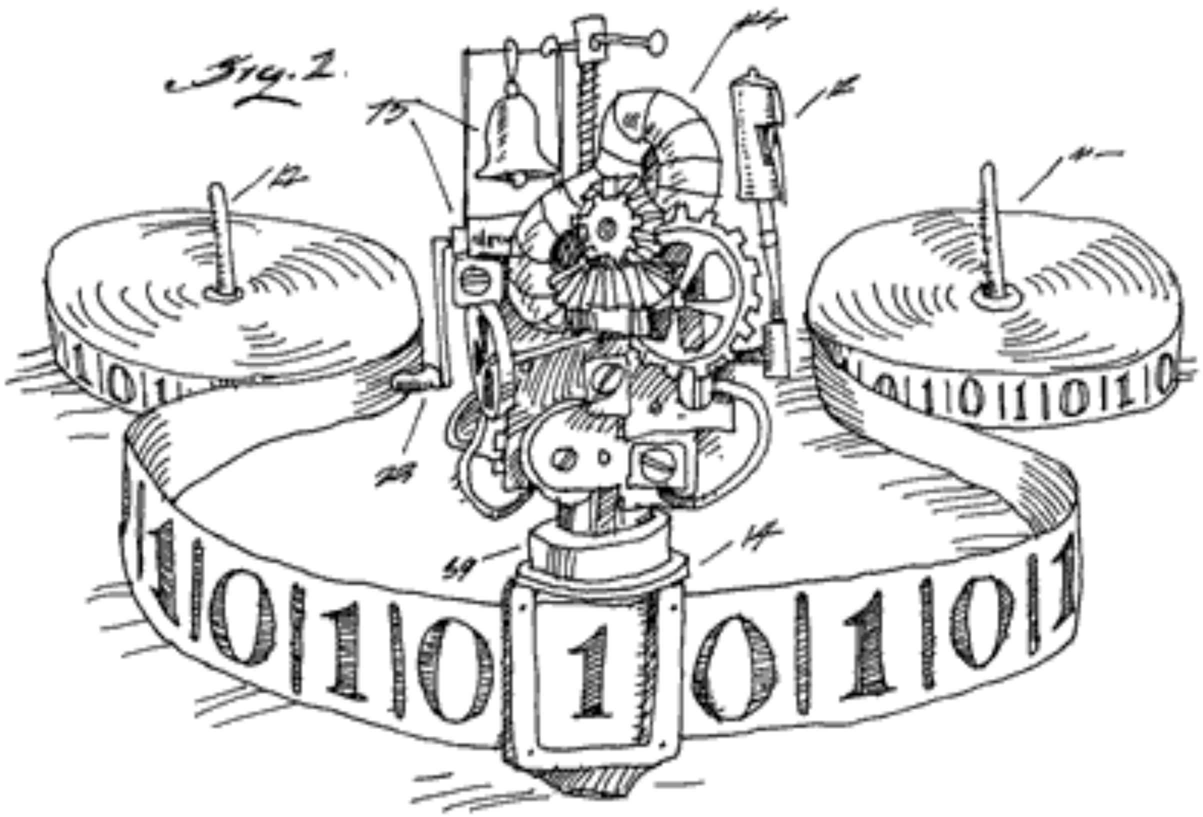
CSE322 Theory of Computation (L15)

Today

Turing Machines

The Turing Machine !!!



TM

= DFA +++++

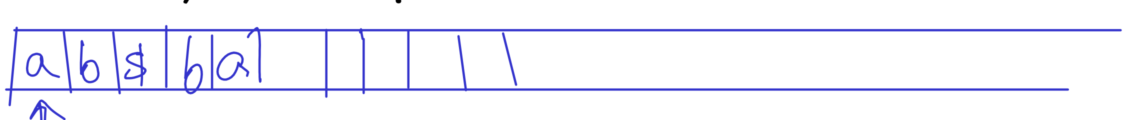
= PDA +++++

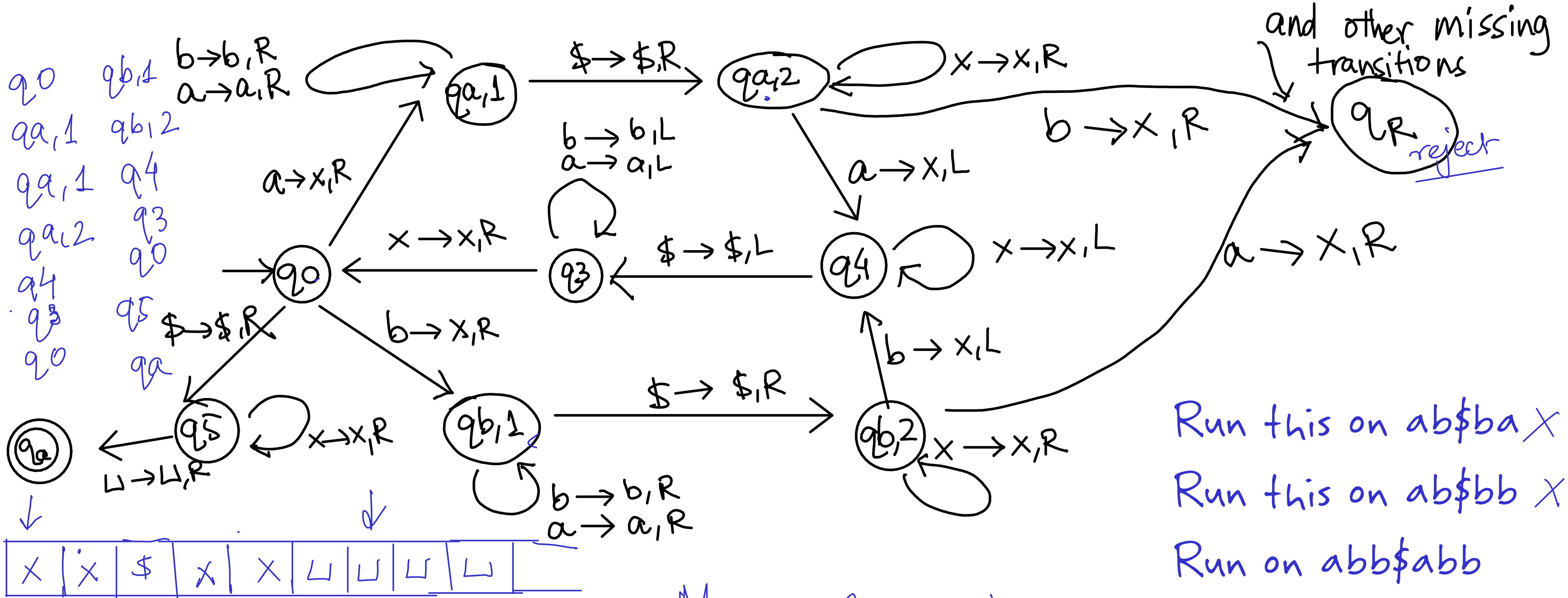
https://www.youtube.com/watch?v=E3keLeMwfHY

Churh-Turing(-Post) Hypothesis :

All reasonable models of (general-purpose) computers are equivalent.

In partiular, they are equivalent to a Turing machine.





M accepts w$w

Run this on ab$ba ✗

Run this on ab$bb ✗

Run on abb$abb

Run on ab$bab

ab$ab ✓

TM = <Q : set of states

Input Alph (does not contain blank symbol) Σ

Tape Alph (contains blank symbol and all input ~~alph.~~ Symbols) Γ

transition fn d : $(Q \times \Gamma) \to (Q \times \Gamma \times \{L, R\})$

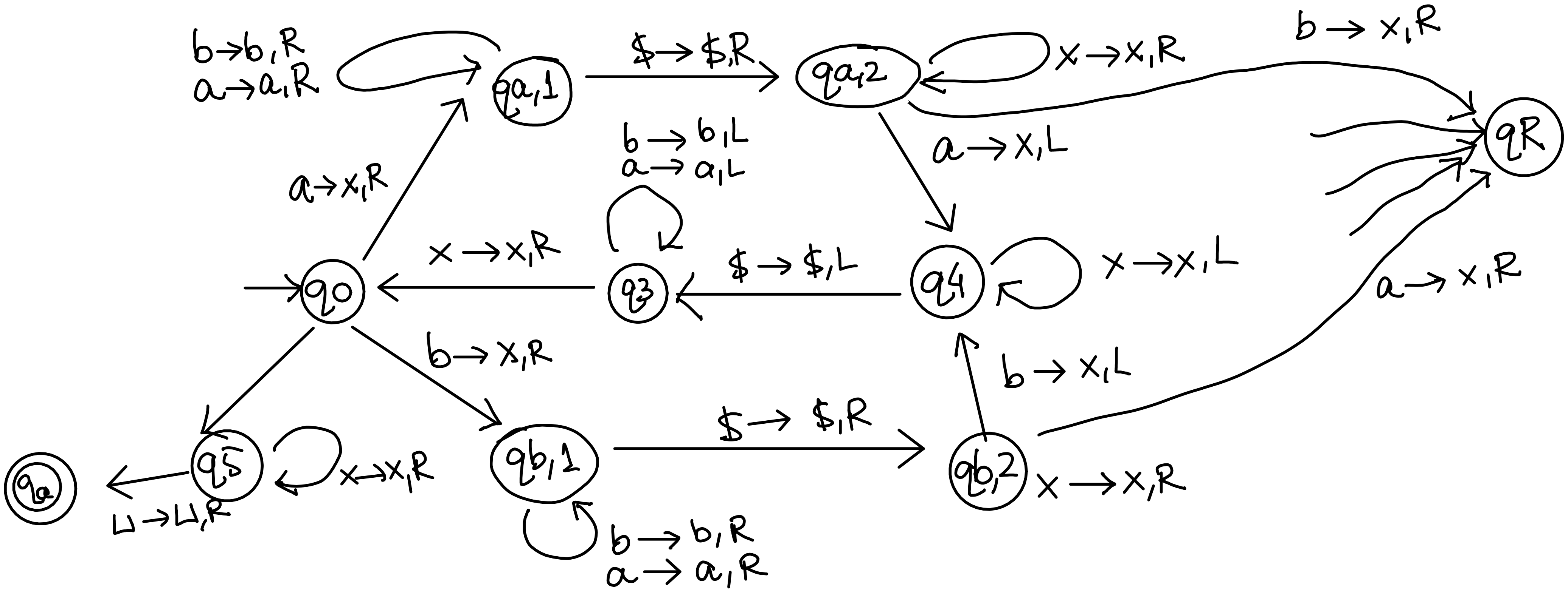
start state q0,

accept state qa,

reject state qr >

If head is on the leftmost cell and d() specifies a left move, head does not move.

	a	b	$\$$	$\sqcup$	x
q_0	$(q_1, \mathtt{x}, \mathrm{R})$	$(q_6, \mathtt{x}, \mathrm{R})$	$(q_5, \mathtt{x}, \mathrm{R})$	reject	reject
q_1 a,1	$(q_1, \mathtt{a}, \mathrm{R})$	$(q_1, \mathtt{b}, \mathrm{R})$	$(q_2, \$, \mathrm{R})$	reject	reject
q_2 a,2	$(q_4, \mathtt{x}, \mathrm{L})$	reject	reject	reject	$(q_2, \mathtt{x}, \mathrm{R})$
q_3	$(q_3, \mathtt{a}, \mathrm{L})$	$(q_3, \mathtt{b}, \mathrm{L})$	reject	reject	$(q_0, \mathtt{x}, \mathrm{R})$
q_4	reject	reject	$(q_3, \$, \mathrm{L})$	reject	(q_4, x, L)
q_5	reject	reject	reject	$(q_{\mathrm{acc}}, \sqcup, \mathrm{R})$	$(q_5, \mathtt{x}, \mathrm{R})$
q_6 b,1	$(q_6, \mathtt{a}, \mathrm{R})$	$(q_6, \mathtt{b}, \mathrm{R})$	$(q_7, \$, \mathrm{R})$	reject	reject
q_7 b,2	reject	$(q_4, \mathtt{x}, \mathrm{L})$	reject	reject	$(q_7, \mathtt{x}, \mathrm{R})$
q_{acc}	No need to define				
q_{rej}	No need to define				



1. Cross off the first character a or b in the input (i.e. replace it with x, where x is some special character)) and remember what it was (by encoding the character in the current state). Let u denote this character.

2. Move right until we see a $.

3. Read across any x's.

4. Read the character (not x) on the tape. If this character is different from u, then it immediately rejects.

5. Cross off this character, and replace it by x.

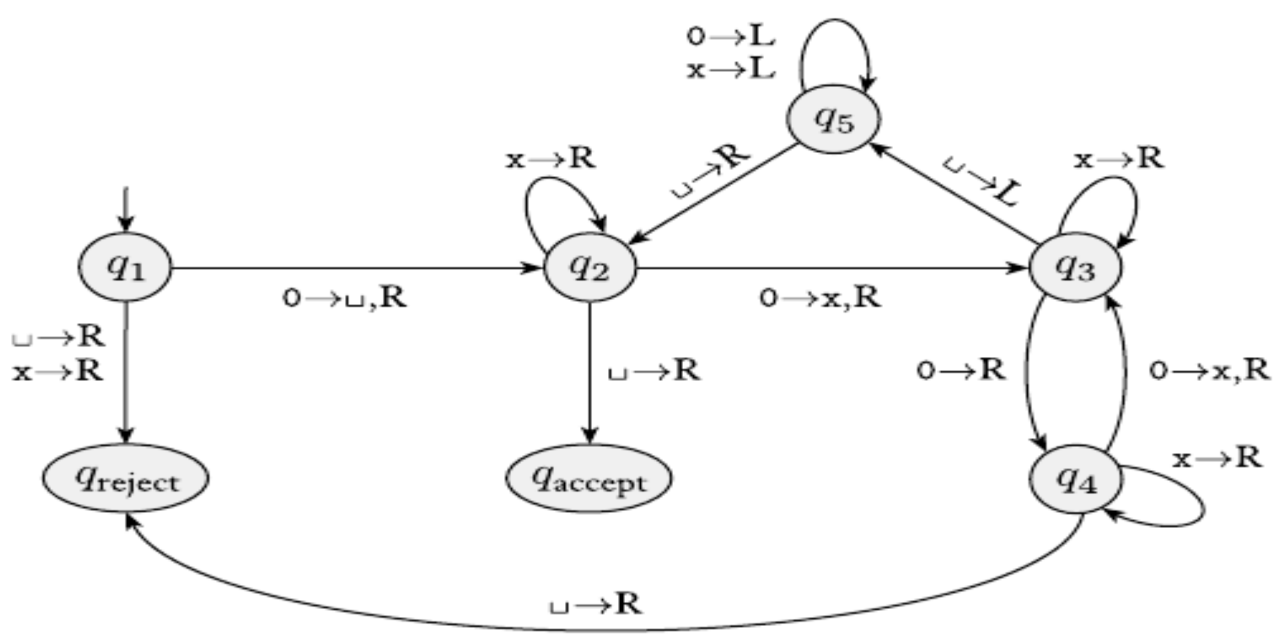
6. Move left past the $ and then keep going until we see an x on the tape.

7. Move one position right and go back to the first step.

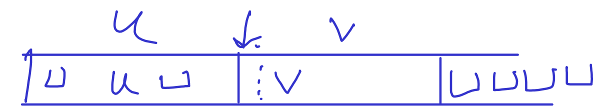
{ w : w is a string over 0 whose length is a power of 2 }

TM(w):

1. Make the first cell blank.
2. Move right until blank is reached. While moving right:
   a. Cross alternate 0
3. When blank is reached,
   a. If tape contains a single 0, go to qa
   b. If tape contains odd number (3 or more) of 0s, go to qr
   c. Else, move left until a dotted cell is found. Goto 2.



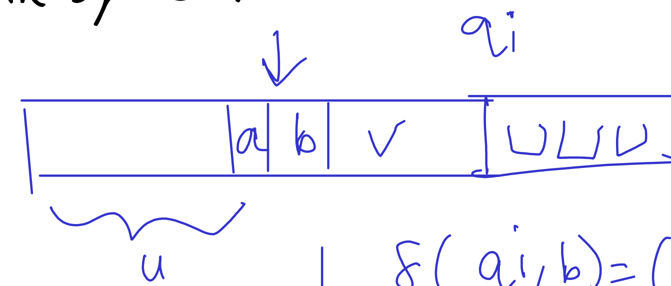
Configuration (instant. description) : u.q.v



u : tape left of head to first ~~non-blank~~ symbol

v : tape from head to last non-blank symbol

q : state

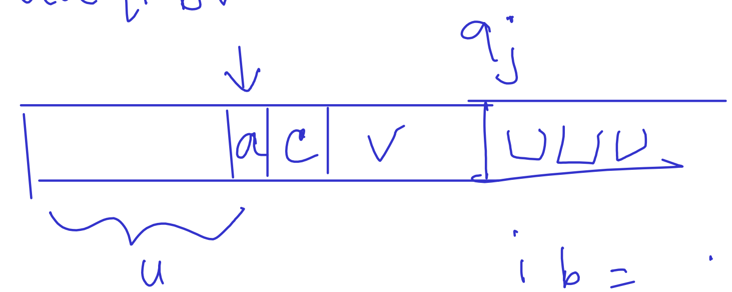


$\delta(q_i, b) = (q_j, c, L)$

$ua\, q_i\, bv$

Yields

Define
$$\begin{cases} ua.q_i.bv \rightarrow u.q_j.acv & \text{if } d(q_i,b) = (q_j,c,L) \\ ua.q_i.bv \rightarrow uac.q_j.v & \text{if } d(q_i,b) = (q_j,c,R) \end{cases}$$



i b =

$u\, q_j\, acv$

Initial configuration on input w.

$q_0 w$

Accepting config. ⇒ state would be q_a.

{ w : w is a string over 0 whose length is a power of 2 }

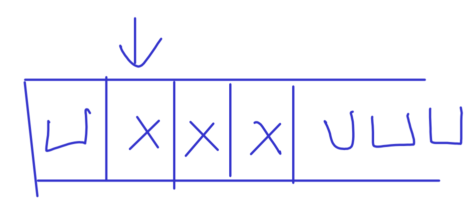
TM(w):

1. Make the first cell blank.
2. Move right until blank is reached. While moving right:
   a. Cross alternate 0
3. When blank is reached,
   a. If tape contains a single 0, go to qa
   b. If tape contains odd number (3 or more) of 0s, go to qr
   c. Else, move left until a dotted cell is found. Goto 2.

$q_1 0000$
$\sqcup q_2 000$
$\sqcup \mathrm{x} q_3 00$
$\sqcup \mathrm{x}0 q_4 0$
$\sqcup \mathrm{x}0\mathrm{x} q_3 \sqcup$
$\sqcup \mathrm{x}0 q_5 \mathrm{x}\sqcup$
$\sqcup \mathrm{x} q_5 0\mathrm{x}\sqcup$

$\sqcup q_5 \mathrm{x}0\mathrm{x}\sqcup$
$q_5 \sqcup \mathrm{x}0\mathrm{x}\sqcup$
$\sqcup q_2 \mathrm{x}0\mathrm{x}\sqcup$
$\sqcup \mathrm{x} q_2 0\mathrm{x}\sqcup$
$\sqcup \mathrm{xx} q_3 \mathrm{x}\sqcup$
$\sqcup \mathrm{xxx} q_3 \sqcup$
$\sqcup \mathrm{xx} q_5 \mathrm{x}\sqcup$

$\sqcup \mathrm{x} q_5 \mathrm{xx}\sqcup$
$\sqcup q_5 \mathrm{xxx}\sqcup$
$q_5 \sqcup \mathrm{xxx}\sqcup$
$\sqcup q_2 \mathrm{xxx}\sqcup$
$\sqcup \mathrm{x} q_2 \mathrm{xx}\sqcup$
$\sqcup \mathrm{xx} q_2 \mathrm{x}\sqcup$
$\sqcup \mathrm{xxx} q_2 \sqcup$
$\sqcup \mathrm{xxx}\sqcup q_{accept}$



M accepts w if ...

there exists a sequence of configurations C1 C2 ... Ck s.t.

$q_0 w$

1. C1 is the starting config. of M on input w
2. Each C(i) yields C(i+1)
3. Ck is an accepting configuration

Similarly, M rejects w if ...

accepts or rejects

M is called a decider if M always halts on any input.

$L = \{w : M \text{ accepts } w\}$

* L is (Turing)-recognizable (or recursively enumerable) if there is some TM M s.t. for all w in L, M(w) halts and accepts.
* L is (Turing)-decidable (or recursive) if there is some TM M s.t. M always halts & for all w in L, M(w) accepts.
* $L(M) = \{ w : M \text{ accepts } w \}$

Are these recognizable? decidable?

{ (a,b,c) : a,b,c are binary strings representing integers & a+b=c}

{ (a1,a2,...,an$k$j) : ai's are binary strings representing integers &
k,j are indices between 1 to n &
the k-th largest integer among {ai}s is aj }

{ (n,p,q) : n,p,q are binary string representing integers &
n has some factor f s.t. p <= f <= q }