

Chapter 3.4

Data Representation
Data Structures
Data Manipulation

3.4 (k) Backus-Naur Form

- BNF is a syntax definition method.
- Syntax diagrams are also used to define the grammar of a language.
- Notation used commonly is:
 - $\langle \rangle$ entity/object
 - | or
 - ::= defined as
- BNF uses Top Down method.

3.5 (k) Backus-Naur Form

Example: An Integer is defined as a sequence of digits (1, 123, 44343 etc)

Therefore, using BNF, a single digit integer is defined as

$\langle \text{integer} \rangle ::= \langle \text{digit} \rangle$

$\langle \text{digit} \rangle ::= 0|1|2|3|4|5|6|7|8|9$

3.5 (k) Backus-Naur Form

However, integers can be more than a single digit (123, 4444, 2322 etc.)

Example: 123

- 1** is a single digit integer
- 23** but, in case of 23
- 2** is a single digit integer
- 3** followed by another single digit integer

Hence an integer with multiple digits is defined as

<integer> ::= <digit><integer>

A LEVEL COMPUTING

3.5 (k) Backus-Naur Form

expanding indefinitely...

$\langle \text{integer} \rangle ::= \langle \text{digit} \rangle \langle \text{integer} \rangle$

$\langle \text{integer} \rangle ::= \langle \text{digit} \rangle \langle \text{digit} \rangle \langle \text{integer} \rangle$

$\langle \text{integer} \rangle ::= \langle \text{digit} \rangle \langle \text{digit} \rangle \langle \text{digit} \rangle \langle \text{integer} \rangle$

Eventually...

$\langle \text{integer} \rangle ::= \langle \text{digit} \rangle | \langle \text{digit} \rangle \langle \text{integer} \rangle$

Since no sign (+/-) is used, this basically is an unsigned integer.

Hence

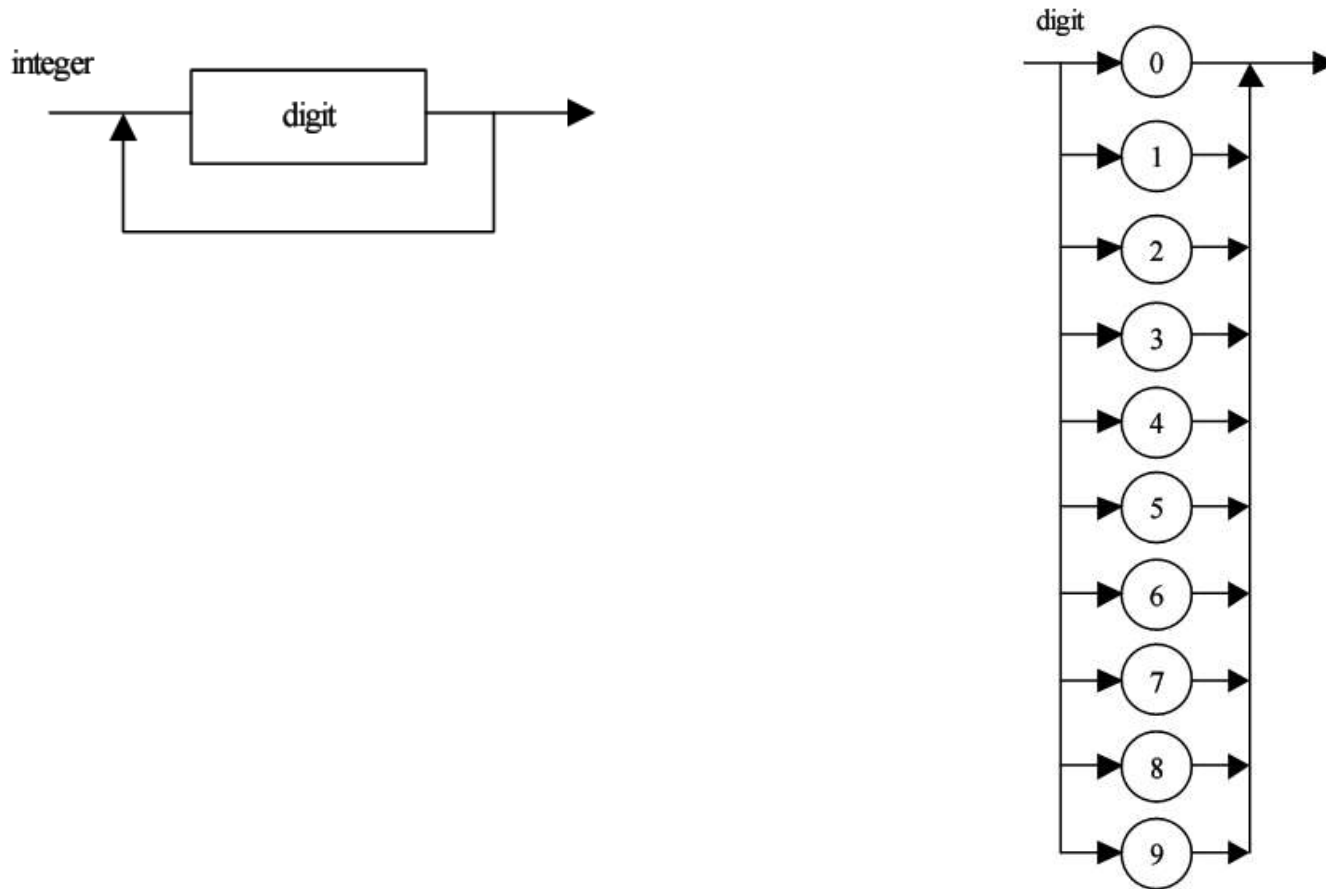
$\langle \text{unsigned integer} \rangle ::= \langle \text{digit} \rangle | \langle \text{digit} \rangle \langle \text{unsigned integer} \rangle$

$\langle \text{digit} \rangle ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9$

A LEVEL COMPUTING

3.5 (k) Backus-Naur Form

Using a syntax diagram, an unsigned integer is shown as



A LEVEL COMPUTING

3.5 (k) Backus-Naur Form

A signed integer is an unsigned integer with a + or – sign in front of it.
 $\langle \text{signed integer} \rangle ::= +\langle \text{unsigned integer} \rangle \mid -\langle \text{unsigned integer} \rangle$

Hence, the complete definition of an integer becomes...

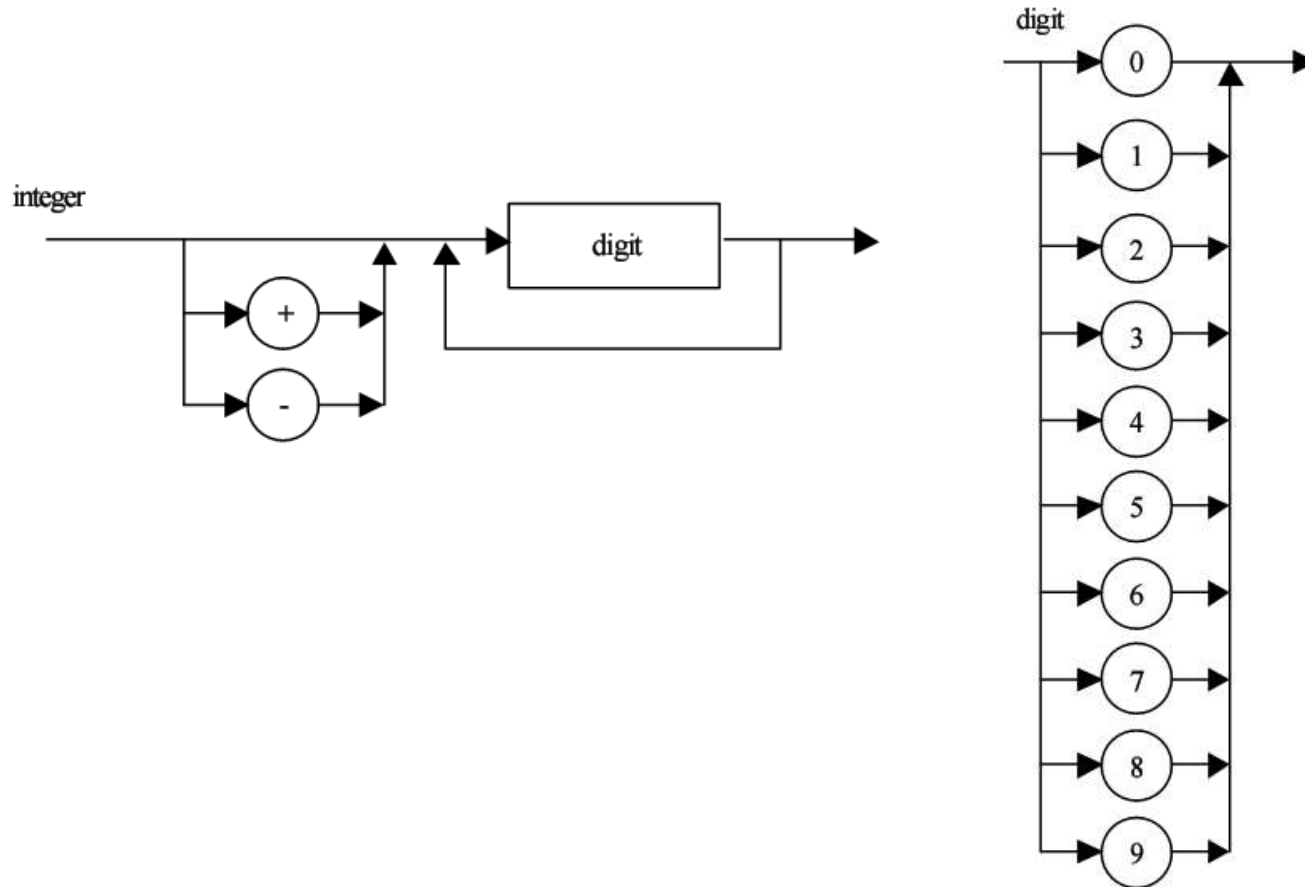
$$\begin{aligned} \langle \text{integer} \rangle &::= \langle \text{unsigned integer} \rangle \mid \langle \text{signed Integer} \rangle \\ \langle \text{signed integer} \rangle &::= +\langle \text{unsigned integer} \rangle \mid -\langle \text{unsigned integer} \rangle \\ \langle \text{unsigned integer} \rangle &::= \langle \text{digit} \rangle \mid \langle \text{digit} \rangle \langle \text{unsigned integer} \rangle \\ \langle \text{digit} \rangle &::= 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9 \end{aligned}$$

We started with top most definition and kept on defining each component until all components are defined, i.e. Top Down approach.

A LEVEL COMPUTING

3.5 (k) Backus-Naur Form

Final syntax diagram



3.2 (b) Lexical Analysis

Keywords:

1. Backus-Naur Form
2. BNF
3. syntax definition
4. syntax diagram
5. Top-Down design