Introduction to Compiler Design

Lesson 5:

Scanner, Regular Expressions to DFA

Scanners

Source Code → **Lexical Analyzer (Scanner)** → Token Stream

Regular Expression → Deterministic Finite State Automata

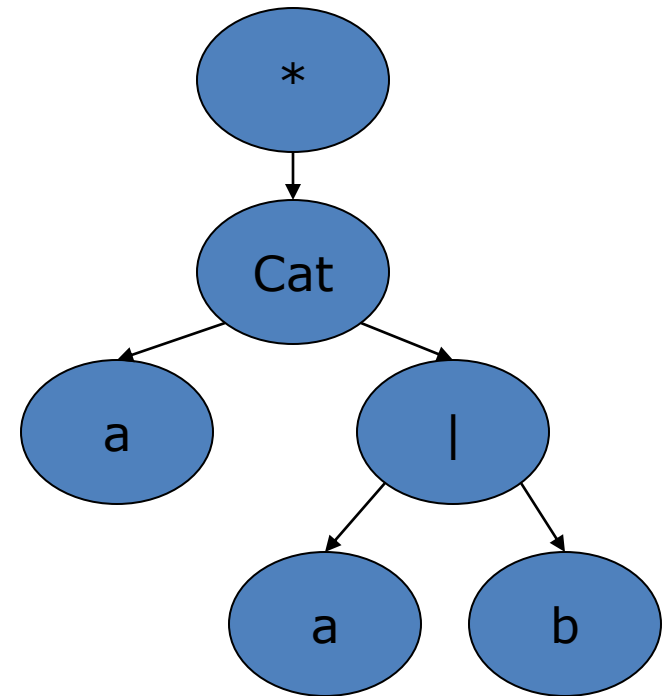Regular Expression → Nondeterministic Finite State Automata → Deterministic Finite State Automata
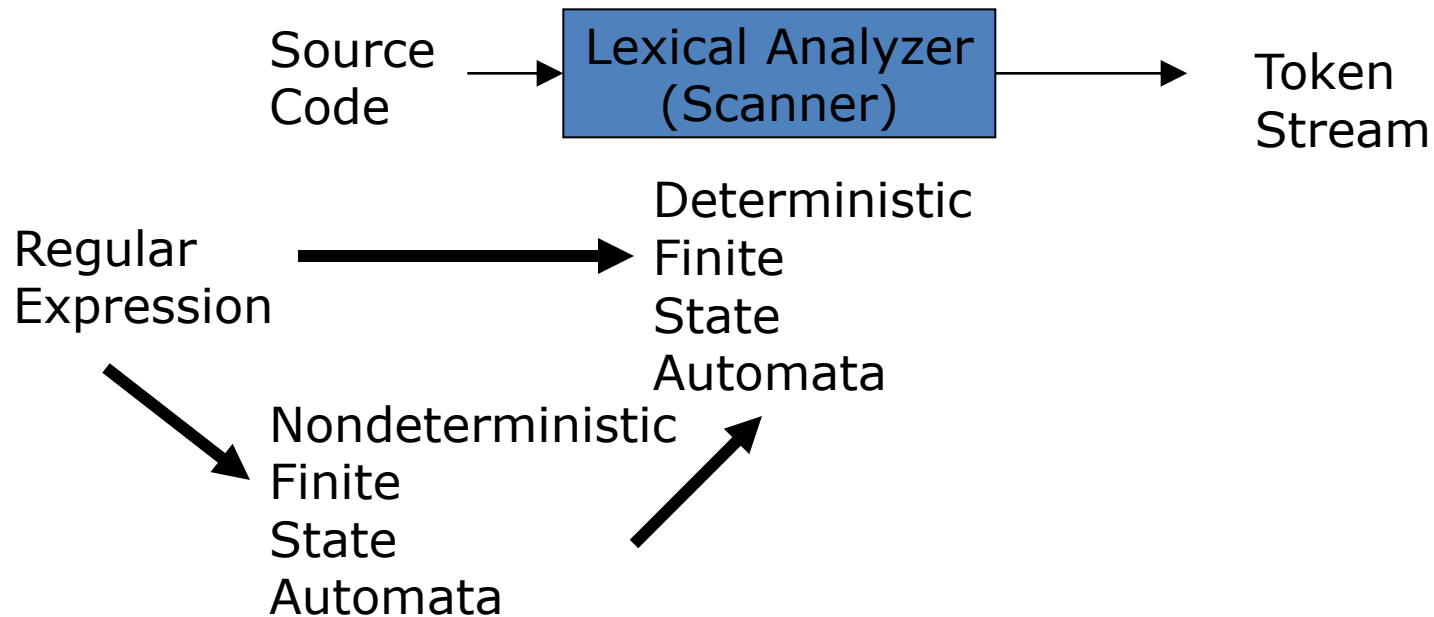
RE to NFA: Step 1

- Create a tree from the Regular Expression
- Example

(a(a|b))*

- Leaf Nodes are either members of Σ or ε
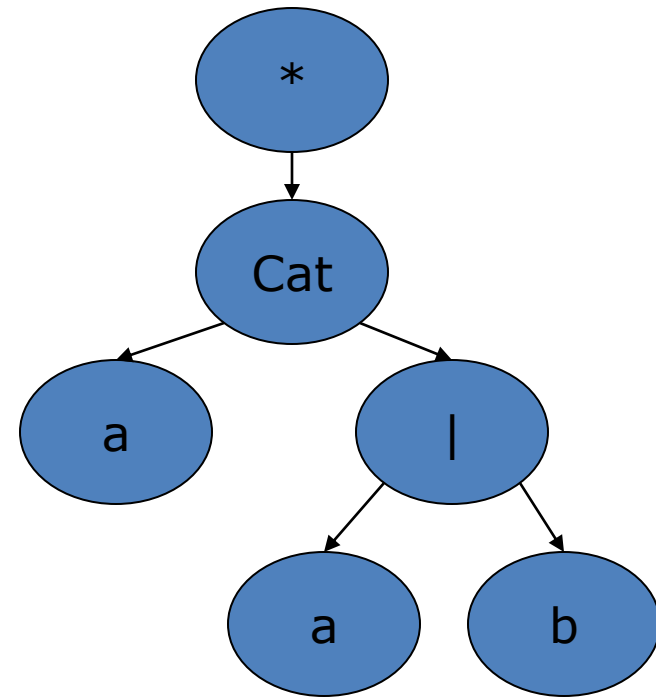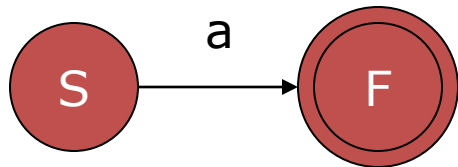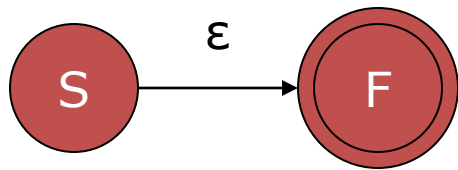- Internal Nodes are operators cat, |, *

RE to NFA: Step 2

- Do a Post-Order Traversal of Tree (children processed before parent)
- At each node follow rules for conversion from a RE to a NFA
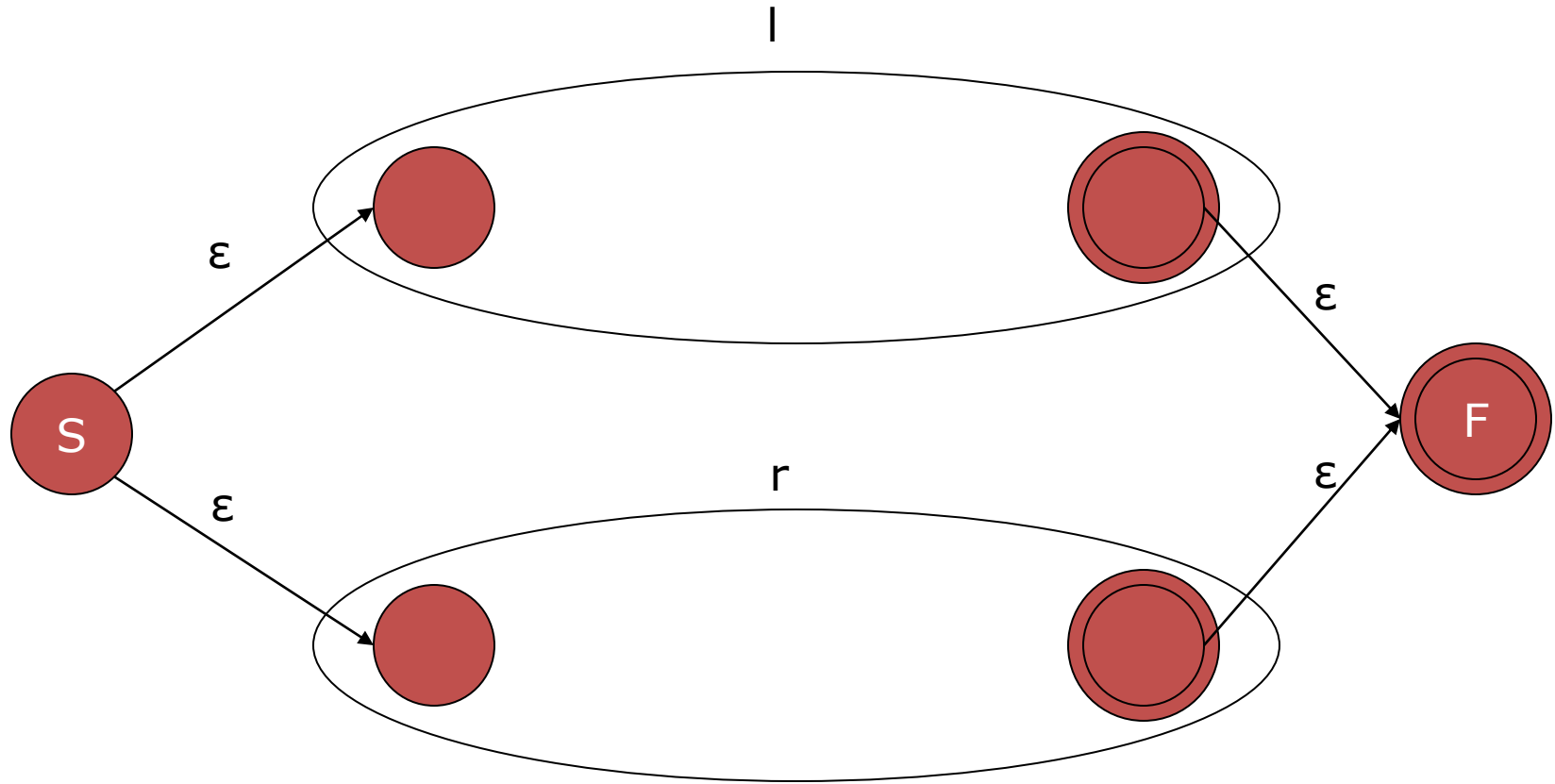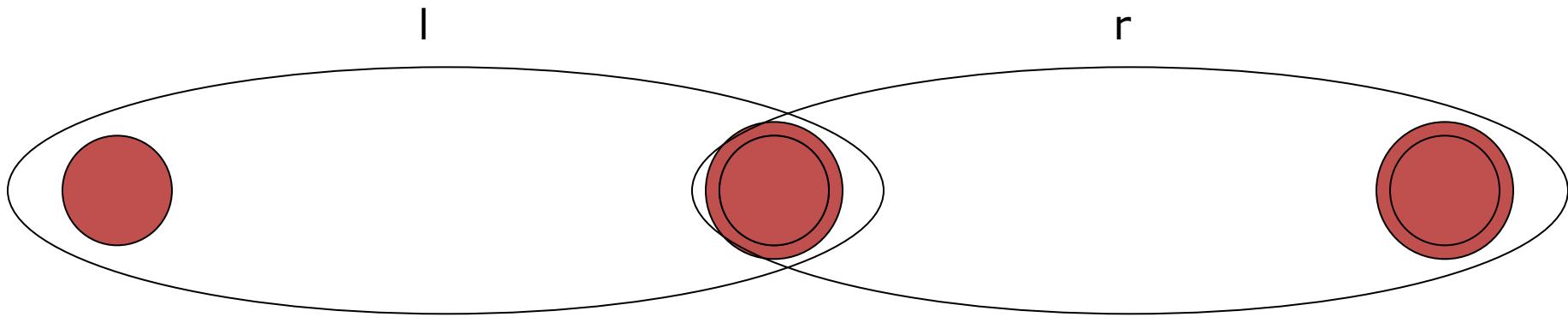
Leaf Nodes

- Either ε or member of Σ

Internal Nodes

- Need to keep track of left (l)and right (r) NFA and merge them into a single NFA
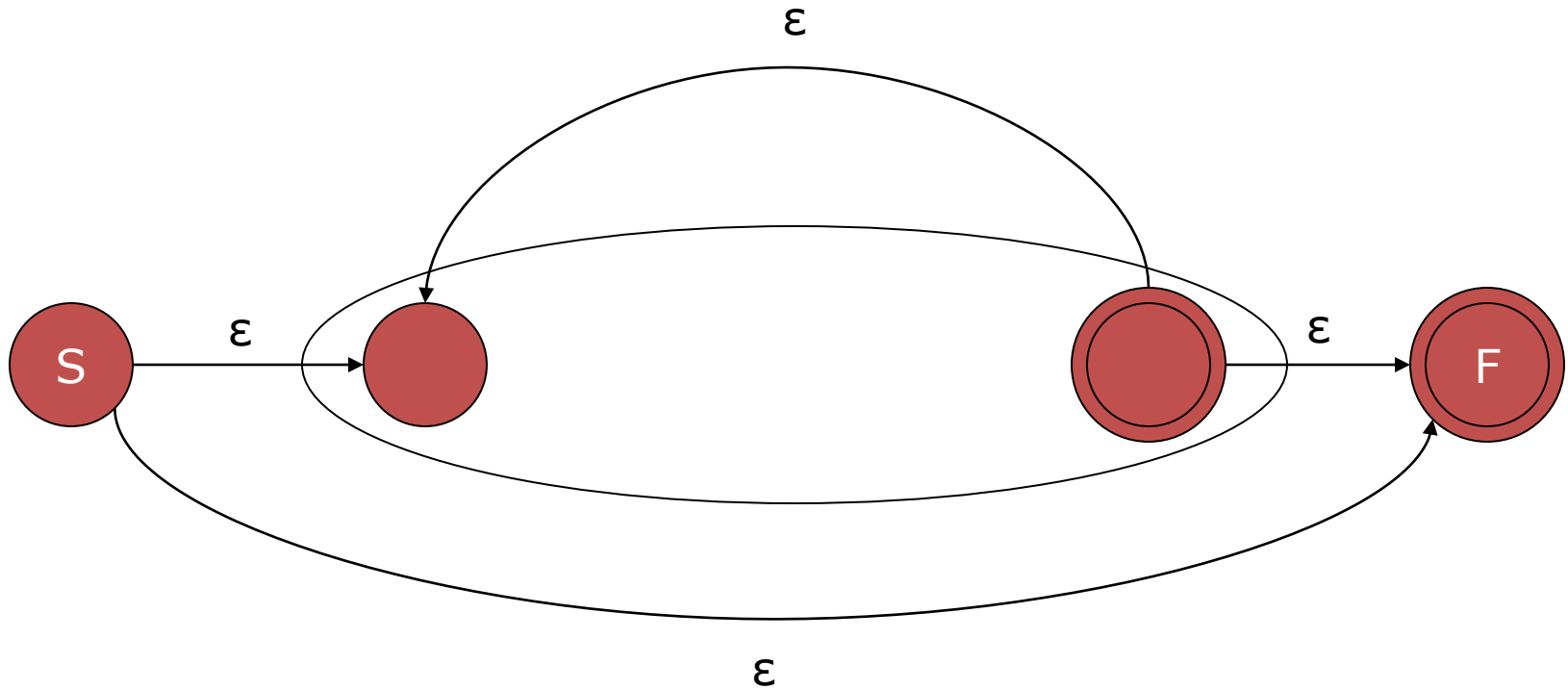 - Or
  - Concatenation
  - Kleene Closure

Or Node

Concatenation Node

Kleene Closure

Example

- Convert the regular expression to a NFA

  (a|b)*abb

  – First convert RE to a tree

  – Then convert tree to NFA

NFA to DFA

- Recall that a DFA can be represented as a transition table

<table>
<tr><td></td><td></td><td colspan="3">Characters</td></tr>
<tr><td></td><td></td><td>+</td><td>-</td><td>Digit</td></tr>
<tr><td rowspan="3">State</td><td>S</td><td>A</td><td>A</td><td>B</td></tr>
<tr><td>A</td><td></td><td></td><td>B</td></tr>
<tr><td>B</td><td></td><td></td><td>B</td></tr>
</table>

Operations on NFA

- ε-closure(t) – Set of NFA states reachable from NFA state t on ε-transitions alone.

- ε-closure(T) – Set of NFA states reachable from some NFA state t in set T on ε-transitions alone.

- move(T,a) – Set of NFA states to which there is a transition on input symbol a from some state t in T

NFA to DFA Algorithm

```
Initially ε-closure(s) is the only state
   in DFA and it is unmarked
While (there is unmarked state T in DFA)
   mark T;
   for (each input symbol a) {
      U = ε-closure(move(T,a));
      if (U not in DFA)
            add U unmarked to DFA
      transition[T,a]=U;
```
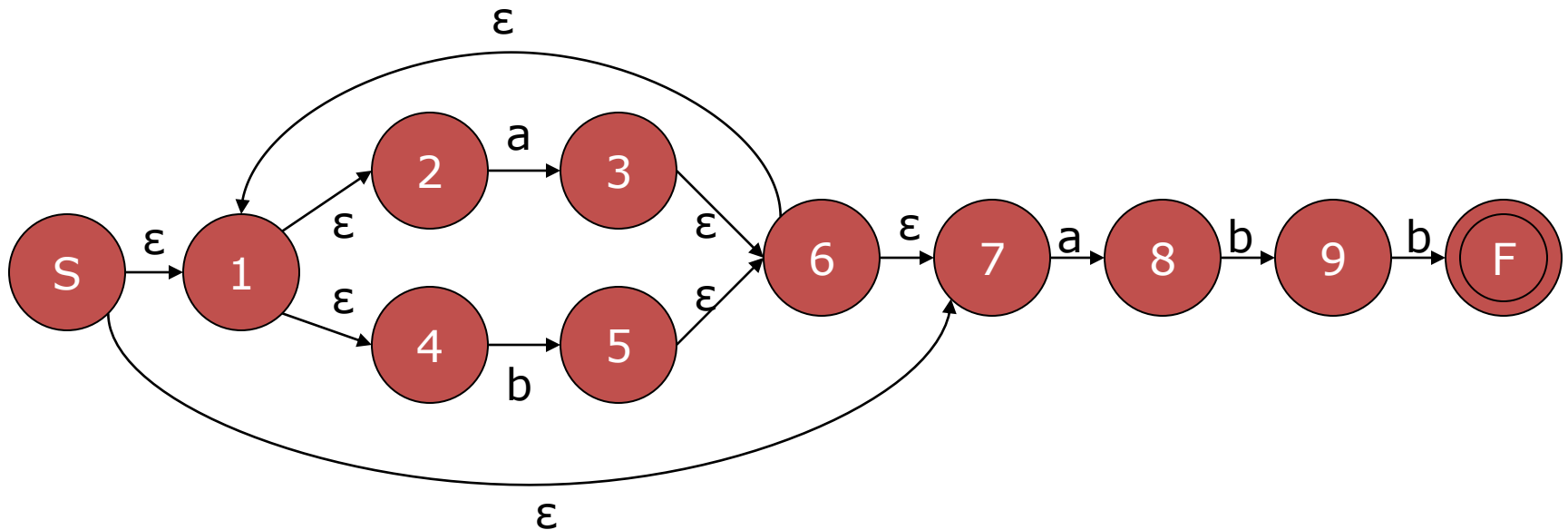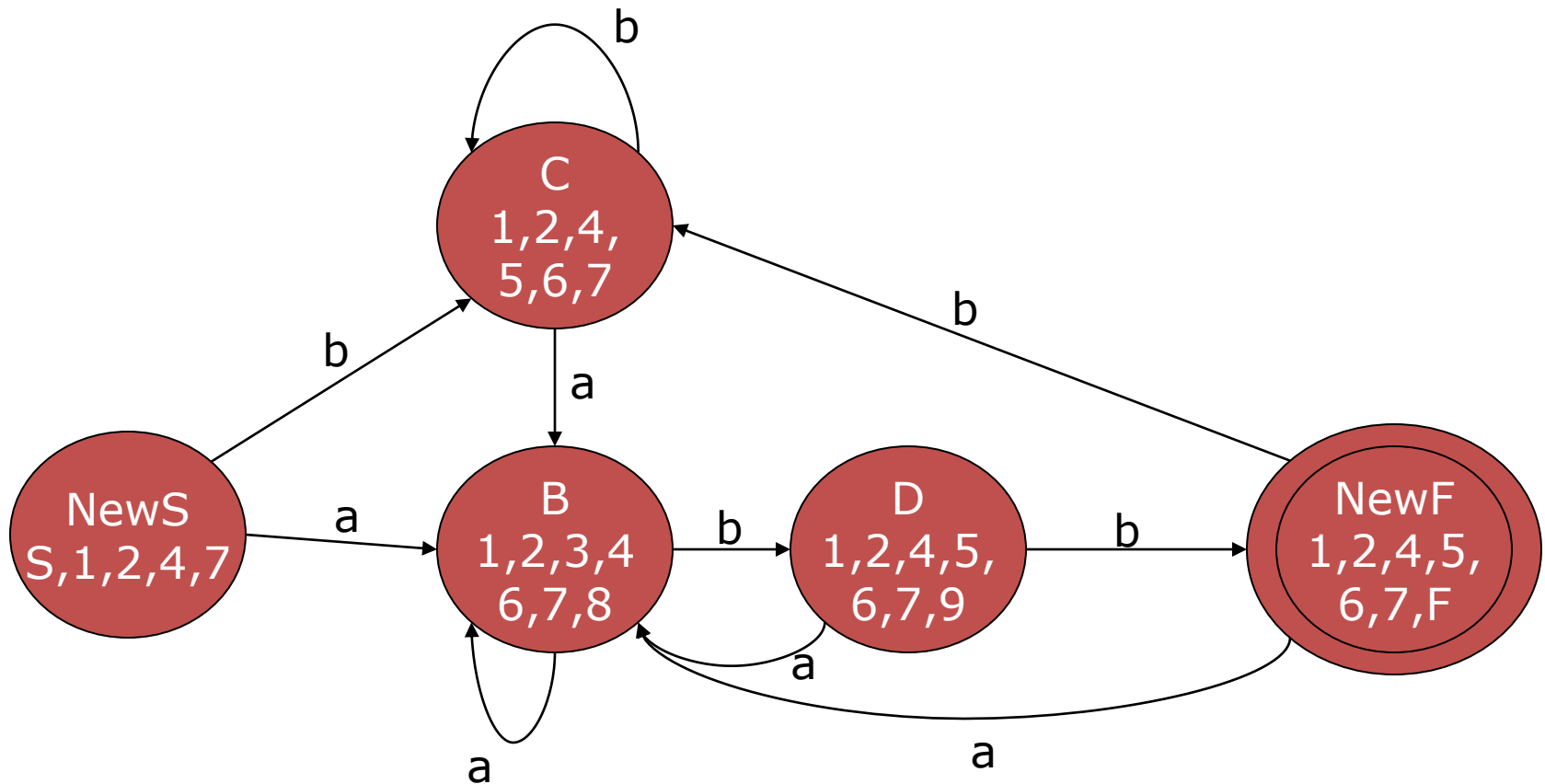
Example

- Take NFA from previous example and construct DFA

Regular Expression: (a|b)*abb

Corresponding DFA

Start State and Accepting States

- The Start State for the DFA is
  ε-closure(s)

- The accepting states in the DFA are those states that contain an accepting state from the NFA

Efficiency of Algorithms

- RE -> NFA

  $O(|r|)$ where $|r|$ is the size of the RE

- NFA -> DFA

  $O(|r|^2 2^{|r|})$ – worst case
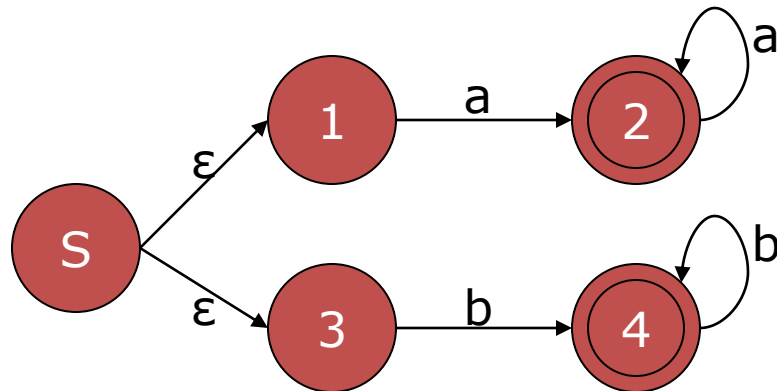 (not seen in typical programming languages)

- Recognition of a string by DFA
  $O(|x|)$ where $|x|$ is length of string x

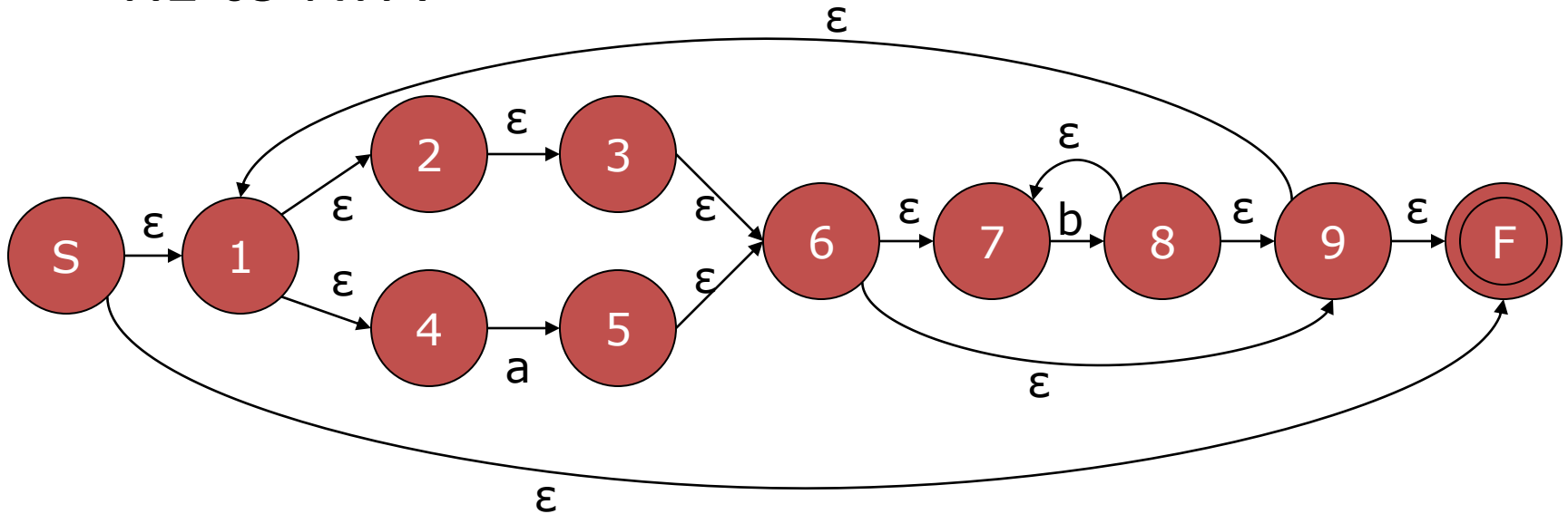17

More Example

- Convert RE to NFA

  ((ε|a)b*)*

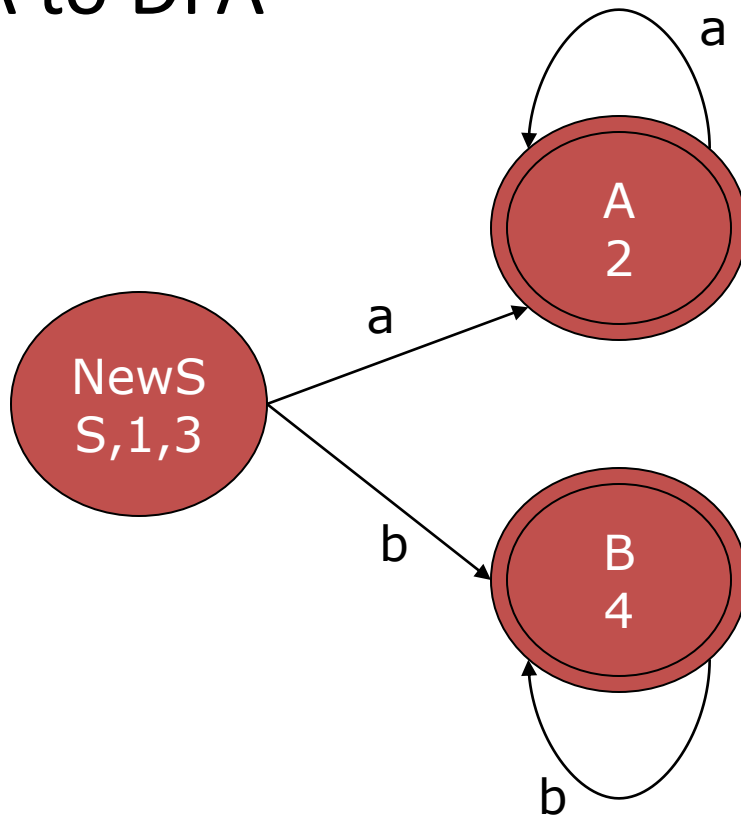
- Convert NFA to DFA

More Example

- RE to NFA

More Example

- NFA to DFA

Summary

- Lexemes
- Tokens
- Regular Expressions, Extended RE
- Regular Definitions
- Finite Automata (DFA & NFA)
- Conversion from RE->NFA->DFA
- JLex