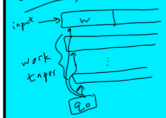
TM — defined
TM behavior — defined

Multitape TMs
High-level descriptions of algorithms

Multitape TM
Def: Let $k \geq 1$. A k-tape TM is a tuple
$$\langle Q, \Sigma, \Gamma, \delta, q_0, q_{acc}, q_{rej} \rangle$$
where everything is as with a TM (1-tape standard) except
$$\delta : Q \times \Gamma^k \longrightarrow Q \times \Gamma^k \times \{L, S, R\}^k$$
L – move left
S – do not move ("stationary")
R – move right

Initially: on input $w \in \Sigma^*$

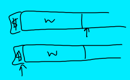


All heads are on leftmost cells
Initial state is q_0

Implementation-level descriptions — not formal, but instead describe tape head movements, etc.

Ex: Reversing a string:

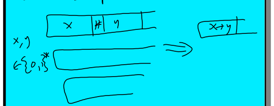
1. heads move to right, w is copied onto the 2nd tape.
2. leaving the head on tape 1 stationary, move the tape 2 head back to the \$



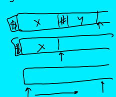
3. moving heads in opposite directions copy each symbol of w on tape 2 to tape 1 (get w^R on tape 1)

Linear time in $n = |w|$.
Best time to do this with a 1-tape machine is $\Theta(n^2)$.

Ex
Binary addition in linear time on a 3-tape machine

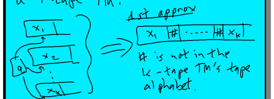


1. copy x onto tape 2:



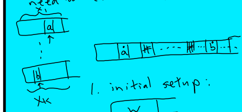
2. move 3rd head to right $|x\#y|$ many cells
3. moving all heads to left write to tape 3 the bitwise sum (with carry bit) for each bit of x, y
[keep track of carry bit in the state of the TM]

Simulating a k-tape TM with a 1-tape TM:

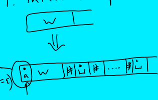


is not in the k-tape TM's tape alphabet.

need to record head positions:



1. initial setup:



a is the first symbol of w (or $\sqcup$ if $w = \varepsilon$)

2. main loop:
Phase 1: scan left to right gathering scanned symbol info into the state

Example state

$q, \langle \underbrace{a_1, a_2, \ldots, a_j, ?, ?}_{k\text{-tuple}} \rangle$

q — current state of the k-tape TM (initially q_0) and the ith tape head is scanning a_i

$?$ — haven't looked at this tape yet.

After Phase 1, all ?'s are gone;

$q, \langle a_1, \ldots, a_k \rangle$

[Phase 1 is read-only]

<u>Phase 2</u>: Letting $\delta(q, \langle a_1, \ldots, a_k \rangle) = (r, \langle b_1, \ldots, b_k \rangle, \langle d_1, \ldots, d_k \rangle)$

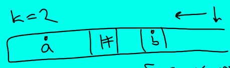
$(d_i \in \{L, S, R\})$

[hard-coded into the state of the 1-tape simulator:]

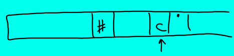
$q, \langle a_1, \ldots, a_k \rangle \longrightarrow r, \langle b_1, \ldots, b_k \rangle, \langle d_1, \ldots, d_k \rangle$

move head from right to left, updating tape contents & marked symbol locations according to the info in the state;

<u>Ex</u>: $k=2$



spse on tape 2, δ says replace b with c & move right.



Until head is leftmost.

Special case

If

& want head to move right,

insert $\dot{\sqcup}$ & bump everything to the right over by one to make room.

To end loop:

$r, \langle \dot{b} \rangle, \langle \dot{\#} \rangle \longrightarrow r, \langle ?, ?, \ldots, ? \rangle$

Slowdown: — 1 step of the k-tape machine is simulated by $\Theta(s)$ steps of the 1-tape TM, where s is the total space used on all the tapes.

Let t = # steps of k-tape machine

Then $s \leq kt = O(t)$

1-tape simulator requires $s \cdot t$ steps, thus $O(t^2)$ steps. <u>Quadratic slowdown</u>