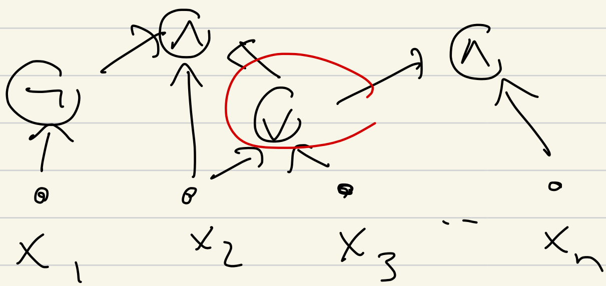
CPSC 536F Jan 18, 2022

Last time:

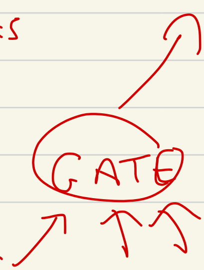
Formulas $\longleftrightarrow$ Tree

Circuits $\longleftrightarrow$ Directed acyclic graph



Circuit

Formula same but each GATE has only one output



For now: Boolean algebra

$\{F, T\}$ or $\{0, 1\}$

GATES: $\neg$ (unary): $x_1 \longmapsto \neg x_1$

$\wedge$ (AND) binary: $x_1, x_2 \longmapsto x_1 \wedge x_2$

$\vdots$

=

Given GATES = $\neg, \wedge, \vee$

any formula has a version (with same size = # occurrences of variables) s.t. all $\neg$ occur ~~just~~ the level just above variables.

There's no loss of generality!

formula $x_1, \dots, x_n$ variables

being given as binary tree,

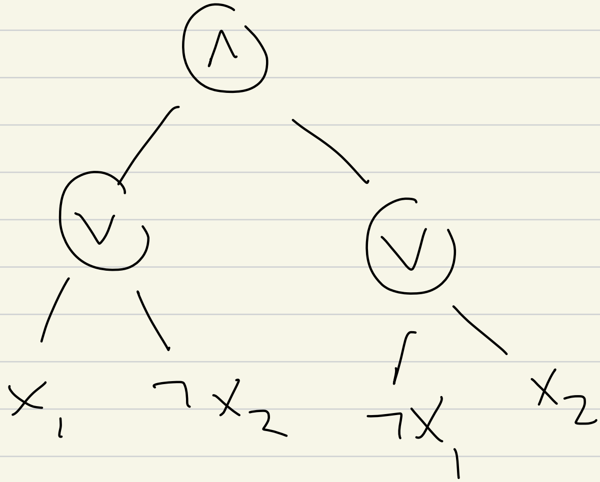
just $\wedge, \vee$ GATES and

leaves

$$\text{"literals"} \begin{cases} x_1, \dots, x_n & \text{variables} \\ \neg x_1, \dots, \neg x_n & \text{negated variables} \end{cases}$$

"De Morgan formula" is just

such a tree!



=

Claim: Any Boolean function can be expressed as

① truth table representation:

say

x_1	x_2	x_3	$f(x_1, x_2, x_3)$
T	T	F	T
F	T	T	T
T	F	~~F~~	T

false otherwise

at most 2^n values where

f is T,

either $\boxed{x_1 = T \quad x_2 = T \quad x_3 = F}$ 1

OR

$x_1 = F \quad x_2 = T \quad x_3 = T$ 2

⋮

$x_1 \wedge x_2 \wedge \neg x_3 = T$

OR

$\neg x_1 \wedge x_2 \wedge x_3 = T$

OR

⋮

(① is T) OR (② is T) OR ...

$$\underbrace{(x_1 \wedge x_2 \wedge \neg x_3)}_{\text{has } n \text{ literals}} \vee \underbrace{(\neg x_1 \wedge x_2 \wedge x_3)}\vee \ldots$$

if you have n variables

$\Rightarrow$ f can be written via T values in f's truth table with $\leq n2^n$ size formula

One improvement (by Victor)

$$\left(\begin{array}{l} \text{If } x_1 \text{ is T, then } f(\overbrace{T, x_2, \ldots, x_n}^{g(x_2,\ldots,x_n)}) \\ \qquad\qquad\qquad\qquad\qquad \text{is T} \\ \text{OR} \\ \text{If } x_1 \text{ is F, then } f(\overbrace{F, x_2, \ldots, x_n}^{\tilde{g}(x_2,\ldots,x_n)}) \\ \qquad\qquad\qquad\qquad\qquad \text{is T} \end{array}\right)$$

$$\left(f(x_1, \ldots, x_n) \text{ is T} \right) \underset{\times}{\text{iff}}$$

$$x_1 \wedge [f(T, x_2 \ldots) = T] \vee (\neg x_1 \wedge [f(F, x_2, \ldots, x_n) = T])$$

Hence is

$L(n) =$ length of the max formula size to express any Boolean of n variables

$$L(n) \leq 2 + 2L(n-1).$$

$L(1)$: Four functions of x_1 for $n=1$

$$T, F, x_1, \neg x_1$$

So $L(1) = 1$

functions constants $T, F \leftarrow$ size 0

Exercise: Find $L(n)$, show that $L(n) \sim 2^n$.

O, Θ.

If $f, g : \mathbb{N} = \{1, 2, \ldots\} \to \mathbb{N}$

we write

$f = O(g)$ if for n_0, C

$$f(n) \leq g(n) C \text{ for } n \geq n_0$$

and $f = o(g)$ if

for any $c > 0$, $\exists n_0$ s.t.

$$f(n) \leq c\, g(n) \quad n \geq n_0$$

=

$$f = o(g) \iff \lim_{n \to \infty} \frac{f(n)}{g(n)} = 0$$

=

Example: $f(n) = n^2 + 5n \log n$

$$= n^2 + o(n^2)$$

O = order, o = asymptotically less than.

So

$$f(n) = 10n^2 + 5n$$

$$= O(n^2)$$

Since

$$f(n) = 10n^2 + 5n \leq$$

$$10n^2 + 5n^2 \leq 15n^2$$

Richard Karp:

"The only constant a comp. sci. theoretician cares about is the one in their salary."

Writing

$$f(x_1, \ldots, x_n) = T$$

$$\iff$$

$$\text{OR}\begin{pmatrix} \text{all } x_1, \ldots, x_n \text{ where} \\ f(x_1, \ldots, x_n) = T \end{pmatrix}$$

$$\iff$$

$$f(x_1, \ldots, x_n) = T$$

iff $\neg(\neg f(x_1, \ldots, x_n)) = T$

$$\neg f(x_1, \ldots, x_n) = F$$

we can also write

$f(x_1, \ldots, x_n)$ formula by

its false values.

x_1	x_2	$f(x_1, x_2)$
F	F	T
F	T	T
T	F	F
T	T	T

$f = T \iff$ one of its variable settings gives T

$\neg f = F \iff \neg$(one of f's false settings is T)

$$f = T \iff$$

$$f = F \iff x_1 = T, \quad x_2 = F$$

$$\iff x_1 \land (\neg x_2) = T$$

$$\updownarrow$$

$$x_1 \text{ AND } (\neg x_2) = T$$

$$f = F \iff x_1 \land (\neg x_2)$$

$$f = T \iff \neg(\underbrace{x_1}_{p} \land \underbrace{\neg x_2}_{q})$$

$$\iff$$

$$\neg(p \land q) \iff (\neg p \lor \neg q)$$

$\iff (\neg x_1 \vee x_2)$

When you write

$f = T \iff$ (some ~~literal~~ all T)

OR

(- - - - T)

OR

(- - -)

$\downarrow$

(x_1 AND $\neg x_2$ AND) OR ( ) OR ( ) ...

OR of ANDS

called OR-normal formal

$\vee$ = OR = disjunctions

( ) OR ( ) or ... or ( )

AND's

Disjunctive normal form.

= DNF

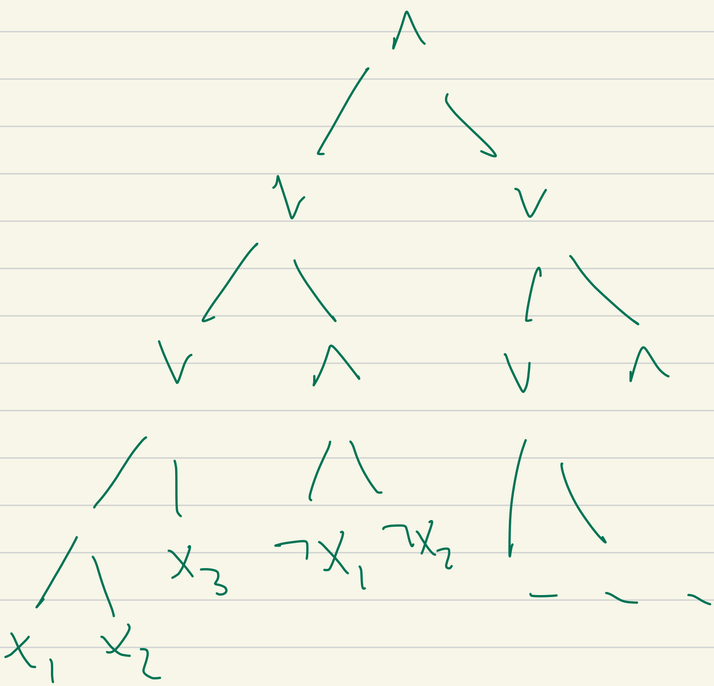
(or ... or) AND (... or ... or) ... – –

Conjunctive normal form

(Conjuction $\Leftrightarrow$ AND) = CNF

Regarding formulas or circuits on n variables!

De Morgan formulas:



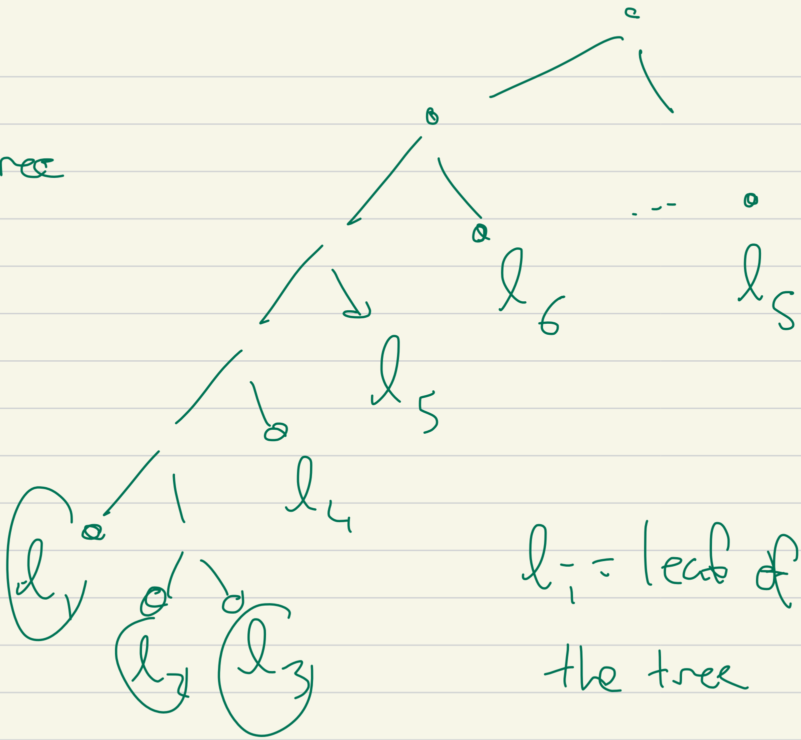
Thm (Shannon):

Most Boolean functions on n variables cannot be expressed as a formula of size $\leq 2^n/3n$ (for large n).

$=$

Lemma: How many formulas are there of size $= S$?

Binary tree



l_i = leaf of the tree

leaves $x_1, \dots, x_n, \neg x_1, \dots \neg x_n$

Formula $\iff$ (0) a binary tree with

(1) s leaves

(2) each interior nodes, of which there are $s-1$, is $\wedge, \vee$

$$\begin{pmatrix} \#\text{ formulas} \\ \text{size } s \end{pmatrix} \le \begin{pmatrix} \#\text{ binary trees} \\ s \text{ leaves} \\ + \text{ one root} \end{pmatrix} (2n)^{s} (2)^{s-1}$$

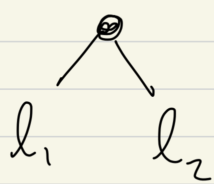
each of s leaves is one of

$x_1, \ldots, x_n, \neg x_1, \ldots, \neg x_n$

=

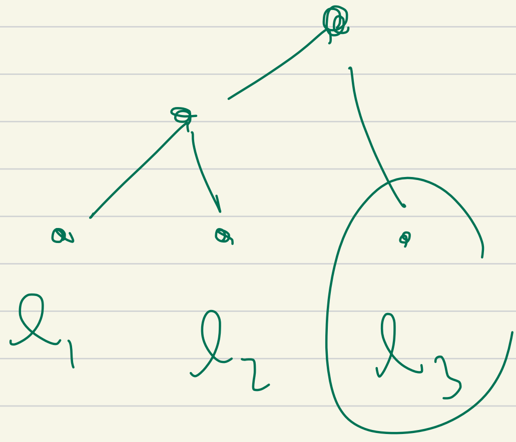
How many binary trees are there?

$s = 2$, $\quad l_1, l_2$



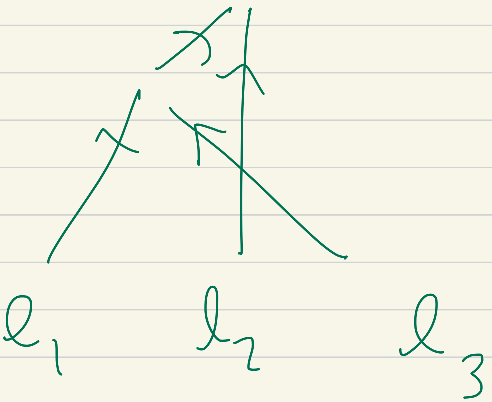
only one possible tree

$S = 3$

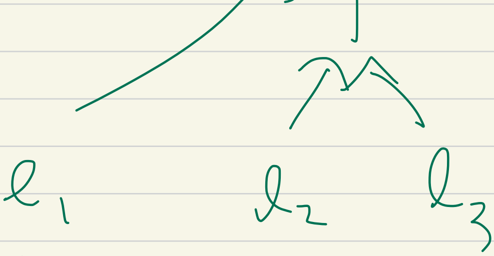


2nd gate

or



3 trees.

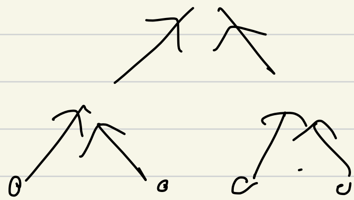


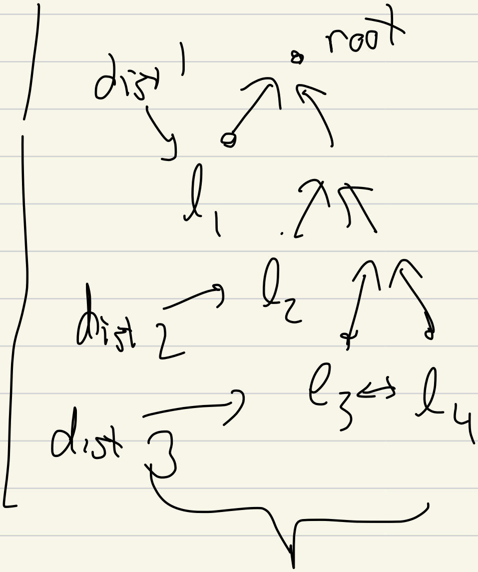
in 2nd gate

What about $s=4$, $s=5$, ... ?

$=$

$s=4$





has $\mathbb{Z}_2$ symmetries

$=$

Ignore symmetries for now

Fact: there is a general formula for # distinct rooted trees with s leaves (all distinct from root)

Break! 10:22 — 10:27

=

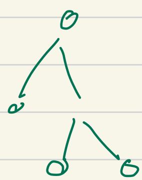
Philip bound $\quad s!/2$

Amr bd $\quad \dfrac{s! \cdot (s-1)!}{2^{s-1}}$

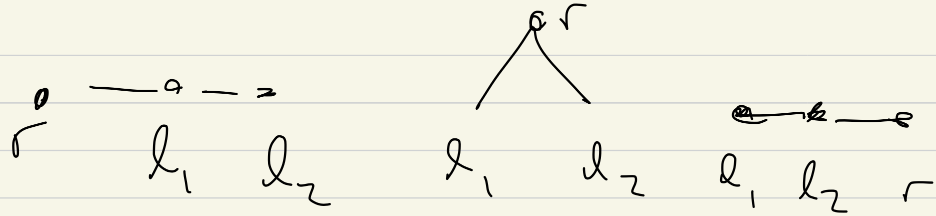
=

$s = 4$:

Bound:



trees on $s+1$ nodes with a given node fixed as root: $\boxed{(s+1)^{s-1}}$



$(S+1)^{S-1}$ based on Laplacian,

which counts # trees on

$S+1$ nodes with a given root.

=

Let's take $(S+1)^{S-1}$

=

$$\begin{pmatrix}\#\text{ formulas}\\ \text{size } S\\ \text{is}\end{pmatrix} \le (S+1)^{S-1}(2n)^{S}\,2^{S-1}$$

# of Boolean functions on n variables : $2^{(2^n)}$

Claim: $S = 2^n / nc$

c large enough, $c=3$ should suffice,

then

$$\boxed{(S+1)^{S-1} (2n)^{S} 2^{(S-1)}}$$

$S = 2^n / nC$

$$\boxed{\leq \left(\frac{1}{2}\right) \left(2^{(2^n)}\right)}$$

Let's take $\log_2$ both sides!

we want

$$\log_2\left(\frac{1}{2}\,2^{(2^n)}\right)$$

$$= \underbrace{\log_2\left(\frac{1}{2}\right)}_{-1} + \underbrace{\log_2 2^{(2^n)}}_{2^n}$$

$$= 2^n - 1$$

(not so bad)

$$\log_2 \left((s+1)^{(s-1)} (2n)^{s} \, 2^{(s-1)} \right)$$

$$=$$

$$(s-1) \log_2 (s+1) + \quad ①$$

$$(s) \log_2 (2n) + \quad ②$$

$$(s-1) \log_2 2 \quad ③$$

$$s = 2^n / nC$$

which term is largest ?

Settle it this way

$$f_1(n) = (S-1)\log_2(S+1)$$

$$S = 2^n/nc\ ,\quad c \text{ fixed}$$

$$f_1(n) = \left(\frac{2^n}{nc} - 1\right)\log_2\left(\frac{2^n}{nc}+1\right)$$

$$f_2(n) = \left(\frac{2^n}{nc}\right)\log_2(2n)$$

$$f_3(n) = \left(\frac{2^n}{nc} - 1\right)\cdot 1$$

Can we claim!

$$\boxed{f_2 = o(f_1) \quad ?}$$

$$\lim_{n\to\infty} \frac{f_2(n)}{f_1(n)} = 0 \quad ??$$

$$\lim_{n\to\infty} \frac{f_2(n)}{f_1(n)} = \lim_{n\to\infty} \frac{\left[\left(2^n/n^c\right)\right] \log(2n)}{\left[\left(2^n/n^c - 1\right)\right] \log\left(\frac{2^n}{n^c}+1\right)}$$

$$\lim_{n\to\infty} \frac{2^n/n^c}{2^n/n^c - 1} = 1$$

$$\lim_{n \to \infty} \frac{\log^{(2n)}}{\log\left(\frac{2n}{nc} + 1\right)} = \log_{n \to \infty} \frac{\log_2 2 + \log_2 n}{\log\left(\frac{2^n}{nc}\right)}$$

$$= \log_{n \to \infty} \frac{1 + \log_2(n)}{n - \underbrace{\log(nc)}_{\log c + \log n}}$$

$$= \lim_{n \to \infty} \frac{\text{Order}(\log n)}{n - \text{Order}(\log n)}$$

$$\to 0$$

So

$$f_2(n) = f_1(n) \cdot \underbrace{o(n)}_{\substack{\text{function} \\ \text{that} \to 0 \\ \text{as } n \to \infty}}$$

Similarly

$$f_3(n) = \left(\frac{2^n}{nc} - 1\right)$$

$$< \frac{2^n}{nc}$$

larger

$$< \frac{2^n}{nc} \boxed{\log_2(2n) = f_2(n)}$$

$f_3(n) = o(f_2(n))$

or $\leq$ Order$(f_2(n))$

and

$f_3(n), f_2(n) \leq f_1(n)\, o(n)$

function $\to 0$

as $n \to \infty$

So

$\log\left((s+1)^{(s-1)}(2n)^s 2^{(s-1)}\right)$

$$= f_1(n) + f_2(n) + f_3(n)$$

$$= f_1(n)\left(1 + o(1) + o(1)\right)$$

$$= f_1(n)\left(1 + \begin{matrix}\text{a function} \\ \to 0 \\ \text{as } n \to \infty\end{matrix}\right)$$

$=$

Want to know:

is

$$f_1(n) + f_2(n) + f_3(n) < 2^n - 1$$

for $n \to \infty$. Finish next time...

$=$

Class ends...