Converting Regular Expressions into Non-deterministic Automata
NFAs

• NFA: like a DFA, except
  – A transition can lead to more than one state, that is, \( \delta: S \times \Sigma \rightarrow 2^S \)
  – One state is chosen non-deterministically
  – Transitions can be labeled with \( \varepsilon \), meaning states can be reached without reading any input, that is,
    \[ \delta: S \times \Sigma \cup \{ \varepsilon \} \rightarrow 2^S \]
Thompson’s construction
Converts regexps to NFA

Build NFA recursively from regexp tree

Build NFA with left-to-right parse of postfix string using a stack

Input = aab|\cdot c·
- read a, push n1 = nfa(a)
- read a, push n2 = nfa(a)
- read b, push n3 = nfa(b)
- read |, n3=pop(); n2=pop(); push n4 = nfa(or, n2, n3)
- read \cdot, n4 = pop(); n1 = pop(); push n5 = nfa(cat, n1, n4)
- read c, push n6 = nfa(c)
- read \cdot, n6 = pop(); n5 = pop(); push n7 = nfa(cat, n5, n6)
Thompson’s construction

• Converts regexps to NFA
• Six simple rules
  – Empty language
  – Symbols
  – Empty String
  – Alternation \((r_1 \text{ or } r_2)\)
  – Concatenation \((r_1 \text{ followed by } r_2)\)
  – Repetition \((r_1^*)\)

Used by Ken Thompson for pattern-based search in text editor QED (1968)
To keep things simple our version is more verbose
Thompson Rule 0

- For the empty language $\phi$ (optionally include a sinkhole state)
Thompson Rule 1

• For each symbol $x$ of the alphabet, there is a NFA that accepts it (include a sinkhole state)
Thompson Rule 2

• There is an NFA that accepts only $\varepsilon$
Thompson Rule 3

• Given two NFAs for $r_1$, $r_2$, there is a NFA that accepts $r_1 | r_2$
Thompson Rule 3

• Given two NFAs for $r_1$, $r_2$, there is a NFA that accepts $r_1 \cup r_2$
Thompson Rule 4

• Given two NFAs for $r_1$, $r_2$, there is a NFA that accepts $r_1r_2$
Thompson Rule 4

- Given two NFAs for $r_1, r_2$, there is a NFA that accepts $r_1r_2$
Thompson Rule 5

• Given a NFA for $r_1$, there is an NFA that accepts $r_1^*$
Thompson Rule 5

• Given a NFA for $r_1$, there is an NFA that accepts $r_1^*$
Example

- Set of all binary strings that are divisible by four (include 0 in this set)
- Defined by the regexp: \(((0|1)^*00) \mid 0\)
- Apply Thompson’s Rules to create an NFA
Basic Blocks 0 and 1

- 0
- 1

(this version does not report errors: no *sinkholes*)
0|1
$(0|1)^*$
$(0|1)^*00$
\(((0|1)^*00)|0\)