

## Context Free Grammar (CFG)

*Team Sains  
Komputasi  
Prodi Ilmu Komputer*



**Definition:** A context-free grammar (CFG) consisting of a finite set of grammar rules is a quadruple  $(N, T, P, S)$  where

- $N$  is a set of non-terminal symbols.
- $T$  is a set of terminals where  $N \cap T = \text{NULL}$ .
- $P$  is a set of rules,  $P: N \rightarrow (N \cup T)^*$ , i.e., the left-hand side of the production rule  $P$  does not have any right context or left context.
- $S$  is the start symbol.

### Example

- The grammar  $(\{A\}, \{a, b, c\}, P, A)$ ,  $P: A \rightarrow aA, A \rightarrow abc$ .
- The grammar  $(\{S, a, b\}, \{a, b\}, P, S)$ ,  $P: S \rightarrow aSa, S \rightarrow bSb, S \rightarrow \epsilon$
- The grammar  $(\{S, F\}, \{0, 1\}, P, S)$ ,  $P: S \rightarrow 00S \mid 11F, F \rightarrow 00F \mid \epsilon$

# Sentensial

- **Turunan** (*derivation*) yang masih memiliki **variabel** (*non-terminal*) disebut dengan **bentuk sentensial**.

- Contoh:

$S \Rightarrow \mathbf{A}b$



A adalah Non Terminal

# Contoh Sentensial

- Dari aturan produksi (P):

$$S \rightarrow \mathbf{A1} \mid \mathbf{0B}$$

$$A \rightarrow 0$$

$$B \rightarrow 1$$

- $S \Rightarrow A1$  merupakan bentuk sentensial dari P.
- $S \Rightarrow 0B$  merupakan bentuk sentensial dari P.
- Karena **A** dan **B** bukanlah Non Terminal

# Sentens

- **Turunan** (*derivation*) yang hanya memiliki **terminal** disebut dengan **sentens**.

- Contoh:

$$S \Rightarrow b$$

$$S \Rightarrow 0$$

$$S \Rightarrow 1$$

# Contoh Sentens

- Dari aturan produksi (P):

$$S \rightarrow A1 \mid 0B$$

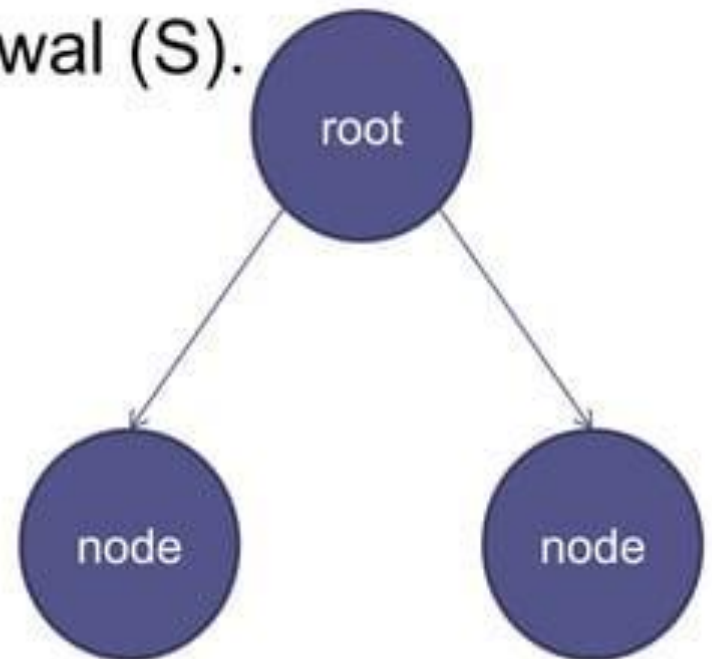
$$A \rightarrow 0$$

$$B \rightarrow 1$$

- $S \Rightarrow A1 \Rightarrow 01$  merupakan sentens dari P.
- $S \Rightarrow 0B \Rightarrow 01$  merupakan sentens dari P.

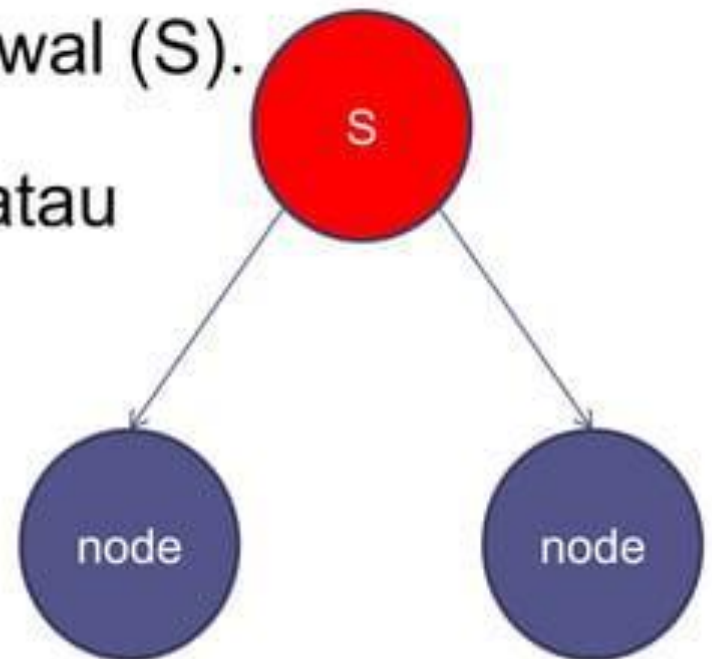
# Pohon Penurunan (*Parse Tree*)

- Turunan dapat juga dinyatakan dalam bentuk *tree*/pohon.
- Sebagai root adalah simbol awal (S).



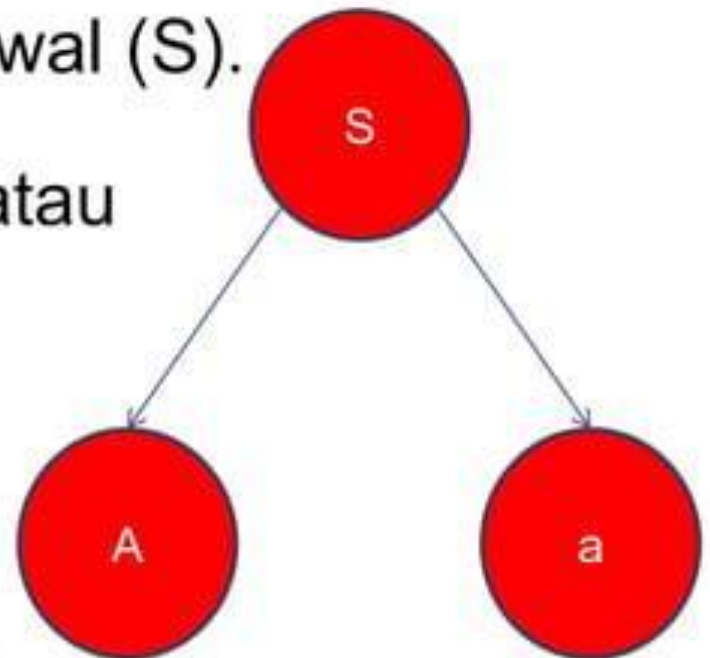
# Pohon Penurunan (*Parse Tree*)

- Turunan dapat juga dinyatakan dalam bentuk *tree*/pohon.
- Sebagai root adalah simbol awal (S).
- Node dapat berupa terminal atau variabel.



# Pohon Penurunan (*Parse Tree*)

- Turunan dapat juga dinyatakan dalam bentuk *tree*/pohon.
- Sebagai root adalah simbol awal (S).
- Node dapat berupa terminal atau variabel.
- Variabel harus diturunkan sampai membentuk terminal.



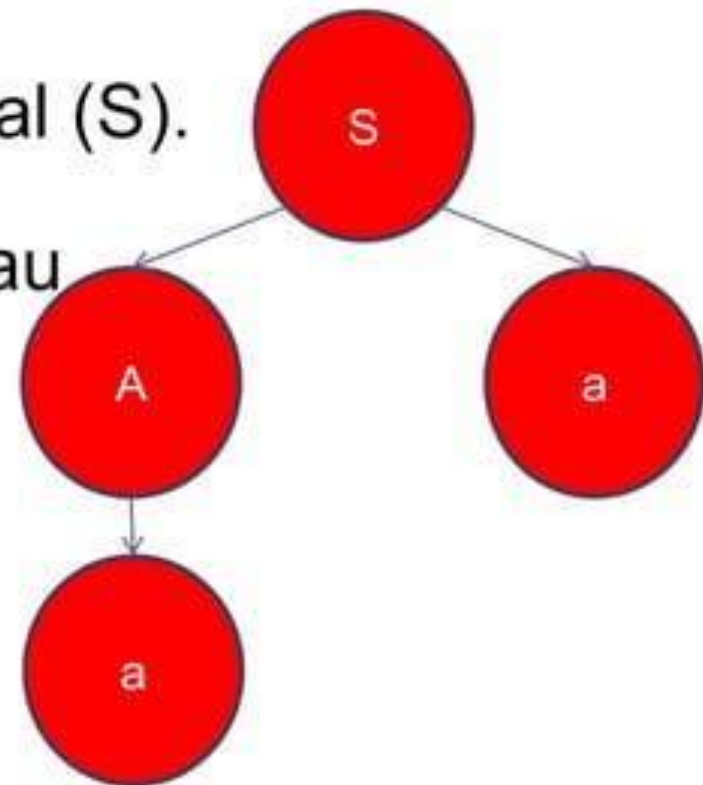
# Pohon Penurunan (*Parse Tree*)

- Turunan dapat juga dinyatakan dalam bentuk *tree*/pohon.

- Sebagai root adalah simbol awal (S).

- Node dapat berupa terminal atau variabel.

- Variabel harus diturunkan sampai membentuk terminal.



## Contoh:

$G = (\{S, A\}, \{a, b\}, P, S)$

P adalah:

$S \rightarrow aAS$

$S \rightarrow a$

$A \rightarrow SbA$

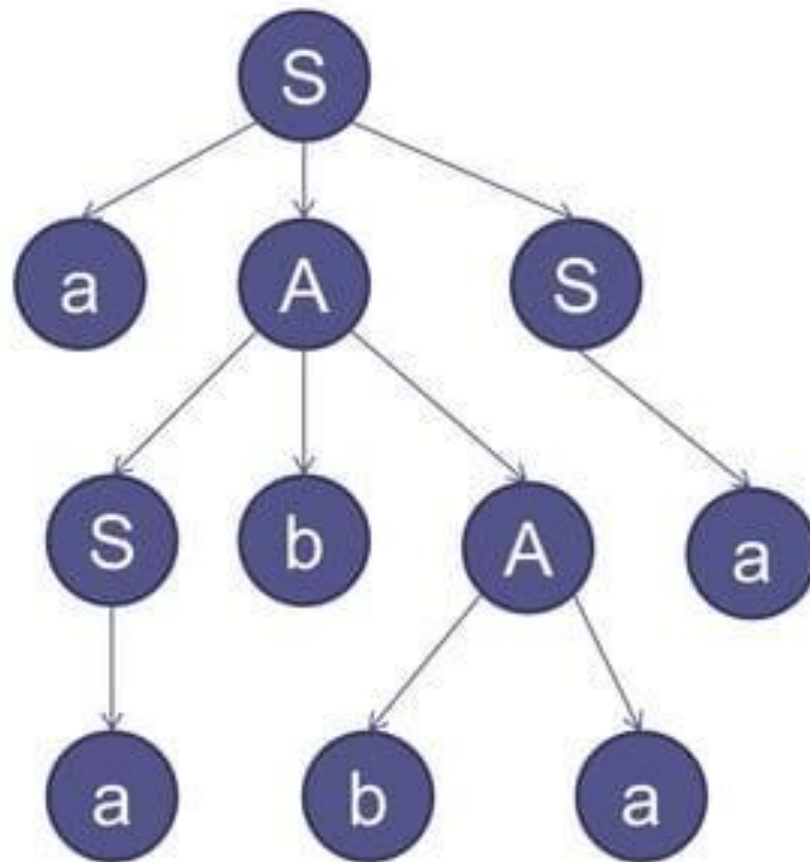
$A \rightarrow SS$

$A \rightarrow ba$

Salah satu turunannya adalah **aabbaa**, buktikan dengan *parse tree*!

# Contoh: Penyelesaian

String **aabbaa** diperoleh melalui:



Aturan  
Produksi:

$S \rightarrow aAS$

$S \rightarrow a$

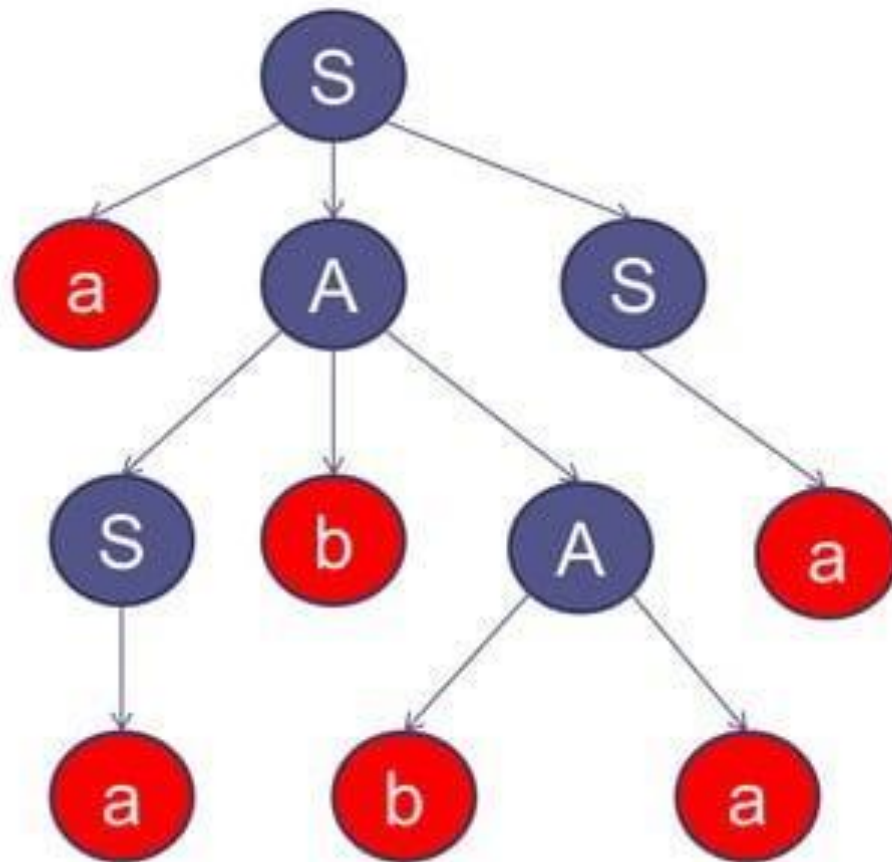
$A \rightarrow SbA$

$A \rightarrow SS$

$A \rightarrow ba$

# Contoh: Penyelesaian

String **aabbaa** diperoleh melalui:



Aturan  
Produksi:  
 $S \rightarrow aAS$   
 $S \rightarrow a$   
 $A \rightarrow SbA$   
 $A \rightarrow SS$   
 $A \rightarrow ba$

# Latihan No.1

Diketahui suatu CFG,

$$G = (\{S\}, \{a,b\}, P, S)$$

dengan P:

$$S \rightarrow aSb$$

$$S \rightarrow aSbb$$

$$S \rightarrow \varepsilon$$

- Buatlah bentuk sentens dari aabbbb!
- Gambarkan parse tree untuk aabb!

## Latihan No.2

Diketahui aturan produksi suatu CFG,

$$S \rightarrow 0A \mid 1B$$

$$A \rightarrow 0AA \mid 1S \mid 1$$

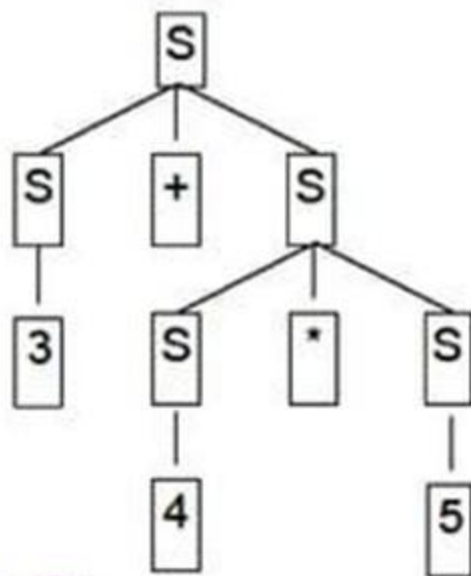
$$B \rightarrow 1BB \mid 0S \mid 0$$

- Tuliskan **quadruple** dari CFG ini!
- Buatlah derivation dari 001101 (LM dan RM)
- Gambarkan parse tree untuk masing-masing derivation tersebut (**LM** dan **RM**)!

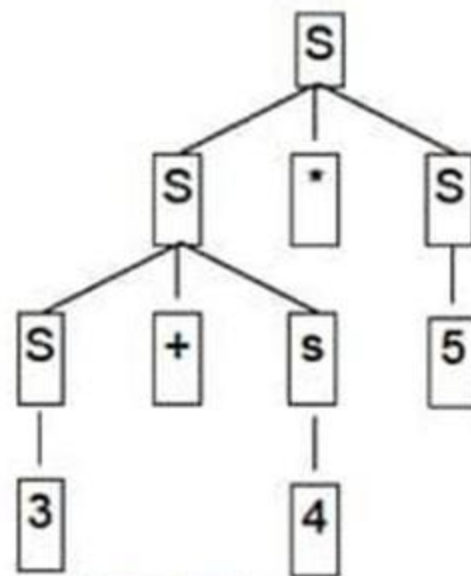
# Contoh Sentence Ambigu

Diketahui Grammar  $\{S \rightarrow S + S \mid S * S \mid \text{angka}, \text{angka} \rightarrow 0 \mid 1 \mid 2 \mid 3 \mid \dots \mid 9\}$

$S \rightarrow S + S$   
 $S \rightarrow S * S$   
 $S \rightarrow \text{angka}$   
Angka  $\rightarrow 0$   
 $\rightarrow 1$   
 $\rightarrow 2$   
 $\rightarrow 3$   
 $\rightarrow \dots$



Atau



23

*kalimat ambigu*

35

## Contoh Sentence Ambigu (Lanjutan)

- Karena hasil keduanya tidak sama maka tidak dapat diimplementasikan pada pelajaran matematika.
- Untuk menyempurnakannya maka diberi **tanda kurung** = UNAMBIGUOUS

$S \rightarrow (S + S) \mid (S * S) \mid \text{angka}$

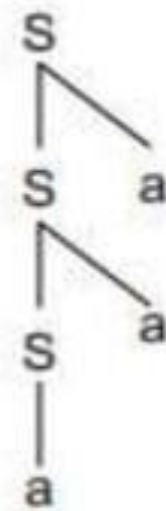
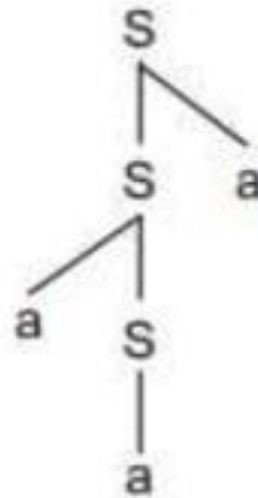
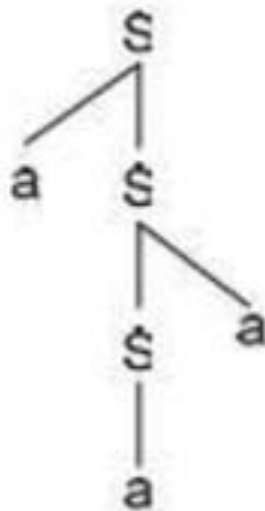
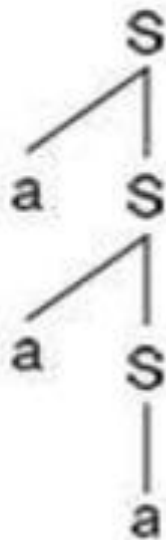
angka  $\rightarrow 0 \mid 1 \mid 2 \mid 3 \dots\dots \mid 9$

# Contoh Ambigu 2

- CFG

$$S \rightarrow aS \mid Sa \mid a$$

bentuk tree dari kata aaa



**Ambiguous**

# Contoh

Perbaiki CFG diatas

$S \rightarrow a S \mid a \rightarrow \text{UNAMBIGUOUS}$

# Contoh Implementasi

Diketahui CFG :

- **S**  $\rightarrow$  \* | + | Angka
- **+**  $\rightarrow$  + + | + \* | + Angka | \* + | \* \* | \* Angka |  
Angka + | Angka \* | Angka Angka
- **\***  $\rightarrow$  \* \* | + \* | + Angka | \* + | \* \* | \* Angka |  
Angka + | Angka \* | Angka Angka
- **Angka**  $\rightarrow$  0 | 1 | 3 | ..... | 9

# Contoh Implementasi

PREORDER

\* + \* + 1 2 + 3 4 5 6

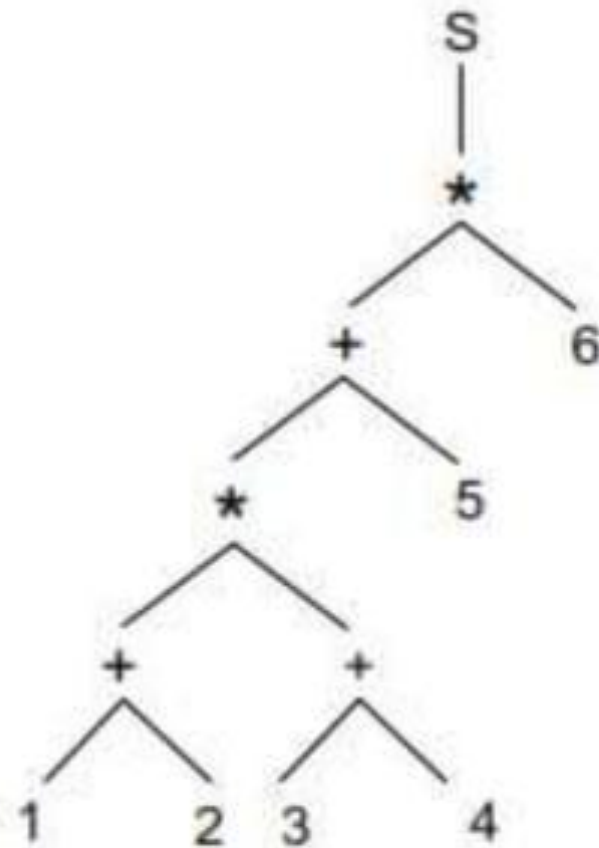
\* + \* + 1 2 7 5 6

\* + \* 3 7 5 6

\* + 21 5 6

\* 26 6

156



# Contoh Implementasi

Diberikan Rule Grammar sebagai berikut

1.  $S \rightarrow x$

2.  $S \rightarrow y$

3.  $S \rightarrow z$

4.  $S \rightarrow S + S$

5.  $S \rightarrow S - S$

6.  $S \rightarrow S * S$

7.  $S \rightarrow S / S$

8.  $S \rightarrow (S)$

Diberikan string  $\rightarrow (x + y) * x - z * y / (x + x)$

# Contoh Implementasi

$$(x + y) * x - z * y / (x + x)$$

S (the start symbol)

→ S - S (by rule 5)

→ S \* S - S (by rule 6, applied to the leftmost S)

→ S \* S - S / S (by rule 7, applied to the rightmost S)

→ ( S ) \* S - S / S (by rule 8, applied to the leftmost S)

→ ( S ) \* S - S / ( S ) (by rule 8, applied to the rightmost S)

→ ( S + S ) \* S - S / ( S ) (etc.)

→ ( S + S ) \* S - S \* S / ( S )

→ ( S + S ) \* S - S \* S / ( S + S )

→ ( x + S ) \* S - S \* S / ( S + S )

→ ( x + y ) \* S - S \* S / ( S + S )

→ ( x + y ) \* x - S \* y / ( S + S )

→ ( x + y ) \* x - S \* y / ( x + S )

→ ( x + y ) \* x - z \* y / ( x + S )

→ ( x + y ) \* x - z \* y / ( x + x )

## Rules

1.  $S \rightarrow x$
2.  $S \rightarrow y$
3.  $S \rightarrow z$
4.  $S \rightarrow S + S$
5.  $S \rightarrow S - S$
6.  $S \rightarrow S * S$
7.  $S \rightarrow S / S$
8.  $S \rightarrow (S)$

# Parsing Tree

