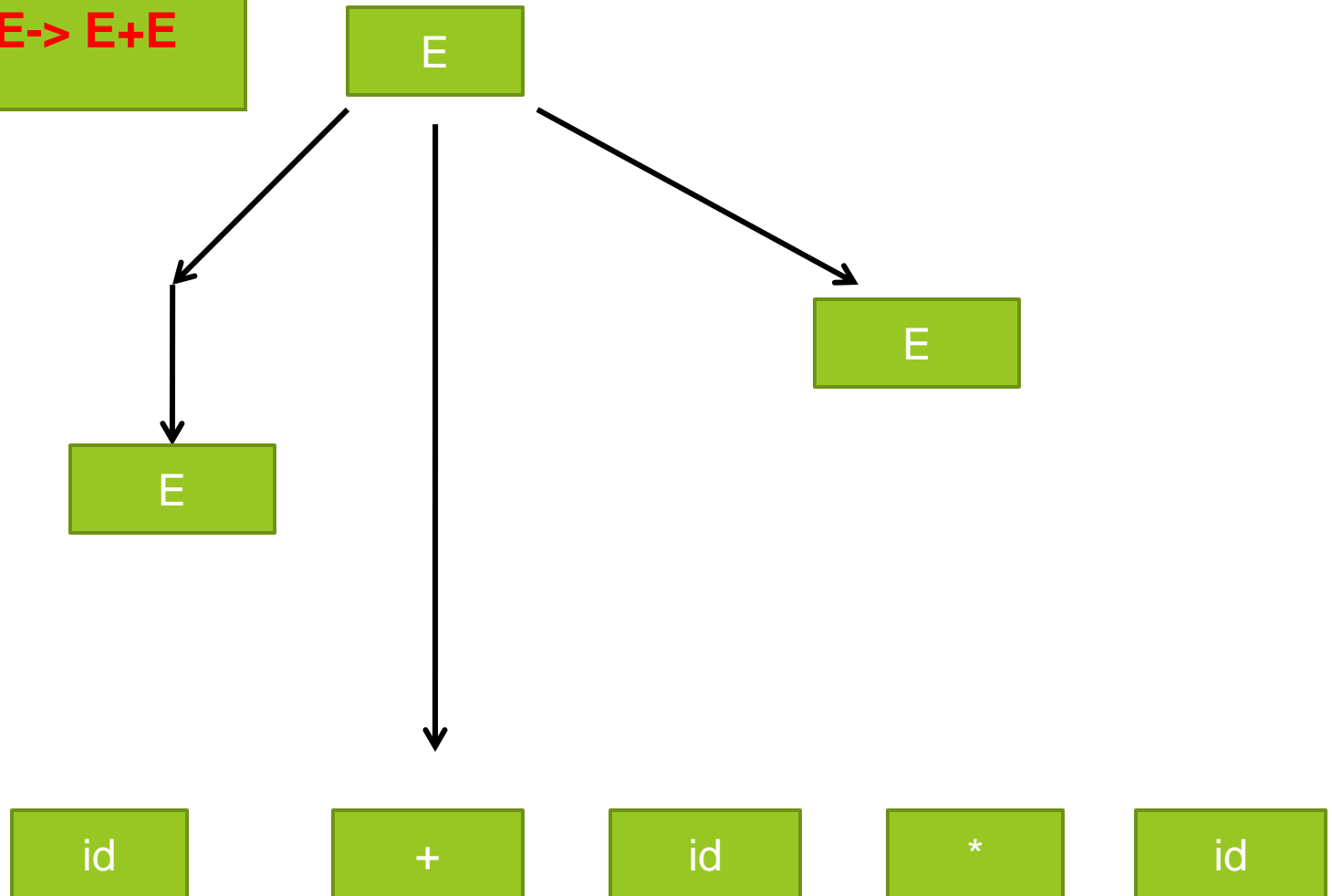
id

*

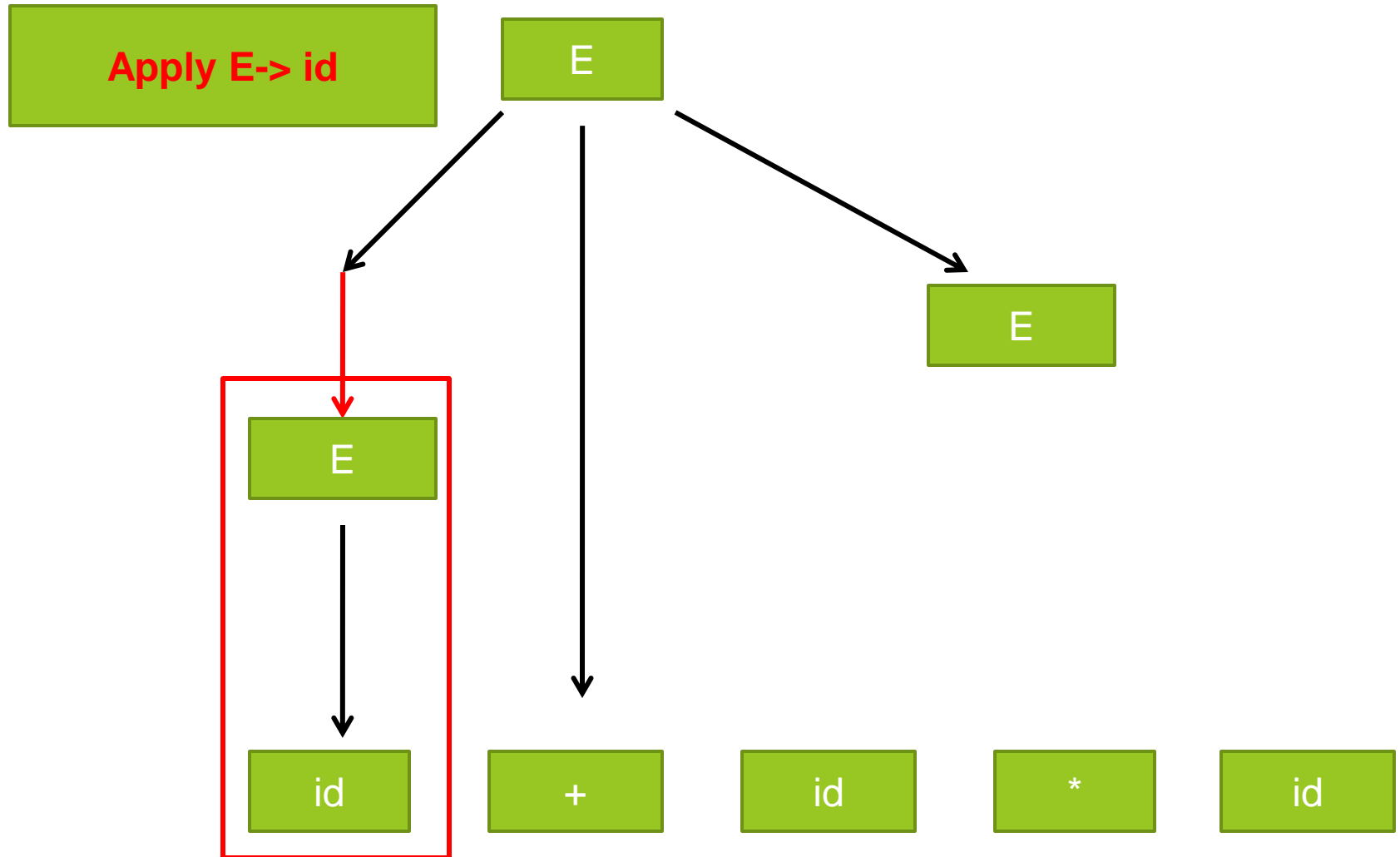
id

Top-down parsers

Apply $E \rightarrow E + E$



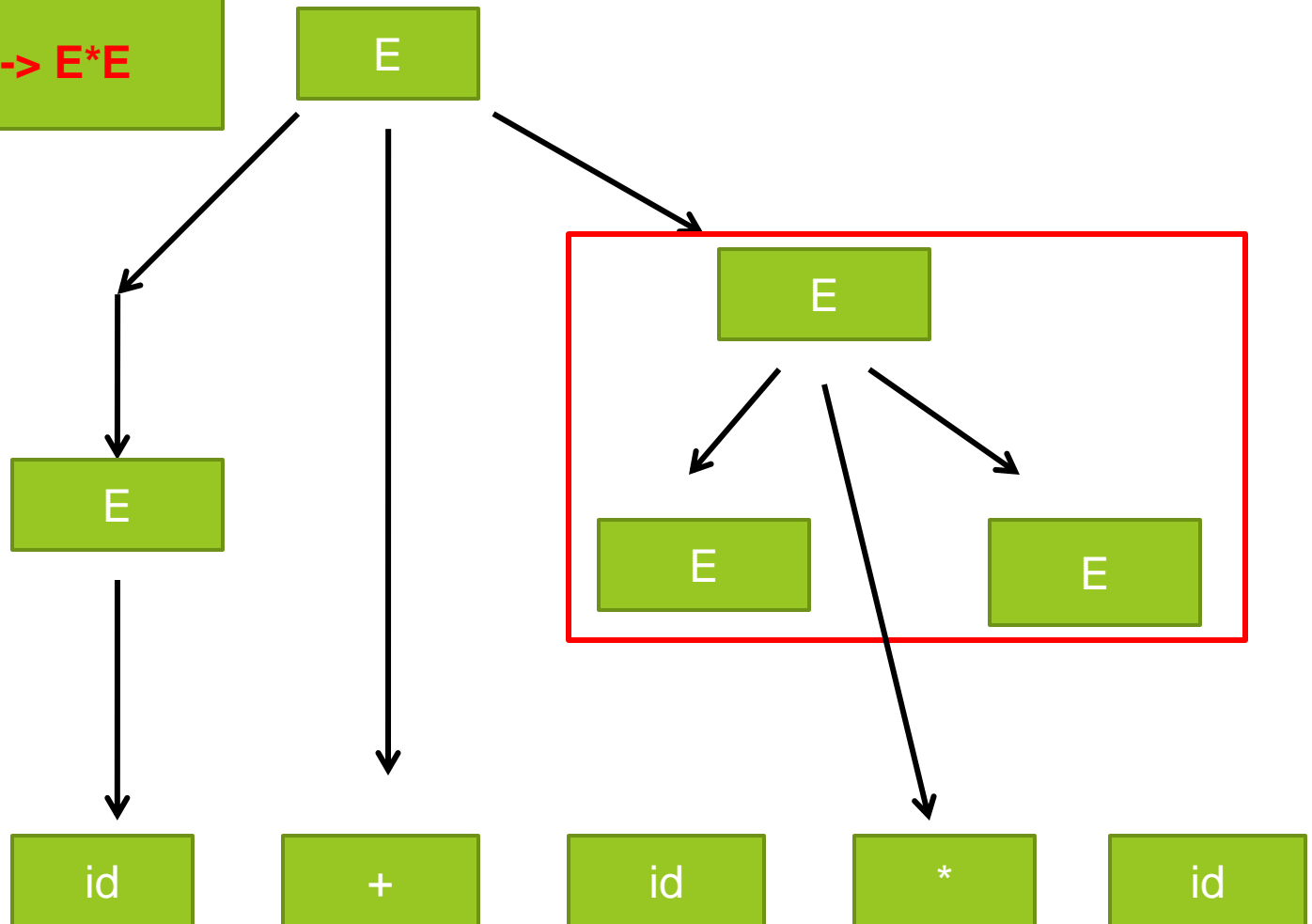
Top-down parsers





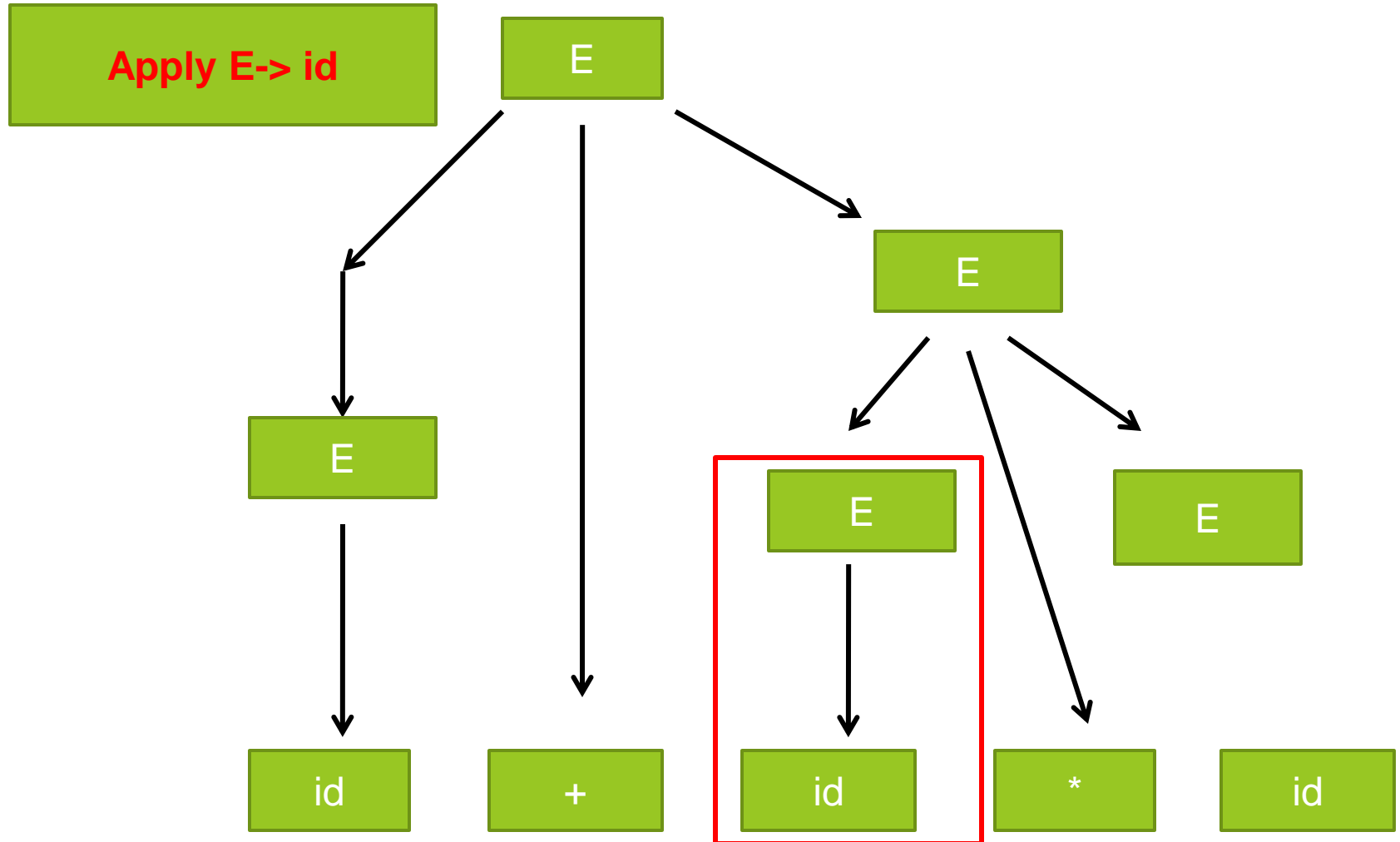
Top-down parsers

Apply $E \rightarrow E * E$

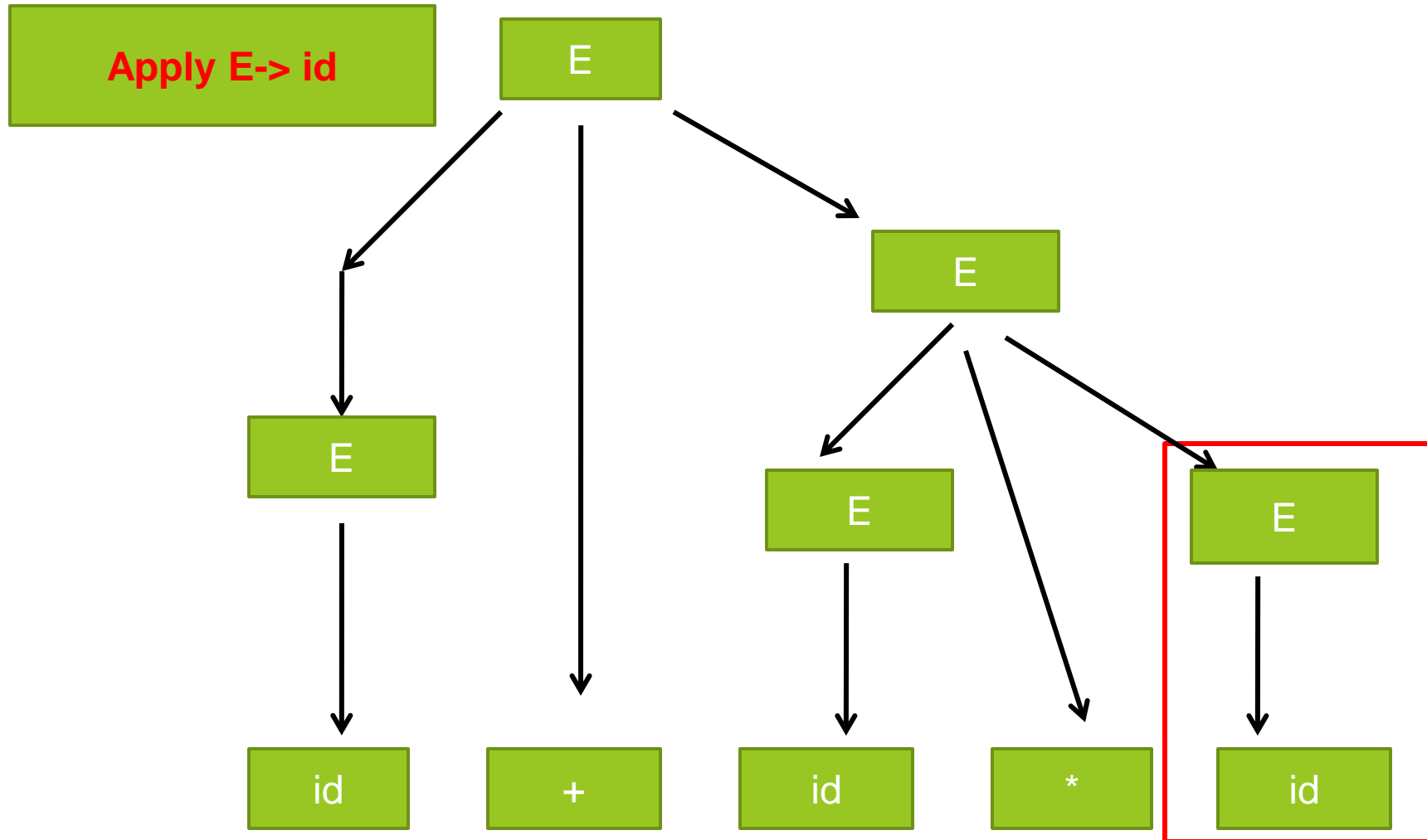




Top-down parsers



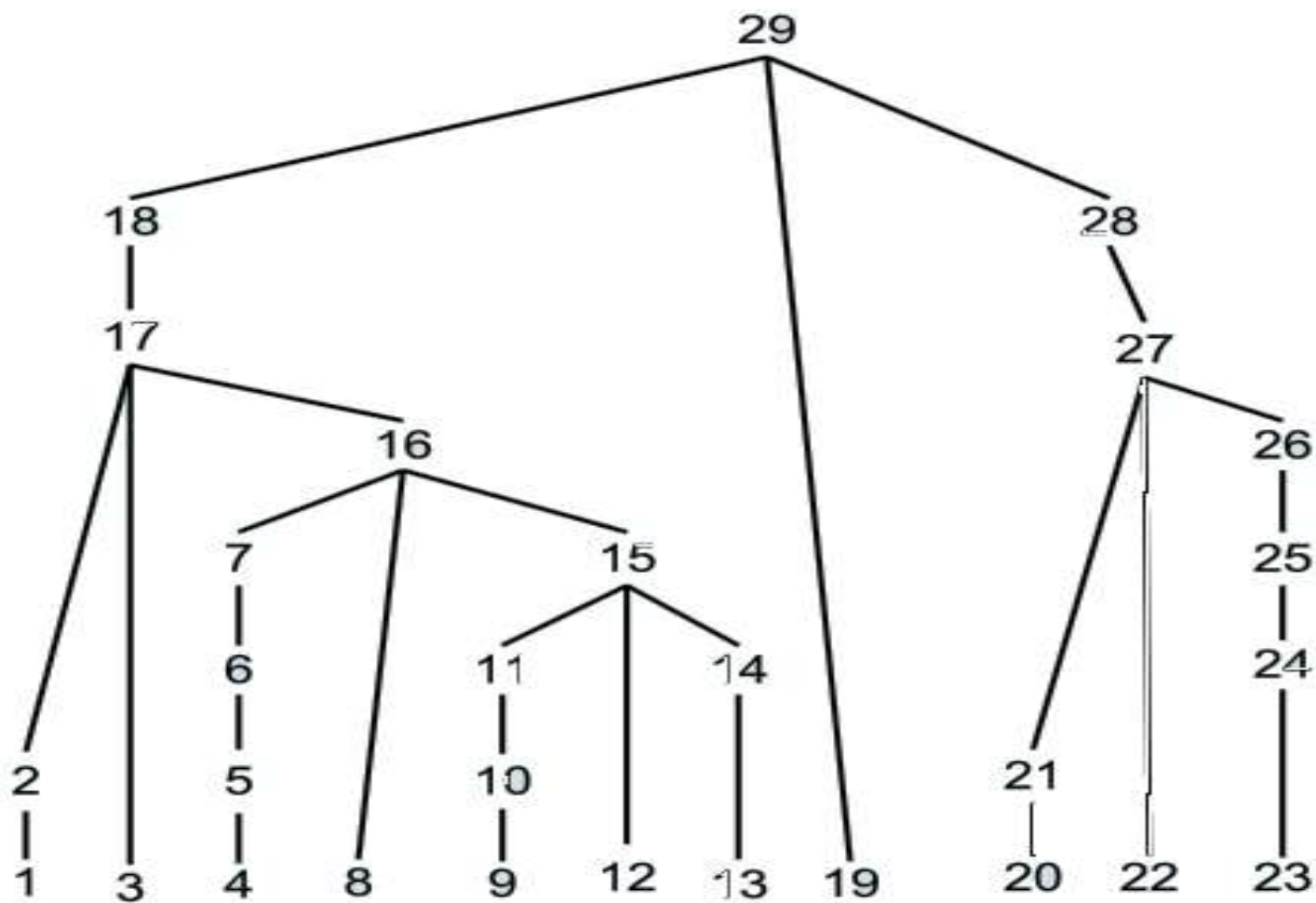
Top-down parsers



Bottom-up Parser

- **Start at the leaves and grow toward root**
 - ▣ And just as efficient
 - ▣ Builds on ideas in top-down parsing
 - ▣ Preferred method in practice
- **Also called LR parsing**
 - ▣ L means that tokens are read left to right
 - ▣ R means that it constructs a rightmost derivation -->

BOTTOM-UP PARSER



Example of Bottom-up Parse

14

int + (int) + (int)

int + (int) + (int)

Example of Bottom-up Parse

15

int + (int) + (int)

E + (int) + (int)

E
|
int + (int) + (int)

Example of Bottom-up Parse

16

int + (int) + (int)

E + (int) + (int)

E + (E) + (int)

E
|
int + (**E**
|
int) + (int)

Example of Bottom-up Parse

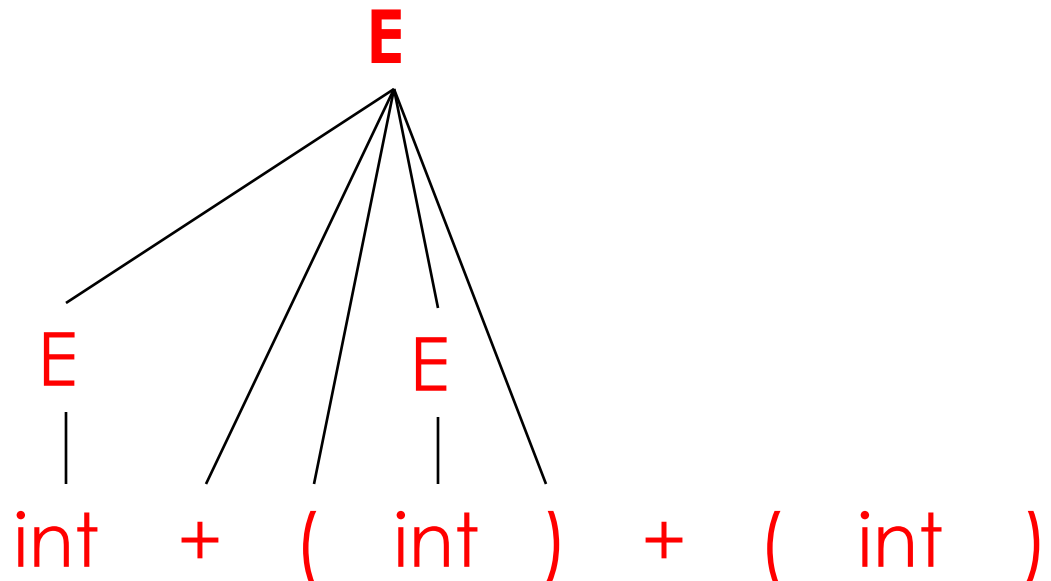
17

int + (int) + (int)

E + (int) + (int)

E + (E) + (int)

E + (int)



Example of Bottom-up Parse

18

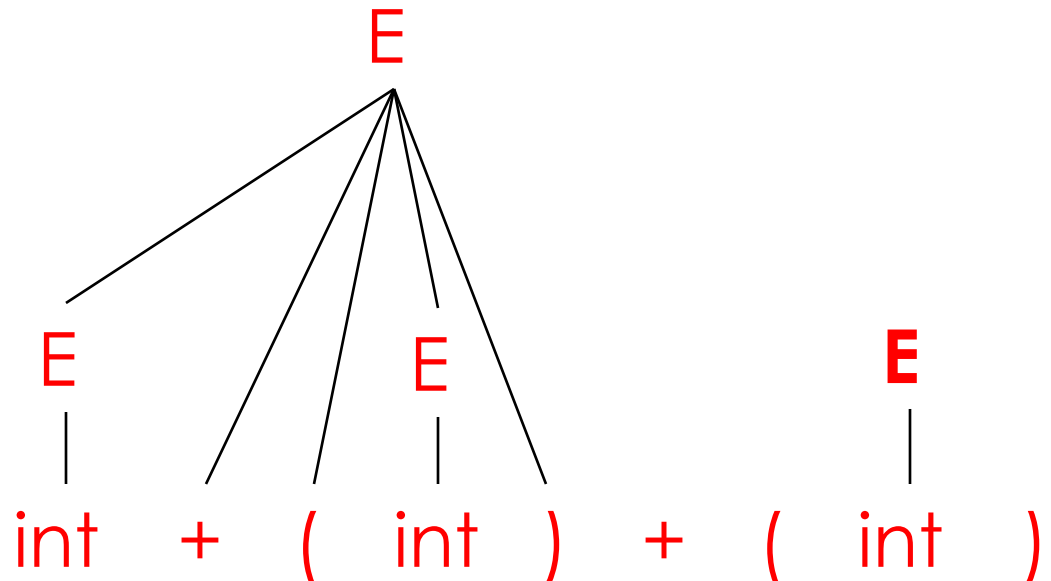
int + (int) + (int)

E + (int) + (int)

E + (E) + (int)

E + (int)

E + (E)



19

E

