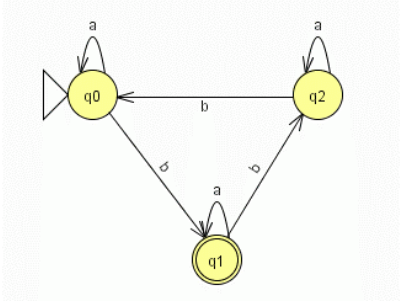
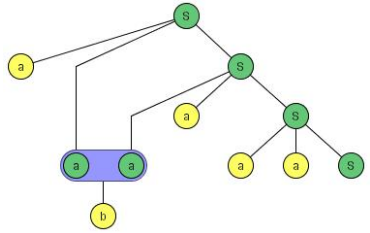
A Hands-on Approach to FLA with JFLAP

Recursively Enumerable Languages

Susan Rodger, Duke University
Thomas Finley, Cornell University
Peter Linz, University of California, Davis

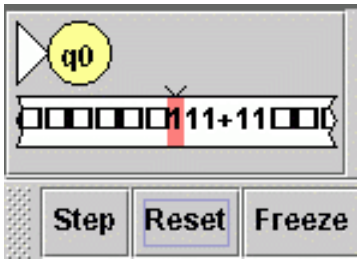




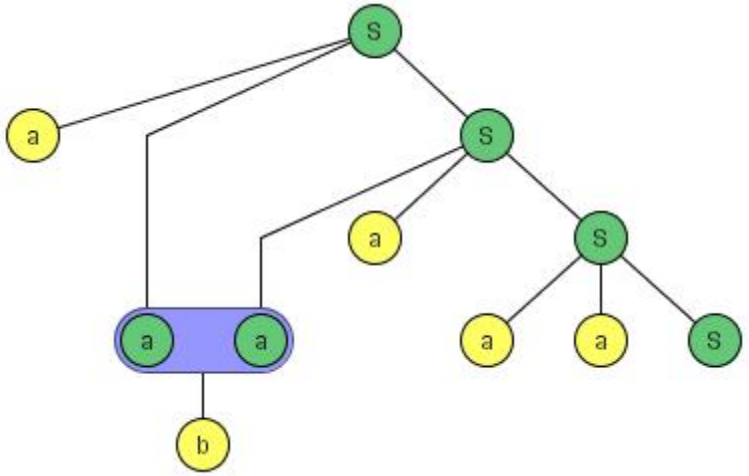
SIGCSE 2006
March 4, 2006

JFLAP – Recursively Enumerable Languages

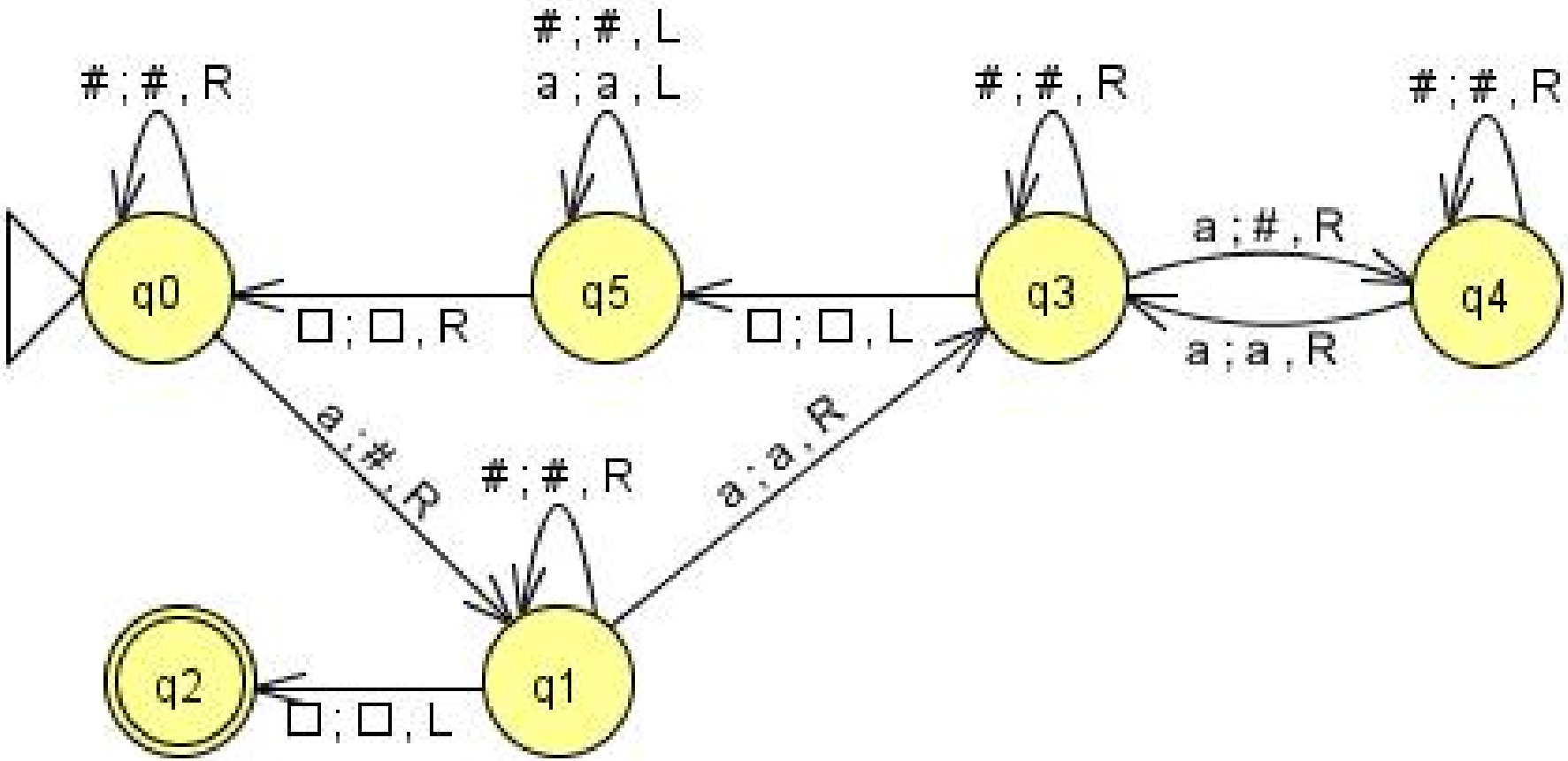
- Create
  - Turing Machine (1-Tape)
  - Turing Machine (multi-tape)
  - Building Blocks
  - Unrestricted grammar



- Parsing
  - Unrestricted grammar with brute force parser

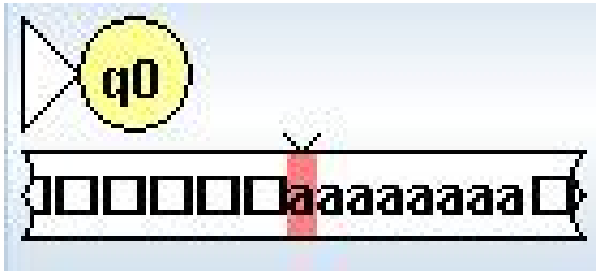


Turing Machine for a^{2^n}

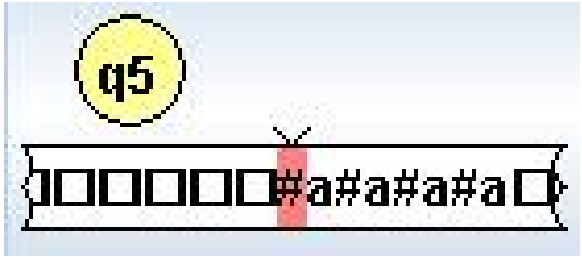


Trace example string aaaaaaaa

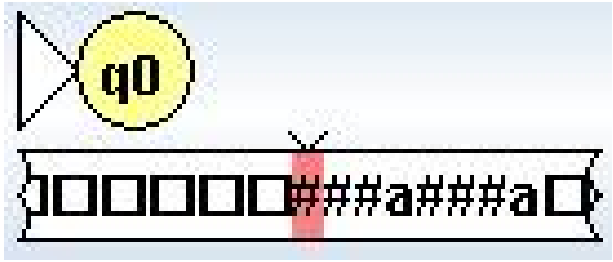
- Start with string



- After several steps

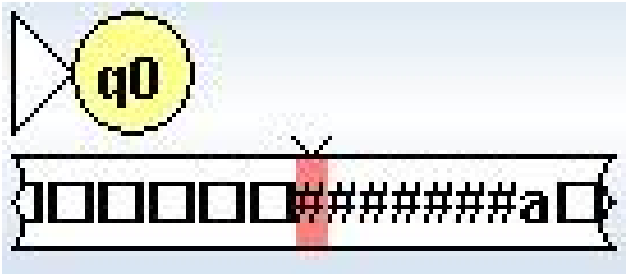


- After several steps

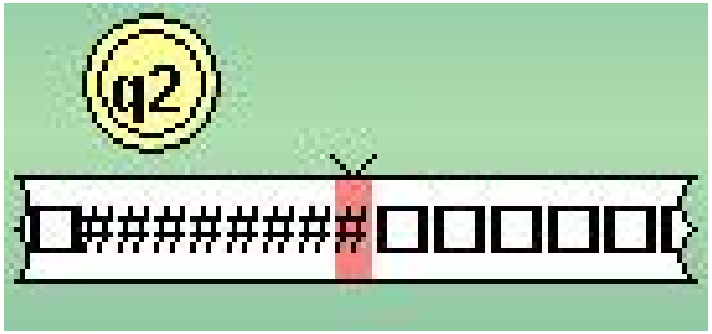


Trace for aaaaaaaa (cont)

- After several steps



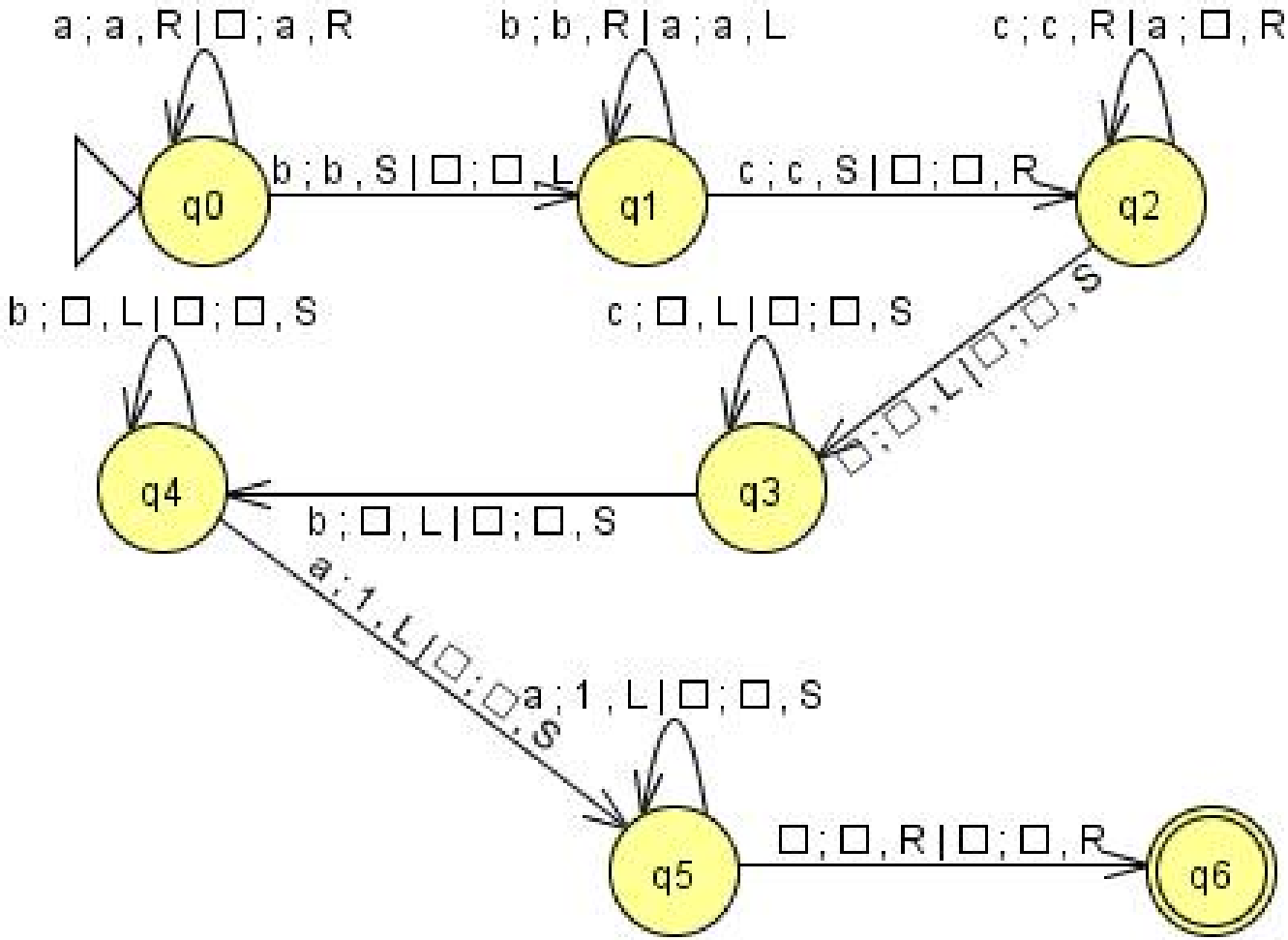
- Accept!



2-Tape Turing Machine for

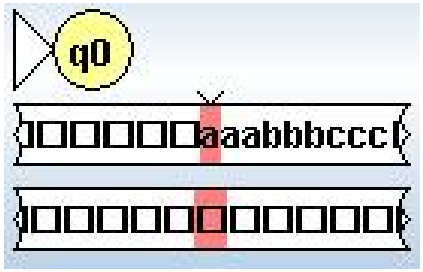
$$f(a^n b^n c^n) = 1^n$$

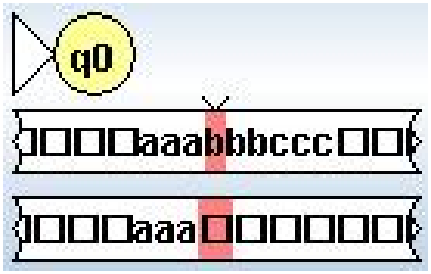
undefined for input not in the correct format

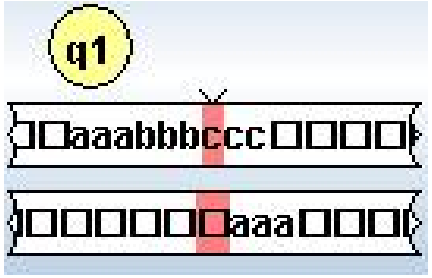


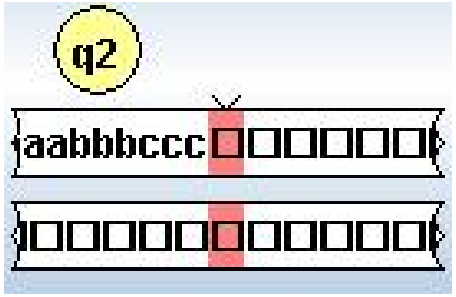
Trace aaabbbccc

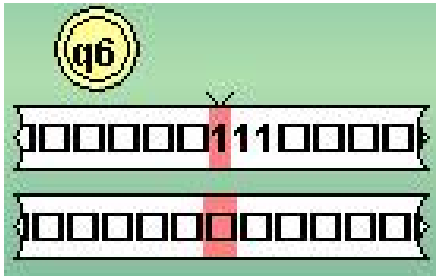
- Start
- For each a, write a on tape 2
- Match a’s/b’s
- Match a’s/c’s, erase a’s
- Erase c’s/b’s, replace a’s with 1’s





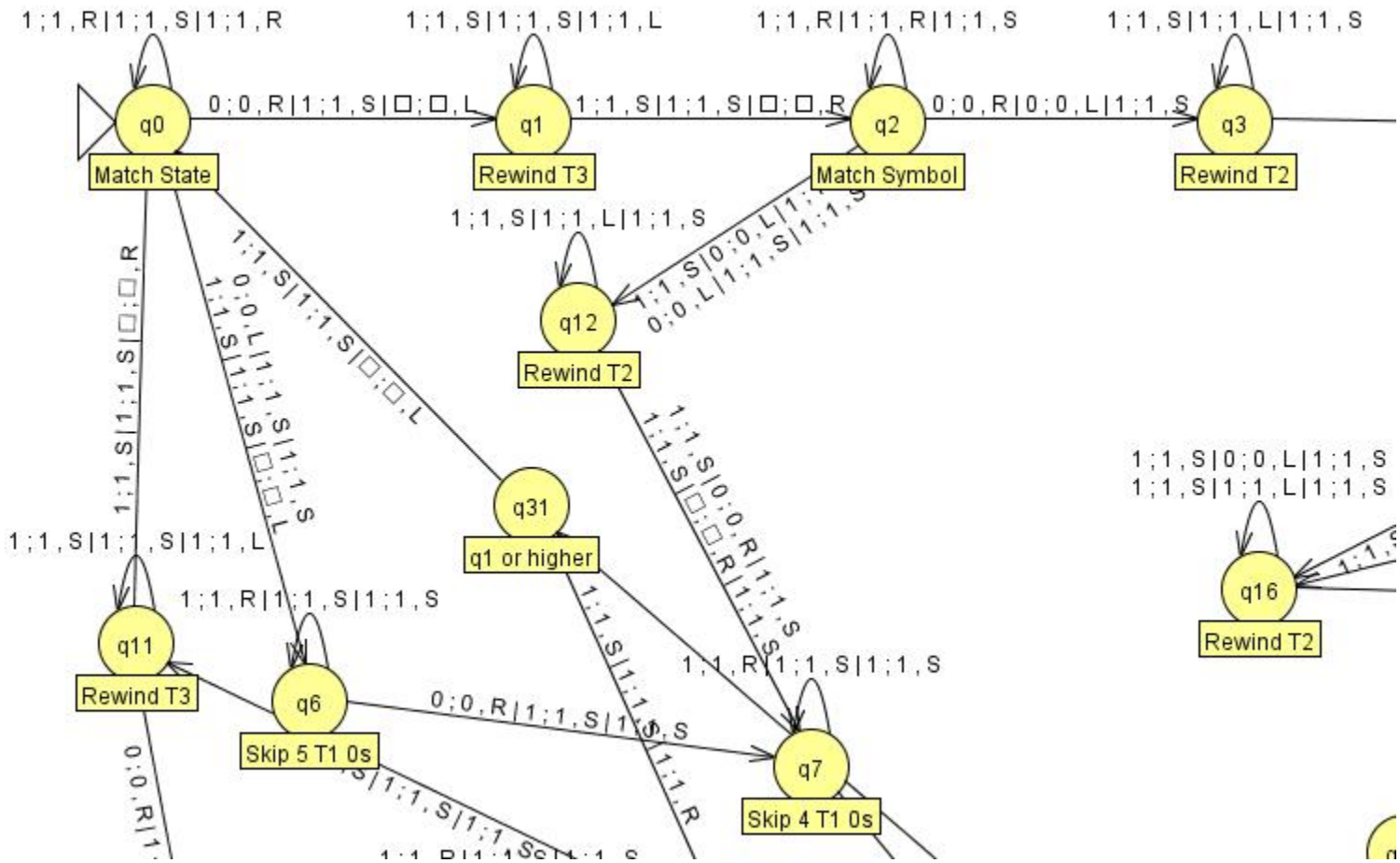






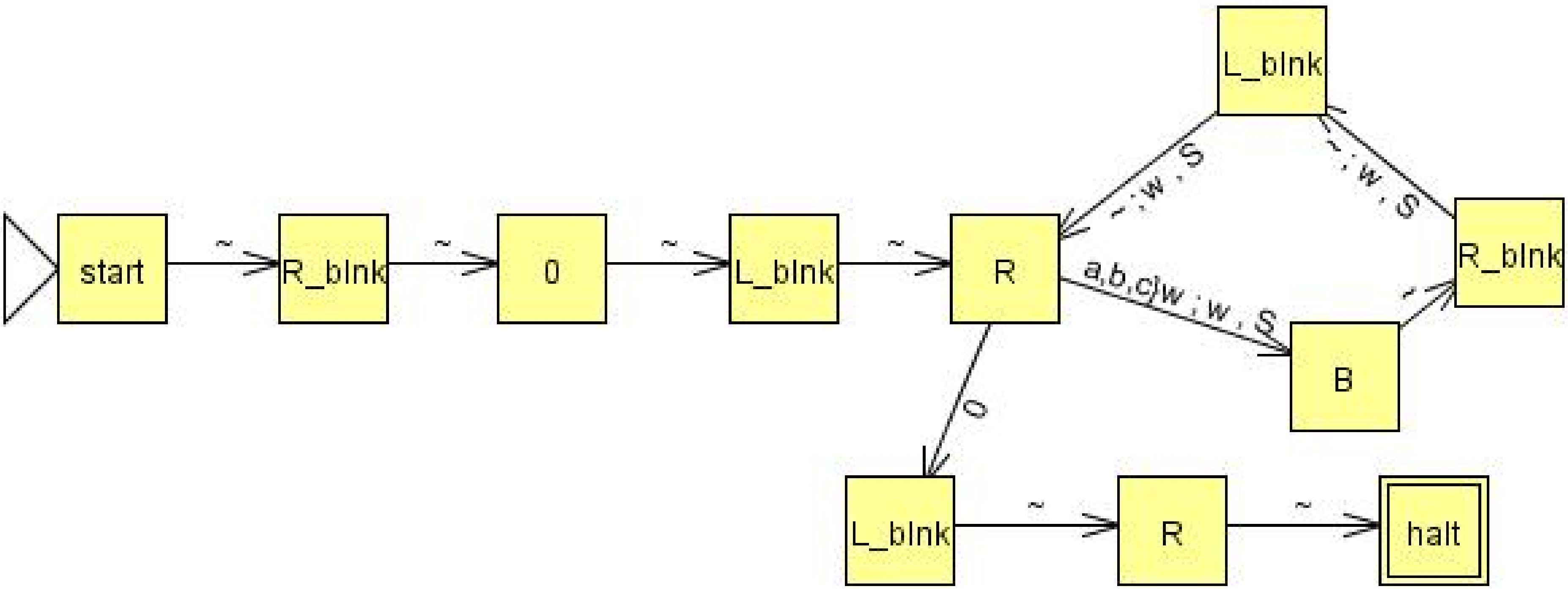
Universal Turing Machine

- 3-Tape TM, 34 states, partly shown below

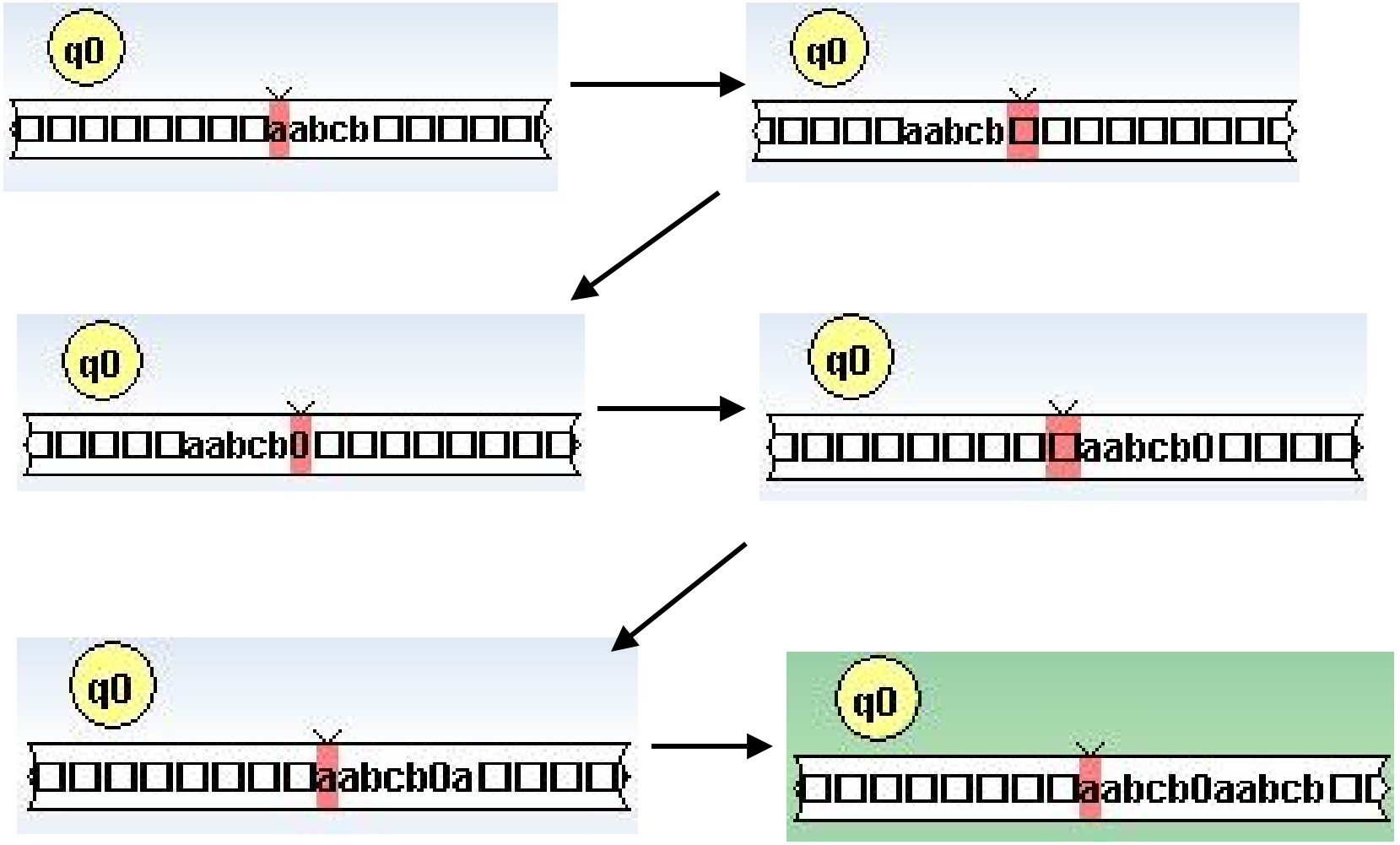


Turing Machine Building Blocks

- SIGCSE 2006 paper
- Copy, f(w) = w0w, *E* = {a,b,c}



Trace of aabcb

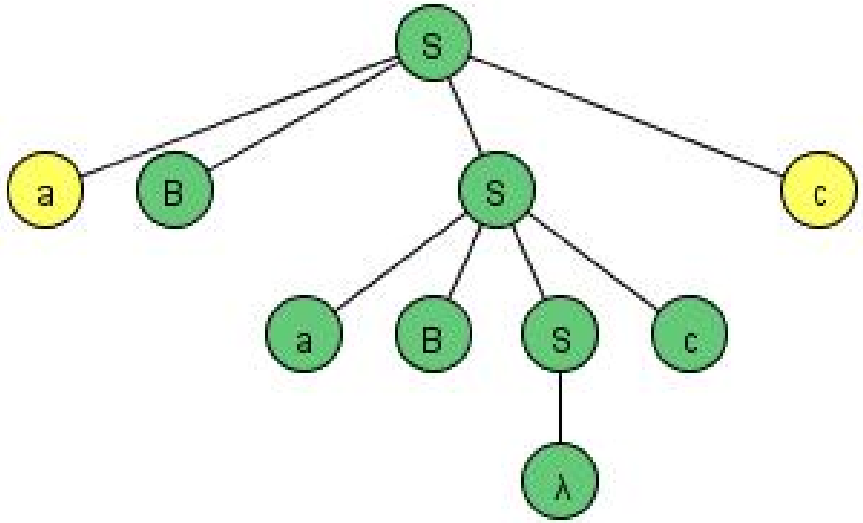


Unrestricted Grammar

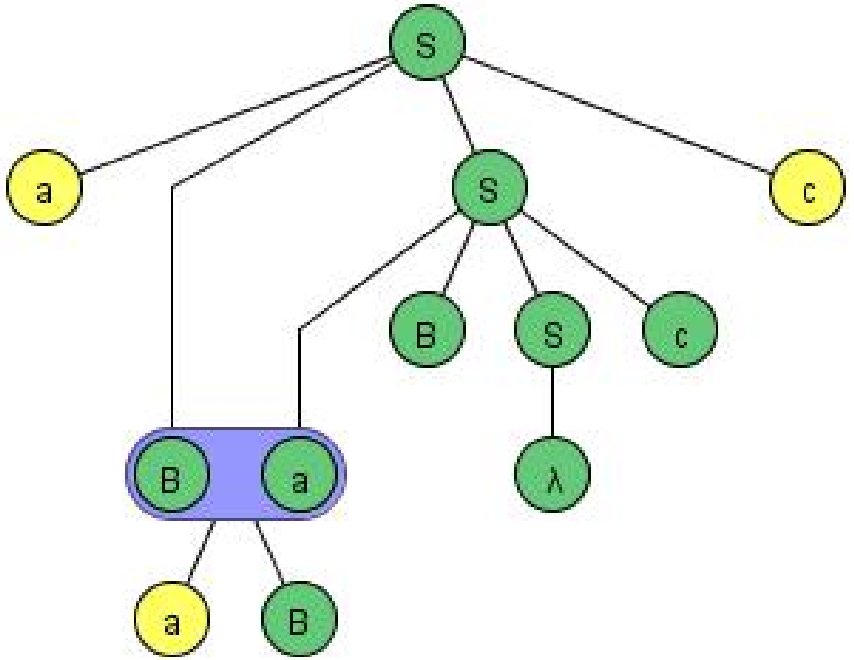
- Multiple symbols on the left side of a rule

S	→	aBSc
S	→	λ
Ba	→	aB
Bc	→	bc
Bb	→	bB

Start of parse tree

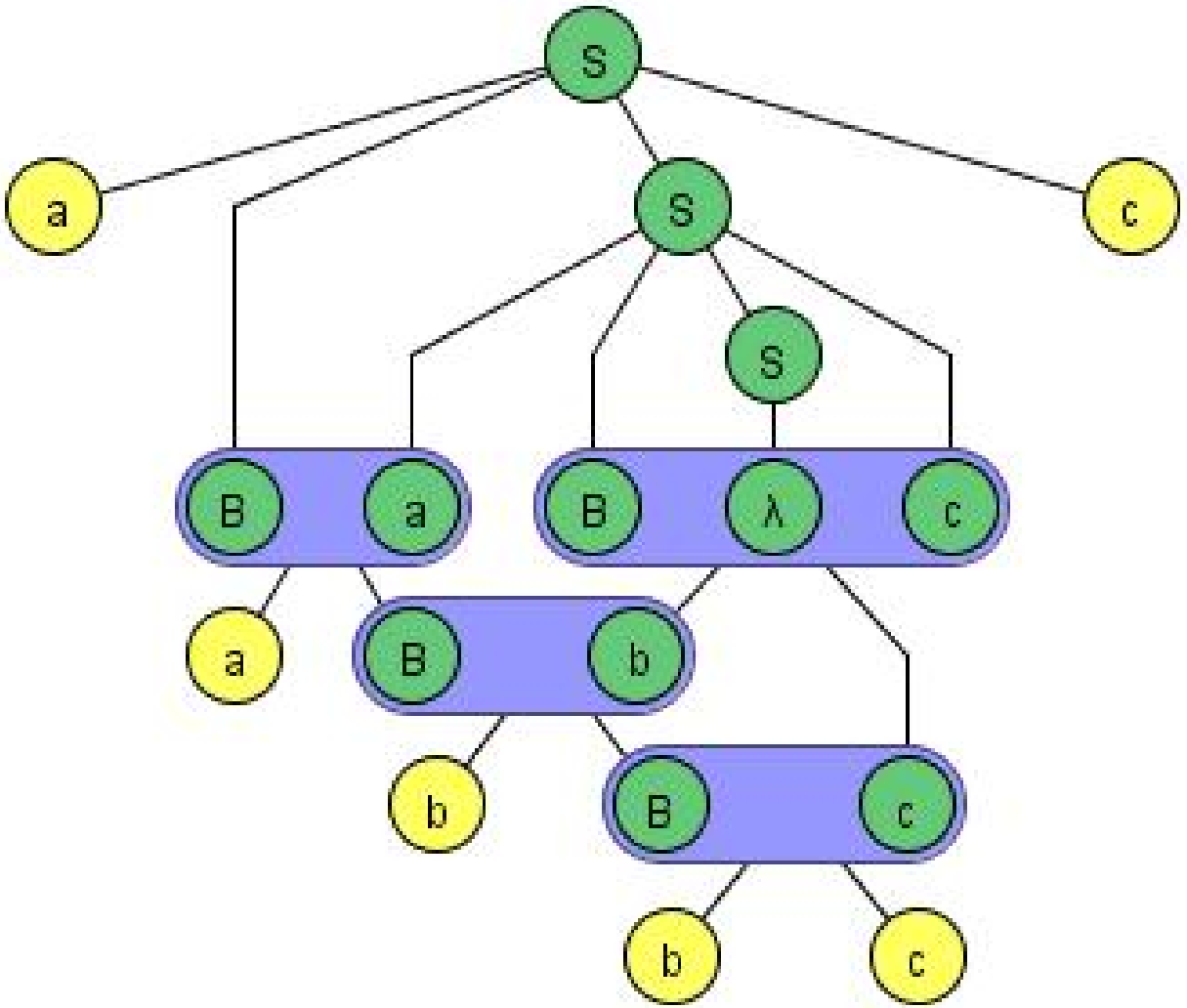


Parse tree one step later



Unrestricted Grammar (cont)

Final tree



Exercises

- Load the TM ex9-a2n and run input
- Write a TM for f(n)=3n, n is unary, E={1}
- Load the 2-Tape TM tm2TAPEanbncn
- Load the TM BB example, duplicateString
- Load the Universal TM ex9-universal
- Enter the unrestricted grammar example