

4.3.4b) 2-stack PDAs = TMs

figure out how to encode TM tape in 2 stacks
(can use bottom of stack markers)

4.4.4) how does mply mess up poly-time simulation of RATM
- pay attention!

4.6.2a) ww 2 approaches: 1) generate w^R
reverse w^R

2) for ex for $aabaaba$
generate aaaabbaa
rearrange

b) a^* think of as repeated doublings

c) idea: take advantage of $1=1$ $4=1+3$ $9=1+3+5$ $n^2 = \sum_{i=0}^{n-1} (2i+1)$

produce $\cdot a$, $\cdot a \cdot aaa$, $\cdot a \cdot aaa \cdot aaaaa$

$S \rightarrow e$	} special cases		
$S \rightarrow a$			
$S \rightarrow \cdot a C \$$	initialization	\cdot marks groups of a's	$\$$ marks end of string
$a C \rightarrow C a A$	make A for a	C makes A for each a (copier)	
$A a \rightarrow a A$	move A's to right of a's	D moves to A's after they've been rearranged	
$a A \rightarrow a \cdot A$	put marker between groups	T changes A's to a's	
$\cdot C \rightarrow \cdot D$	change mode	E erases \cdot 's	
$D a \rightarrow a D$	find marker...		
$D \cdot \rightarrow \cdot T$... found marker, start transforming A's		
$T A \rightarrow a T$	transform A to a		
$T \$ \rightarrow a a C \$$	} all transformed; make 2 more and quit or start again	$S \Rightarrow \cdot a C \$$	$\Rightarrow^2 \cdot a \cdot C a A a A a \$$
$T \$ \rightarrow a a E$		$\Rightarrow \cdot C a A \$$	$\Rightarrow^3 \cdot a \cdot C a a a \cdot A A A \$$
$a E \rightarrow E a$	skip over a's	$\Rightarrow \cdot D a A \$$	$\Rightarrow^4 \cdot a \cdot D a a a \cdot A A A \$$
$\cdot E \rightarrow E$	erase \cdot	$\Rightarrow \cdot D a \cdot A \$$	$\Rightarrow \cdot a \cdot a a a \cdot D \cdot A A A \$$
$\cdot E \rightarrow e$	erase E	$\Rightarrow \cdot a \cdot T A \$$	$\Rightarrow^3 \cdot a \cdot a a a \cdot a a a T \$$

if do @ wrong time, can't erase all \cdot 's

if do @ wrong time, can't transform all A's

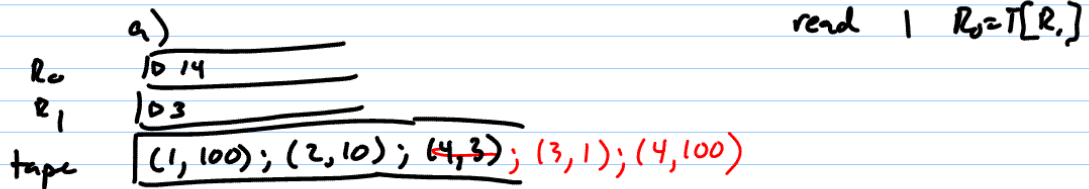
$\Rightarrow \cdot a \cdot a a C a A \$$

$\Rightarrow^4 a a a a a a a a$

Random Access TM

- tape for each reg
- tape for input/output
- tape with $(a, T[a])$ pairs for tape squares w/ non-zero contents
- maybe a temp tape

If RATM computes f in poly time, there is a standard TM that computes f in poly time.



simulating one step scans tape; how long to scan tape depends on length of tape (m)

$$O(m)$$

how fast can tape grow?

for $i=1$ to ∞ step i adds $\log_{10} i + 10$ chars to tape
 $T[i] = 100000$ \uparrow \downarrow
 decimal rep ; (, 100000)
 of i

$pow = 2$
for $i=1$ to ∞ step i adds $O(i)$ chars to tape
 $T[i] = 2^i \cdot pow$
 $pow = pow + pow$

at each step, tape grows at most by $O(\text{size of largest value in reg or on tape})$

largest value on tape or in reg can increase at most by a constant at each step

\therefore size of largest value is $O(t)$

\therefore tape increases by $O(t)$ at each step

\therefore size of tape is $O(t^2)$

\therefore Each of t steps takes $O(t^2)$ to simulate

∴ Simulating t steps takes $O(t^3)$ time.

Nondeterministic TMs

transition relation might have $(q, a), (p, b)$
and
 $(q, a), (s, \rightarrow)$

NTM M accepts a string w if at least one computation path halts on w (semidecides L iff M accepts $w \iff w \in L$)

NTM M decides a language (or computes f) iff

a) no computation path takes longer N steps (N depends on input)

b) every computation path ends in y if $w \in L$
or n if $w \notin L$

(every path ends w/ $f(w)$ on tape)