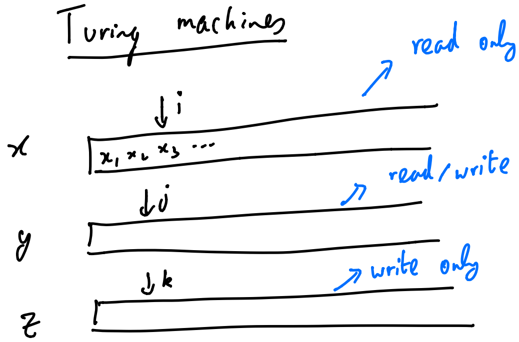
Turing Machines and Boolean Circuits (uniform $f: \{0,1\}^* \to \{0,1\}^*$)

Turing machines



$x \in \{0,1\}^*$

Program

step: (read x_i, y_j) → write to y_j, z_k

→ increment or decrement i, j, k

→ jump to new step or HALT

Example:

1. If x_i is empty, HALT.
   else set $z_k = x_i$, increment i, k, goto 2.
2. If x_i is empty, HALT
   else increment i, k, goto 1.

- Everything we can do efficiently with real computers, we can do with Turing machine and vice/versa.

(Extended) Church-Turing Thesis

Anything that can be computed by a physical process can be computed by a Turing Machine.

Time: Running time $T(n)$ if machine halts after $T(n)$ steps on every input of length n.

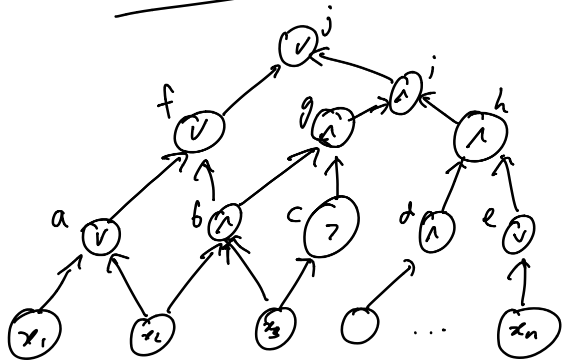
Space: Space is $S(n)$ if max value of j is at most $S(n)$ on every input of length n.

FACT: $S(n) \le T(n)$.

Claim: Any program written for a machine using alphabet of size A that runs in time $T(n)$ can be simulated by our machine in time $O((\log A) \cdot T(n))$.

Claim: Any program for an L-tape machine that runs in time $T(n)$ can be simulated by our machine in time $O(L \cdot T(n)^2)$!

Boolean Circuits (non-uniform) $f: \{0,1\}^n \to \{0,1\}$



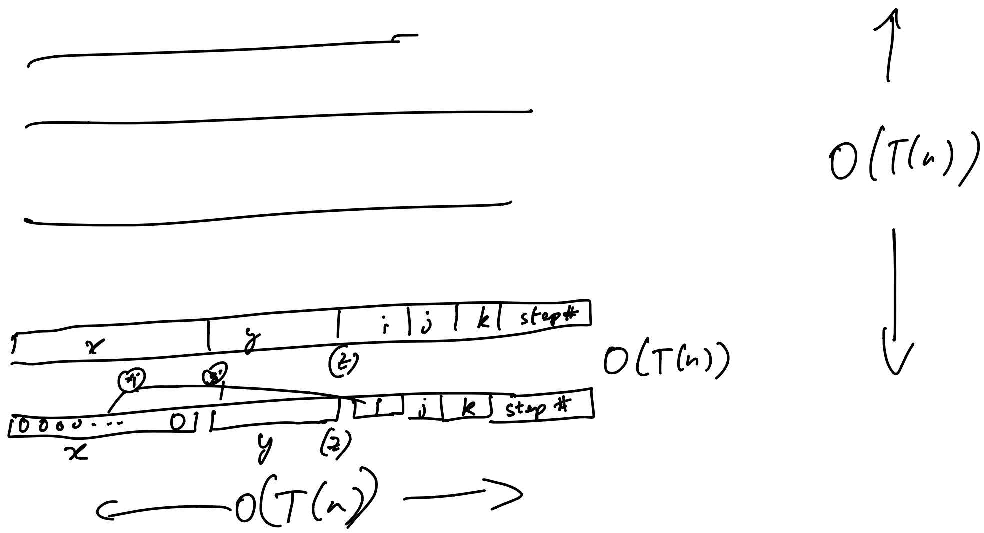
$a = x_1 \vee x_2$
$b = x_2 \wedge x_3$
$\vdots$

Complexity
size : # of nodes
depth : length of longest input → output path.

Open problem: Find a function $f: \{0,1\}^n \to \{0,1\}$ that cannot be computed by a circuit of size $O(n)$.

Open problem: Prove or disprove
Every circuit of size s can be simulated by a circuit of depth $O(\log s)$.

FACT: If $f: \{0,1\}^* \to \{0,1\}$ that can be computed by Turing machines in time $T(n)$, then $\forall n$ there is a circuit of size $O(T(n) \log T(n))$ that computes f on inputs of length n.


x
y
z
i
j
k
step#
O(T(n))
0000 . . . 01
y
z
i
j
k
step #
x
O(T(n))
O(T(n))