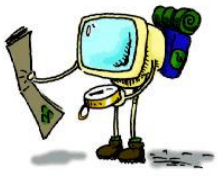
Programming Languages

Tevfik Koşar

Lecture - III
January 24th, 2006

Roadmap

- Creating Language Grammars
  - Regular Expressions
  - Context Free Grammars
  - Parse Trees
  - Ambiguity



Expressing Digits

digit -> 0
digit -> 1
digit -> 2
digit -> 3
digit -> 4
digit -> 5
digit -> 6
digit -> 7
digit -> 8
digit -> 9

digit -> 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

Regular Expression

3

Expressing Numbers

digit -> 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

two-digit-number -> digit digit

number -> digit digit*

* : Kleene star (zero more repetitions of the symbol to its left)

natural_number -> non_zero_digit digit*
non_zero_digit -> 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

Non-terminals Terminals

4

Operators

- op -> + | - | * | /
- expr -> number op number op number

5

Context Free Grammars

- Adds recursion to regular expressions

expr -> number op number op number

expr -> number | expr op expr

6

Example

3 + 4 * 5 ← derive this

expr -> number | expr op expr ← from this

```
expr   -> expr op expr
       -> expr op number
       -> expr * number
       -> expr op expr * num
       -> expr op num * num
       -> expr + num * num
       -> num + num * num
          (3)     (4)     (5)
```

7