

# Compiler Construction (Spring 2020)

Instructor:

Muhammad Nadeem Khokhar

Master of Science (CIS), UK [1992]

DIMS, DTN, RHOC

Assistant Professor

Tel: 4863353-65, ext. xxx

nadeem@szabist-isb.edu.pk

# BNF

- Backus-Naur Form
  - John Backus
  - Peter Naur
- Formal, Mathematical way to specify CFGs
- Meta Language
  - Language to describe a language that can describe syntax of programming language
- Precise and Unambiguous

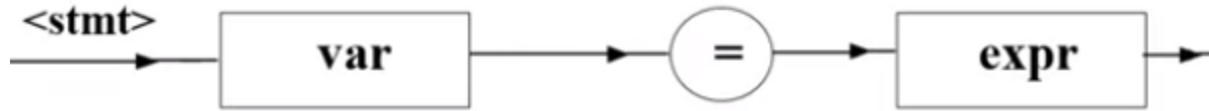
# BNF

- Parts:
  - The start symbol
  - Set of Terminals
  - Set of Non-terminals
  - Set of Productions
- Non-terminals  $\rightarrow$  Terminals/Non-terminals
  - Assignment  $\rightarrow$  id := expr
  - id  $\rightarrow$  A | B | C
  - expr  $\rightarrow$  id+expr | id\*expr | (expr) | id

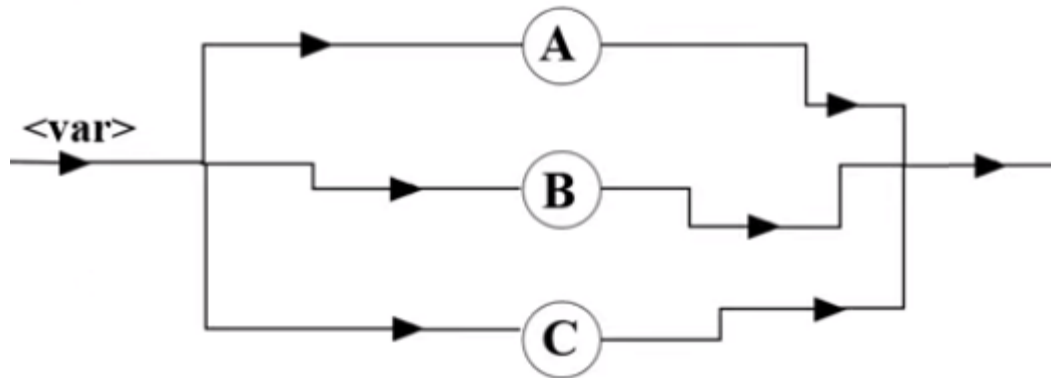
# EBNF

- Extended Backus-Naur Form
- Increases Readability and Writeability
- Special cases:
  - Optional []
  - Repetition {}
    - $\langle \text{stmts} \rangle \rightarrow \langle \text{stmt} \rangle \{ ; \langle \text{stmt} \rangle \}$
    - $\text{digit} \rightarrow \{ \text{digit} \}^+$
  - Group ()
    - $\text{value} \rightarrow +\text{int} \mid -\text{int}$
    - $\text{value} \rightarrow (+ \mid -)\text{int}$

# Syntax Graphs

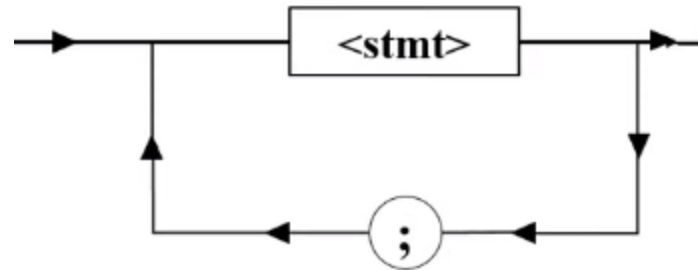


$\langle \text{stmt} \rangle \rightarrow \langle \text{var} \rangle = \langle \text{expr} \rangle$

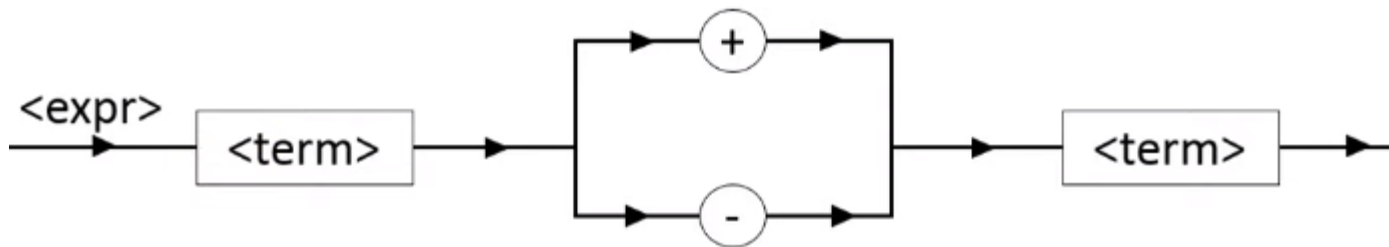


$\langle \text{var} \rangle \rightarrow A \mid B \mid C$

# Syntax Graphs

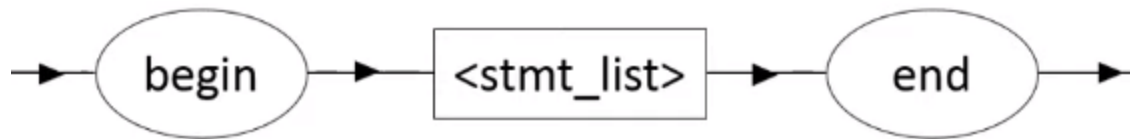


`<stmts> → <stmt> {; <stmt>}`

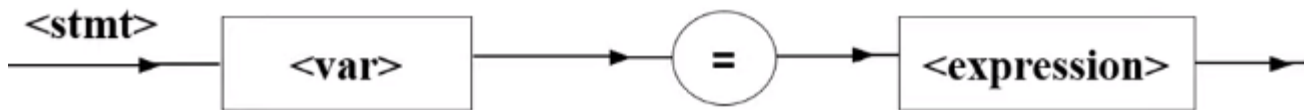


`<expr> → <term> + <term> | <term> - <term>`

# Syntax Graphs



<program>  $\rightarrow$  begin <stmt-list> end



<stmt>  $\rightarrow$  <var> = <expression>