Simple grammer for arithmetic expression.

G:
- $S \rightarrow expr + expr$ (1)
- $expr \rightarrow expr - expr$ (2)
- $expr \rightarrow digit$.
- $digit \rightarrow 0|1|\cdots|9$

Derivation.

start: S

derivation ↓

(1) $expr + expr$

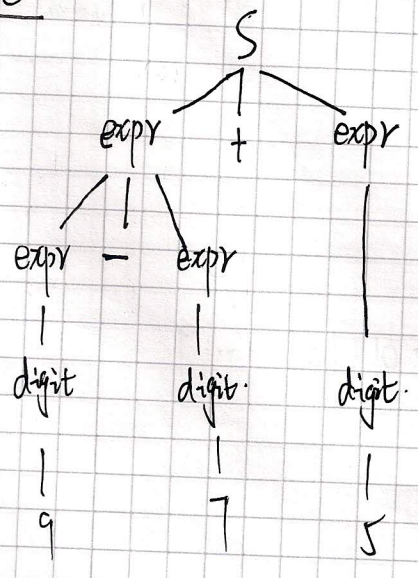
(2) $expr - expr + expr$

(3) $digit - expr + expr$
$digit - digit + expr$
$digit - digit + digit$.

(4) $9 - digit + digit$.
$9 - 7 + digit$.

(5) $9 - 7 + 5$

Parse Tree:



Derivation

$G \overset{*}{\Longrightarrow} 9-7+5$

Notice: G_1 generates a subset of arithmetic expressions with +, −

Example: Simple grammar for ~~arithmetic~~ arithmetic expressions with +, -, *, /

G_1:

$S \to expr$ (1)

$expr \to expr + expr$ (2)

$| expr - expr$ (3)

$expr \to expr * expr$ (4)

$expr \to digit$ (5)

$digit \to 0 | 1 | \cdots | 9$ (6)

Example ~~start: S~~ 7+6*9

G_2:

(1) start: S

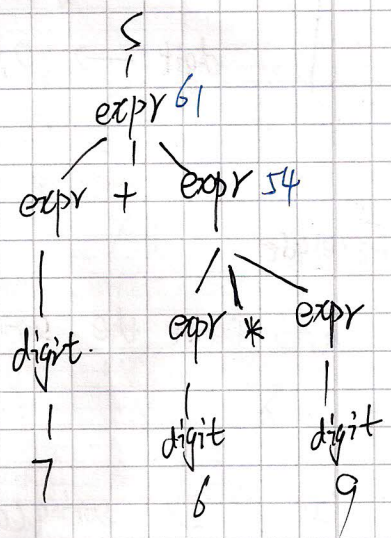
a. expr

b. expr + expr

c. expr + expr * expr

(5)(5)(5) digit + digit * digit.

(6)(6)(6) 7 + 6 * 9



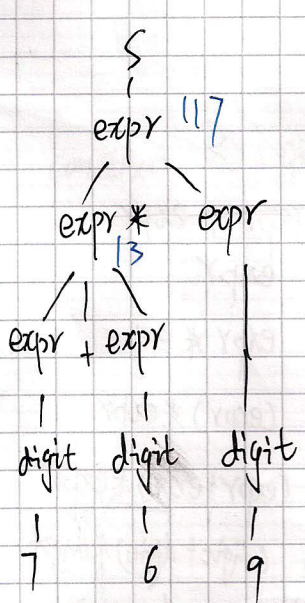
(2) start: S. a. expr

c. expr * expr

b. expr + expr * expr.

(5)(5)(5) digit + digit * digit.

(6)(6)(6) 7 6 9



The grammar G_2 is ambiguous: Here are more than 1 parse tree of a given string.

Notice: Any context-free grammars might be ambiguous.

How can we resolve "ambiguous"?

Strategy I: use brackets.

G_3:

$S \rightarrow$ ~~(expr)~~ expr (1)

expr $\rightarrow$ (expr) (2)

expr $\rightarrow$ expr + expr (3)

expr $\rightarrow$ expr * expr (4)

expr $\rightarrow$ digit (5)

digit $\rightarrow$ 0 | 1 | ... | 9 (6)

Excercise

(1) Derive the strings (a) (7+6)*9

(b) 7+6*9

(2) Is string (a) ambiguous.

Answer:

(a) Start: S

~~(4) expr * expr~~

(1) expr

(4) expr * expr

(2) (expr) * expr

(3) (expr + expr) * expr

(5)(5)(5) (digit + digit) * digit

(6)(6)(6) (7+6) * 9

(b) Start: S

(1) expr

(3) expr + expr

(4) expr + expr * expr

(5)(5)(5) digit + digit * digit.

(6)(6)(6) 7 + 6 * 9.

ambiguous.

(1) expr

(4) expr * expr

(3) expr + expr * ex

(5)(5)(5) digit + digit * e

(6)(6)(6) 7 + 6 *