Workshop Week 2

Christian Nassif-Haynes

13 August 2013

5 An Algorithm Outputting A Regular Expression In The Rev Language

In the below pseudocode, the language defined by r is over the alphabet Σ.

Algorithm 1 Reverses a regular expression r.

```
1: procedure REVERSE(r)
2:     if r = ϵ then
3:         return r
4:     else if r = ∅ then
5:         return r
6:     else if r ∈ Σ then
7:         return r
8:     else if r = r1r2 then
9:         return REVERSE(r2)REVERSE(r1)
10:    else if r = r1|r2 then
11:        return REVERSE(r1)|REVERSE(r2)
12:    else if r = r1* then
13:        return REVERSE(r1)*
14:    else if r = (r1) then
15:        return (REVERSE(r1))
16:    end if
17: end procedure
```

6 Regular Expressions For Lexical Analysers

- Identifiers that must start with a letter or an underscore and can continue with letters, underscores, or digits: `[a-zA-Z_][a-zA-Z_0-9]*`
- Integers that contain one or more digits: `[0-9]+`
- Real numbers that contain zero or more digits followed by a decimal point followed by one or more digits: `[0-9]*\.[0-9]+`
- Comments that are delimited by curly braces; comments cannot be nested: `\{.+\}`