

# CSC 135 Section 5 (Friday)

Kevin Cendana

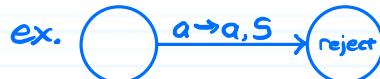
Turing Machine HW

Design a Turing machine for the following languages:

- a.  $A = \{a^m b^n c^n \mid m, n \geq 0\}$
- b.  $B = \{a^n b^n c^m \mid m, n \geq 0\}$
- c.  $C = \{a^i b^j c^k \mid i, j, k \geq 0 \text{ and } i + j = k\}$
- d.  $D = \{a^i b^j c^k \mid i, j, k \geq 0 \text{ and } i + k = j\}$
- e.  $E = \{a^m b^n c^m d^n \mid m, n \geq 1\}$
- f.  $F = \{a^i b^j c^k \mid i, j, k \geq 0 \text{ and } i = j \text{ or } i = k\}$
- g.  $G = \{W \mid \text{all the strings that contains at least one "aaa"}\} \text{ over the alphabet } \{a, b\}. G = \{\text{aaa, babaaaab, ...}\}$

## notes

- if a transition leads to rejected state,  
don't change current symbol & stay in place



- if the current symbol isn't written in any of  
the transitions of the current state, it's rejected or impossible  
(but I tried to write all the rejected transitions anyway)

$$a. A = \{a^m b^n c^n \mid m, n \geq 0\}$$

$L = \text{any } \# \geq 0 \text{ of } a, \text{ same } \# \geq 0 \text{ of } b \text{ then } c$   
 $L = \{\epsilon, bc, abc, aabc, bbcc, \dots\}$

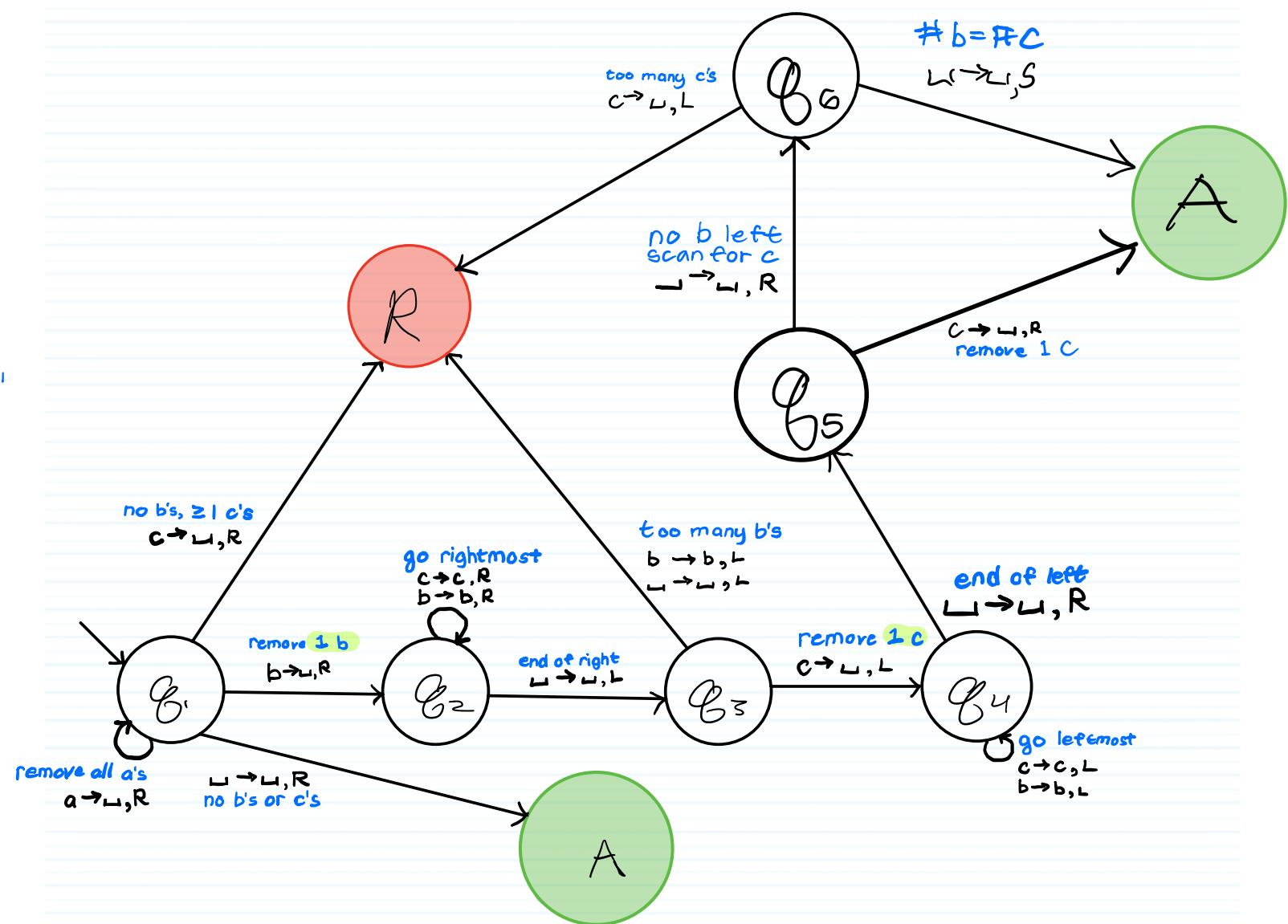
ex) aabbcc      start  
aabbcc  
  bc  
  bc

- ① ignore a, go R til b  
 if no b but c found: reject  
 if no b or c: accept

- ② remove b, go rightmost (go R til  $\sqcup$ , then L one)  
 if no c: reject (too many b's)

- ③ remove c, go leftmost (go L til  $\sqcup$ , R one)  
 if nothing left  $\sqcup$ : go R one  
 if  $\sqcup$ : accept, else reject  
 if c: reject (too many c's)  
 if b: repeat ②

can go to nearest c/b instead  
 but you have to check order at  
 the end or it will accept bcba, bccb, etc



$$b. B = \{a^n b^n c^m \mid m, n \geq 0\}$$

$L = \text{same } \# \text{ of } a \text{ then } b, \text{ any } \# \text{ of } c \geq 0$   
 $L = \{\_, c, cc, abc, aabbcc, \dots\}$

example

$\begin{array}{c} \downarrow \\ \boxed{a \ a \ b \ b \ c \ c} \end{array}$

steps

0) accept if  $\_ | c$

1)  $x \ a \ b \ b \ c \ c$

1) check for 1 a:  
leftmost  $a \rightarrow x$ , go R til b

2)  $x \ a \ x \ b \ c \ x$

2) check for 1 b:

3a)  $x \ x \ x \ b \ c \ c$

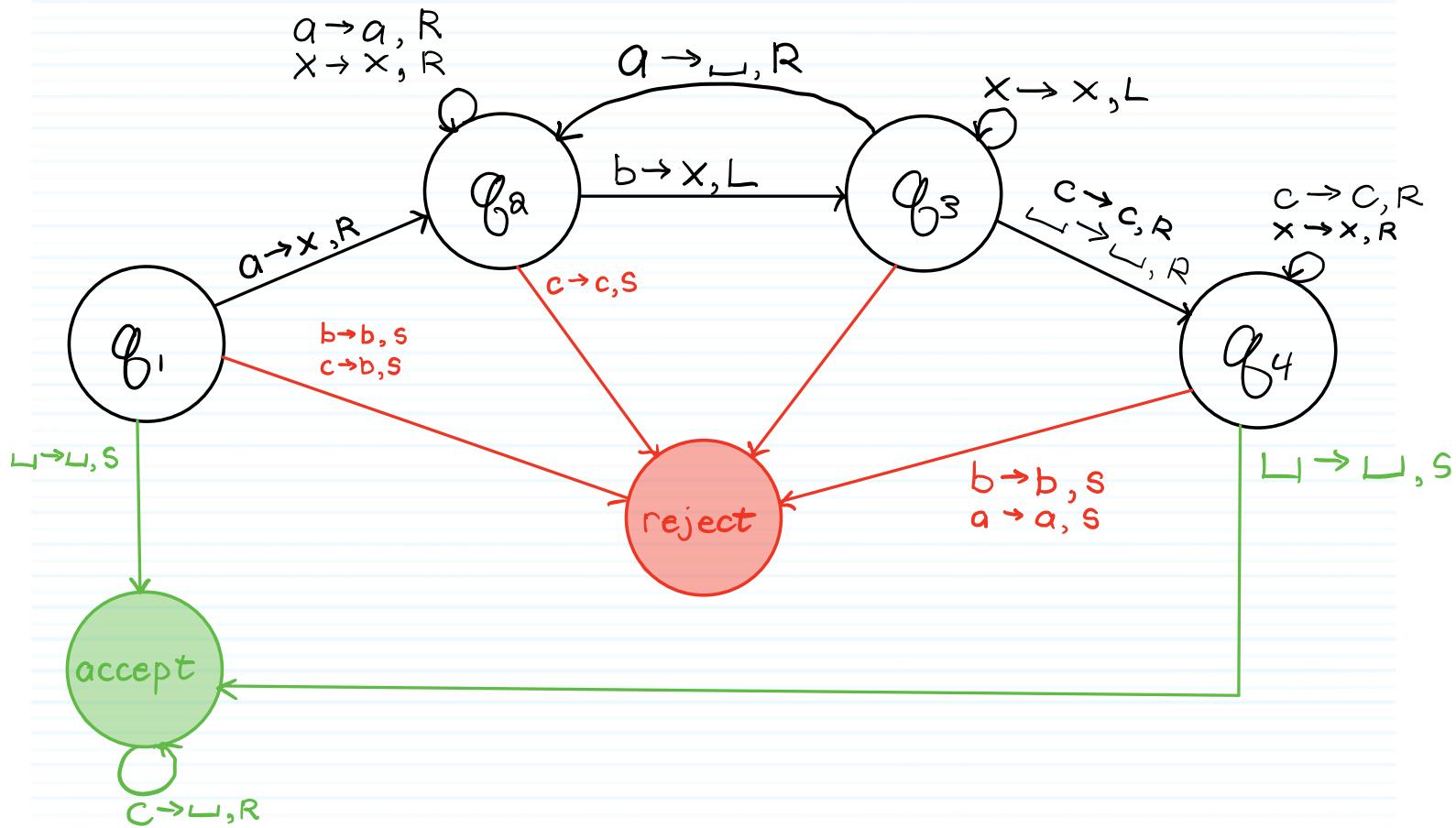
leftmost  $b \rightarrow x$ , go L til  $a | \_$

2)  $x \ x \ x \ x \ x \ c \ c$

3a) if a found: loop back to step 1)

3b)  $x \ x \ x \ x \ x \ c \ c \ \underline{\underline{\underline{\underline{\underline{\_}}}}}$

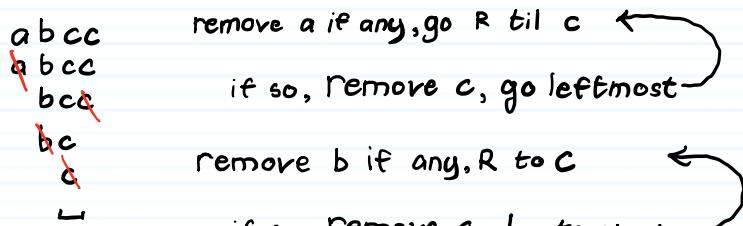
3b) if  $\_$  found: keep going R to ensure no more a/b  
accept



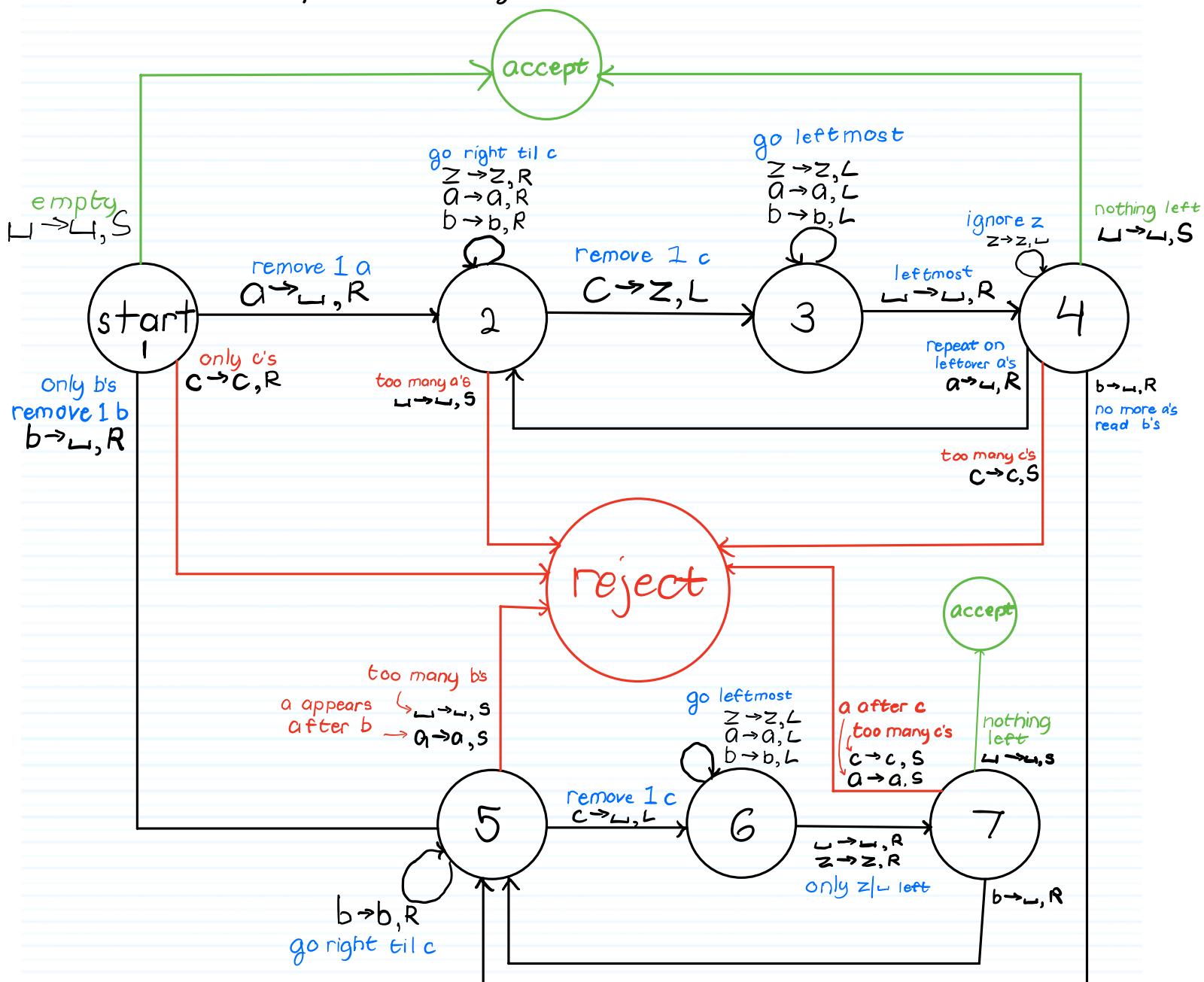
$$c. C = \{a^i b^j c^k \mid i, j, k \geq 0 \text{ and } i + j = k\}$$

# of a's + # of b's = # of c's, all  $\geq 0$

$L = \{\_, ac, bc, abcc, aacc, bbcc \dots\}$



reject if there's no matching c  
for a/b when going L/R



d.  $D = \{a^i b^j c^k \mid i, j, k \geq 0 \text{ and } i + k = j\}$

$$L = a^i b^{i+j} c^j, i, j, k \geq 0$$

$$L = \{\_, abbc, aaabbcc, \dots\}$$

ex)

1 if  $\_$ : accept  
else: start

2 2 replace  $a \rightarrow x$ , go R to b

3 3 replace  $b \rightarrow z$ , go R 1

4 4 replace  $b \rightarrow z$ , go R to c

5 5 replace  $c \rightarrow y$ , go L to  $x$

6 6 nothing, go R 1

summary

go R & change  $a \rightarrow x$ , two  $b \rightarrow z$ ,  $c \rightarrow y$   
go L & repeat if a til no  $a/b/c$

steps 2-5:

if replace value is different, reject

7A  $\rightarrow$  2 7A if current is  $a$ : go to step 2 & repeat

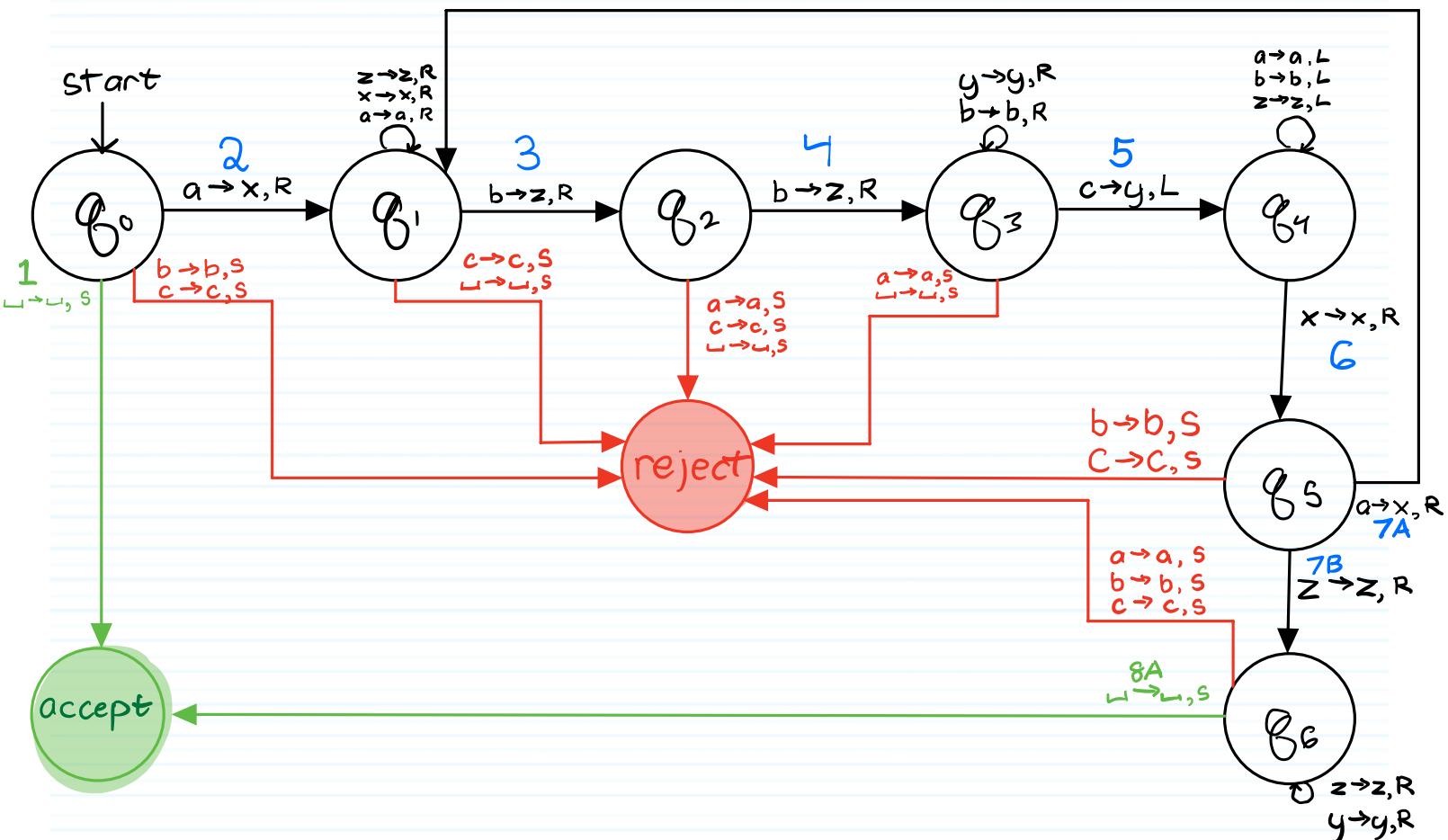
3 7B else if: current is  $z$ : go R to  $\_$ , check every symbol:

4 8A if there are no  $a, b, c$ : accept

5 8B else: reject (a is out of order or too many b or c)

6 7C else: reject

8A

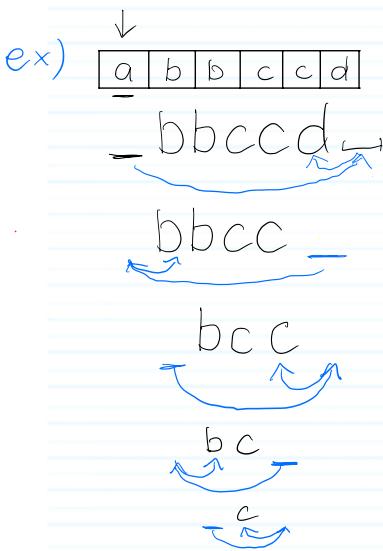


$$e. E = \{a^n b^m c^m d^n \mid m, n \geq 1\}$$

$L = a$ , same # of  $b$  then  $c$ , same # of  $d$  as  $a$   
 $m, n \geq 1$

$$L = \{abcd, aabccdd, abbcccd, aabbccdd \dots\}$$

2 steps to find  
outermost  $a/b/c/d$



read 1 outermost  $a$  then  $d$  til no  $a$  | no  $d$  | neither

→ replace  $a \rightarrow \sqcup$ , go R to  $\sqcup$ , L 1

→ replace  $d \rightarrow \sqcup$ , go L to  $\sqcup$ , R 1

if  $a$ : repeat ①

else if  $w$ : go R 1 to  $b$

read 1 outermost  $b$  &  $c$  til both are gone

if  $b$ : replace  $b \rightarrow \sqcup$ , go R to  $\sqcup$ , go L one

if  $c$ : replace  $c \rightarrow \sqcup$ , go L to  $\sqcup$ , go R one

if  $b$ : repeat ⑤

if symbol isn't found in a transition: reject/impossible

else accept

keep going R  
 $a \rightarrow a, R$   
 $b \rightarrow b, R$   
 $c \rightarrow c, R$   
 $d \rightarrow d, R$

remove 1  $a$ , we  
now need 1  $d$   
 $a \rightarrow \sqcup, R$

not enough  $b$

$\sqcup \rightarrow \sqcup, S$

$c \rightarrow c, S$

$d \rightarrow d, S$

R

go L to rightmost  $d$

$\sqcup \rightarrow \sqcup, L$

$g_3$

process 1  $d$

$d \rightarrow \sqcup, L$

go L til  $\sqcup$

$\sqcup \leftrightarrow \sqcup, L$

repeat to get  
every  $a, d$

$a \rightarrow \sqcup, R$

$\sqcup \leftrightarrow \sqcup, L$

no  $b/c$

$\sqcup \rightarrow \sqcup, S$

R

no more  $a, d$

process  $c$

$c \rightarrow \sqcup, L$

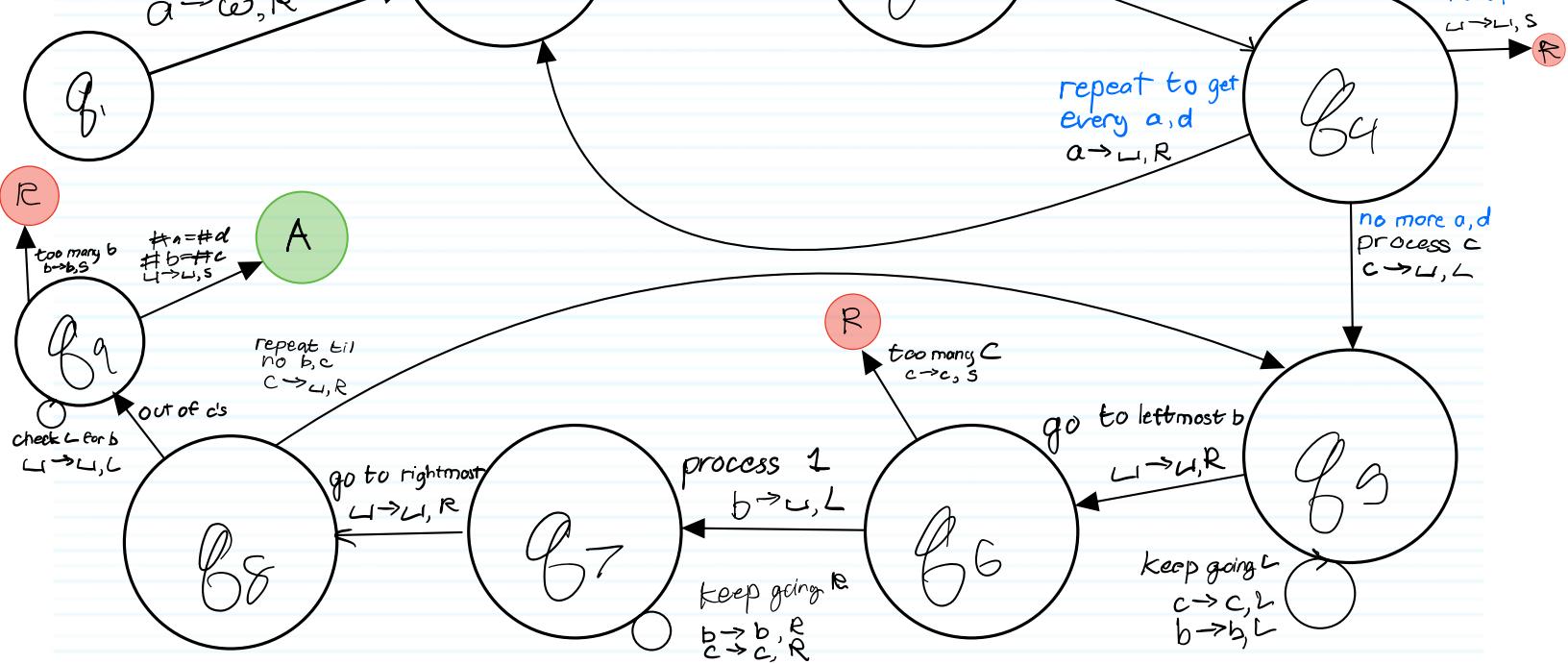
$\sqcup \leftrightarrow \sqcup, L$

no more  $a, d$

process  $c$

$c \rightarrow \sqcup, L$

R



$$f. F = \{a^i b^j c^k \mid i, j, k \geq 0 \text{ and } i = j \text{ or } i = k\}$$

$L = \{ \text{ same # of a's then b's, any # of c's } \} \text{ or } \{ \# \text{ of a, any # of b, same # of a's as c} \} \quad i, j, k \geq 0$

$L = \{ \_ , c, b, abc, aabbc, aabcc \dots \}$

start  $\rightarrow$  edge cases:  
 if  $\_ | c$ : accept  
 else if  $b$ : go R one  
 if  $a$ : accept

if symbol doesn't appear in a transition: reject/impossible  
 ● check for  $\#a = \#b$   
 ○ check for  $\#a = \#c$

ex)  $\underline{a}abcc$

wabcc  
 wabc  
 waycc

WWycc  
 WWycc

Wxycc

Wxyzc

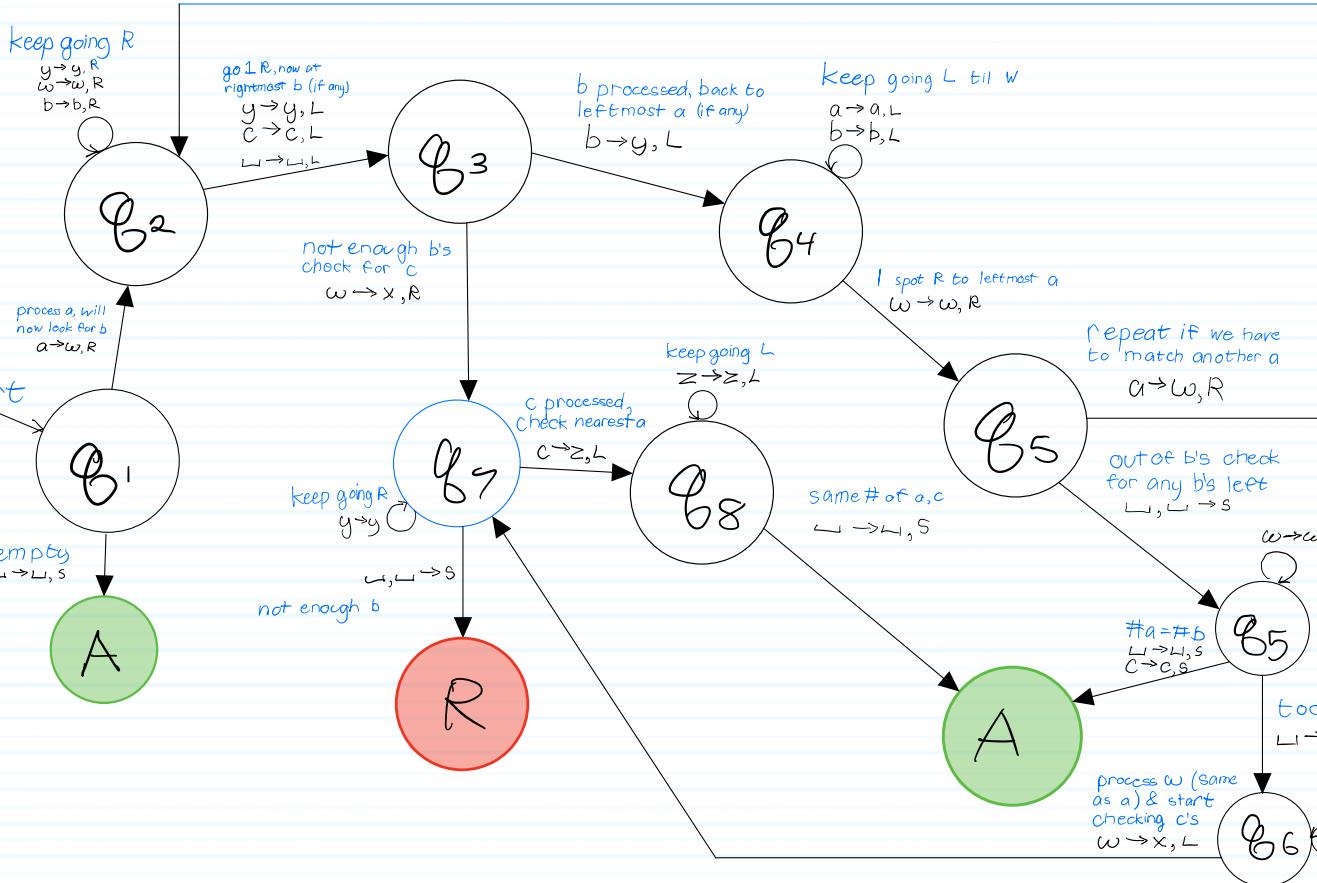
Xxyzc

Xxyzz

xxyzz

2 steps to find outermost a/b

- change  $a \rightarrow w$ , go R to rightmost b (go R till  $\_ | c | x, L$  one)
- Change  $b \rightarrow x$ , go L to leftmost a (go L till  $\_ | x, R$  one)
- repeat as much as possible (no a left | no b right | no a or b)
- if no a going L: go R till b
  - ↳ if  $\_$ : accept ( $\#a = \#b$ )
  - ↳ else if: go L to  $\omega$
  - else if no b going R: go to nearest a
- change  $a/w$  to  $y$ , go R to nearest a
- change  $c$  to  $z$ : go L to nearest  $a/w$
- repeat as much as possible
  - (til no a left | no  $\omega$  left | no c right | nothing left)
- go L till  $a/w$
- go R till  $c$ 
  - if no  $a/w$  left & no  $c$  R: accept



g.  $G = \{W \mid \text{all the strings that contains at least one "aaa"}\}$  over the alphabet  $\{a,b\}$ .  $G = \{\text{aaa, babaaaab, ...}\}$

$$L = \{ \text{aaa, baaa, aaab, abbabaaa...} \}$$

ex  
↓

b	a	b	a	a	b		
---	---	---	---	---	---	--	--

$a \rightarrow a$   
 $b \rightarrow b$   
 $a \rightarrow a$   
 $a \rightarrow a$   
 $a \rightarrow a$   
 $b \rightarrow b$

① if  $b$ :  $b \rightarrow L_1, R$

② if  $a$ :  $a \rightarrow L_1, R$  one

if  $a$ :  $a \rightarrow L_1, R$  one  
if  $a$ : accept

if  $b$  at any point: go to ①  
if  $L_1$  at any point: reject

