<circa 1979>

Some Notes on Teaching Boolean Algebra

by

Louis H. Kauffman

I) Introduction

These notes constitute a sketch of some ideas for teaching boolean algebra that I have found particularly useful. I feel that the approaches sketched here are particularly helpful on a number of levels:

We shall begin with a very simple symbol system that is mildly geometric. This makes for an easy introduction for students harboring the usual fears of symbolics. It also allows for discussion of just what it is we _do_ when we set out a formal system. Finally, this system leads immediately into some simple problems involving recursive calculations (hence its interest for computer science).

Recall that a _boolean algebra_ is a set B with binary operations $+, \times$ and a unary operation $a \mapsto a'$ satisfying:

1) $+$ and $\times$ are commutative, associative.
2) $+$ and $\times$ distribute over each other.

   That is: $a + (b \times c) = (a+b) \times (a+c)$

   $a \times (b+c) = (a \times b) + (a \times c)$

   $\forall a,b,c \in B$.
3) $(a')' = a \ \forall a \in B$.
4) $\left.\begin{array}{l} a \times a = a \\ a + a = a \end{array}\right\} \ \forall a \in B$.

5) There exist unique elements $0, 1 \in B$ so that $0 \times a = 0$, $1 \times a = a$

$0 + a = a$, $1 + a = 1$

$\forall a \in B$. And $0' = 1$.

6) $(a+b)' = a' \times b'$, $\forall a,b \in B$.

7) $\left.\begin{array}{l} a + a' = 1 \\ a \times a' = 0 \end{array}\right\} \forall a \in B.$

<u>Examples of boolean algebras abound</u>:

i) Let B = the set of subsets of a set X.
Let $0 = \emptyset$, the empty set, and $1 = X$.
Let $a + b = a \cup b$ (union of a and b)
and $a \times b = a \cap b$ (intersection of a and b).

ii) Let B be the set of diagrams composed from elementary diagrams of type —→— p —→— and —→— p′ —→— where $p \in \mathcal{L}$ ($\mathcal{L}$ a set of symbols). The elementary diagrams are interpreted as <u>single-throw switches</u>.

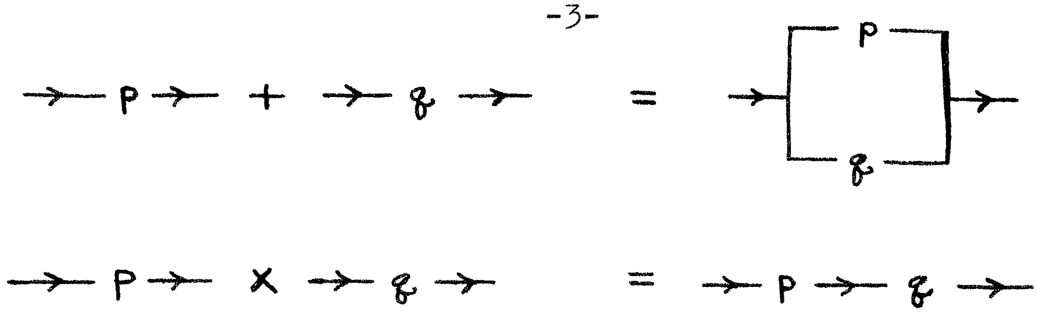
p

p = 1

p = 0

Each letter $p \in \mathcal{L}$ can take the two values $p = 0$, $p = 1$. For a switch labelled p, 0 denotes "Open" and 1 denotes "closed". (For p' it is the opposite.)

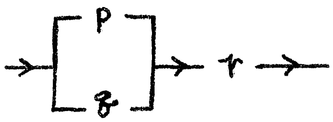
We define addition and multiplication of elementary diagrams:



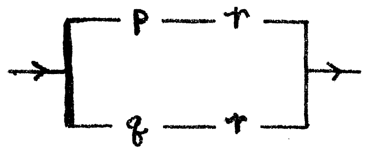
In general an element of B will be a diagram of type  where the inside of the box is some switching network whose only free ends are the input and output of the box.

The boolean identities refer to the behavior of these nets:

Thus (p + q) x r = (p x r) + (q x r) means that

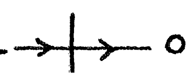


and

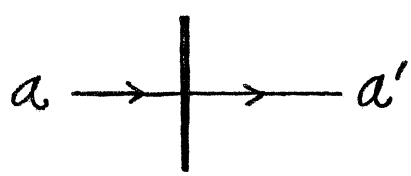


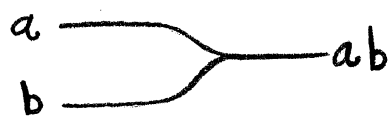
have the same switching behavior.

This important example is due to Claude Shannon and forms the essential basis for designing computer circuitry.

iii) Let  denote a device that <u>inverts a signal</u>. That is, we now imagine digital signals flowing through the network and 0  1, 1  0 indicate that 0 inverts to 1 and 1 inverts to 0.

a)

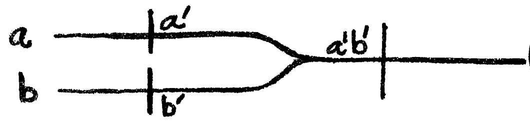


β) 

Concurrence of wires will denote boolean multiplication.

(Thus the 0-signal dominates the 1-signal.)

Addition can be manufactured:



$(a'b')' = a'' + b'' = a + b.$

This third example serves to introduce our symbol system.

We write $\overline{a}\rceil$ for a'.

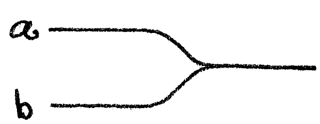
That's all. The idea is due to G. Spencer Brown in his book Laws of Form.

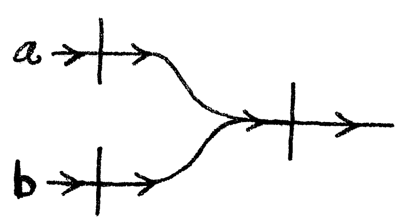
This has the advantage of letting us eliminate parentheses:

$$(a'b')' = \overline{\overline{a}\rceil\ \overline{b}\rceil}\rceil$$

(We have already stopped writing a x b and just use ab.)

$a' = \overline{a}\rceil =$ a 

ab = 

$a + b = (a'b')' = \overline{\overline{a}\rceil\ \overline{b}\rceil}\rceil =$ 

<u>Brown had one other notational idea</u>:

Let ⌉ denote 0

Let (blank) denote 1!

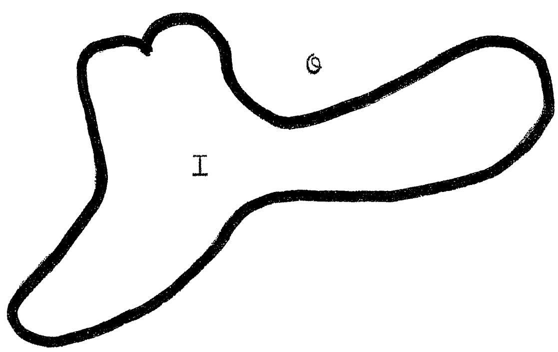
Then ⌉⌉ = 0 x 0 = 0 = ⌉

while $\overline{\urcorner}|$ = 0′ = 1 = (blank)

<u>Hence</u>:

⌉⌉ = ⌉
$\overline{\urcorner}|$ =

iv) This last business may be clarified by the following consideration: We are really talking about a simple binary distinction such as inside (I) versus outside (O).



We let ⌉ denote the operation of changing sides so that

$\overline{O}|$ = I

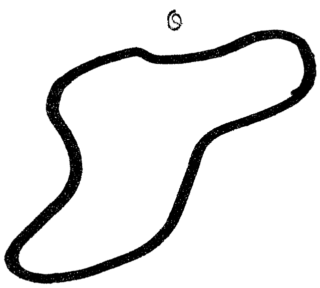
$\overline{I}|$ = O

We agree that II = I and OO = O (redundancy of name-calling).

This gives the mini-boolean algebra isomorphic to {0,1}.

(If we decide that O <u>dominates</u> so that OI = O. But, see next page...)

Now suppose we are <u>so lazy</u> that we decide to indicate the outside by @ but the <u>inside</u> by (blank).



<u>Then</u>: $\overline{@}|$ = (blank)

$\overline{\text{(blank)}}|$ = @

<u>or</u>: $\overline{@}|$ =

$\overline{\;\;}|$ = @ (!)

and substituting, $\overline{\overline{\;\;}|}|$ = .

Similarly, @@ = @ gives $\overline{\;\;}|\overline{\;\;}|$ = $\overline{\;\;}|$. Once again we have

$$\boxed{\begin{array}{l} \overline{\overline{\;\;}|}| \; = \\ \overline{\;\;}|\overline{\;\;}| = \overline{\;\;}| \end{array}}$$

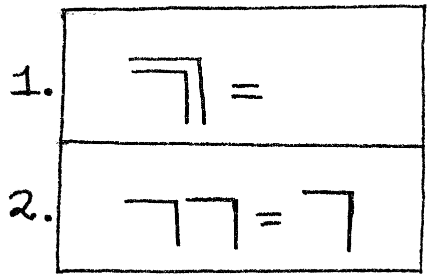
.

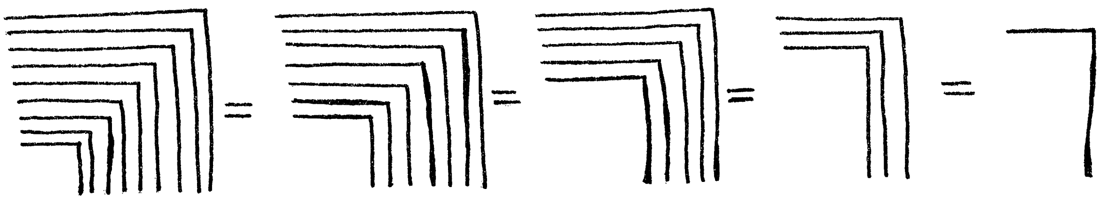
It is easy to wax philosophical at this point, but we shall refrain! Note however that $\overline{\;\;}|$ has a dual interpretation: sometimes as <u>operator</u> $\overline{X}|$ = X′, sometimes as <u>operand</u> $\overline{\overline{\;\;}|}|$ = $\overline{\;\;}|'$ = 0′ = 1 = (blank). But then, so does the lowly pawn in chess have a <u>dual</u> <u>role</u>.

(Since I = (blank), @ automatically dominates:)

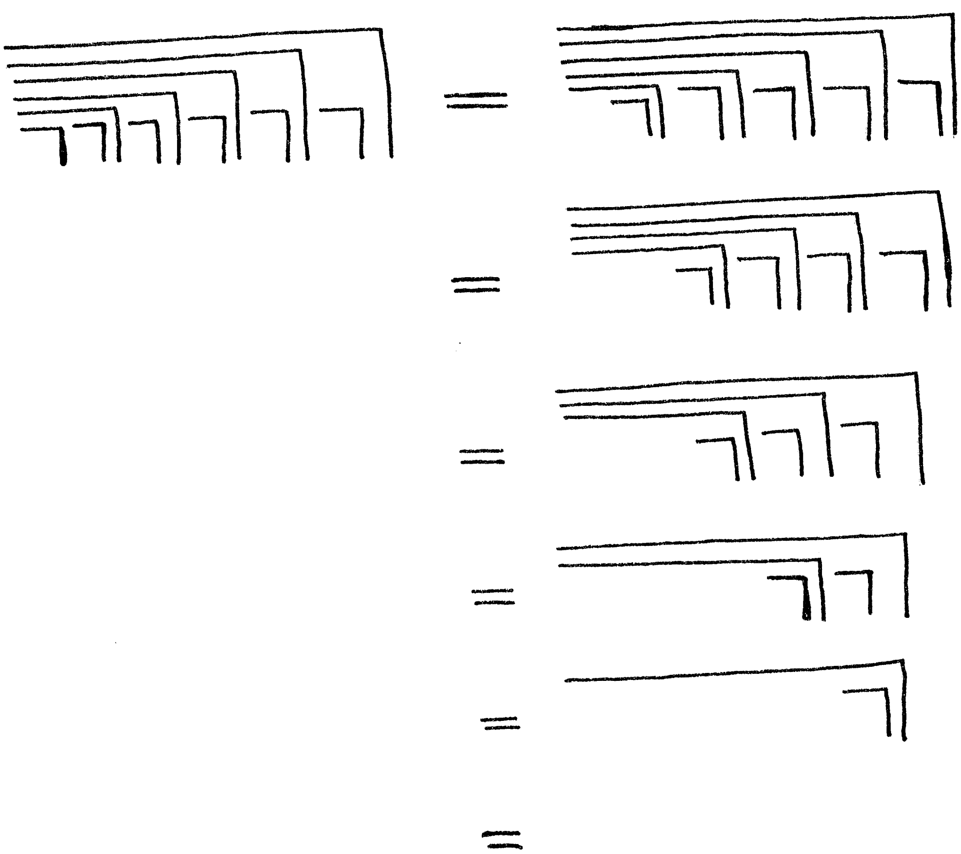
$$\boxed{I@ = @}$$

II) Marks and Crosses





(1.)



(2.)

(1.)

(1.)

(1.)

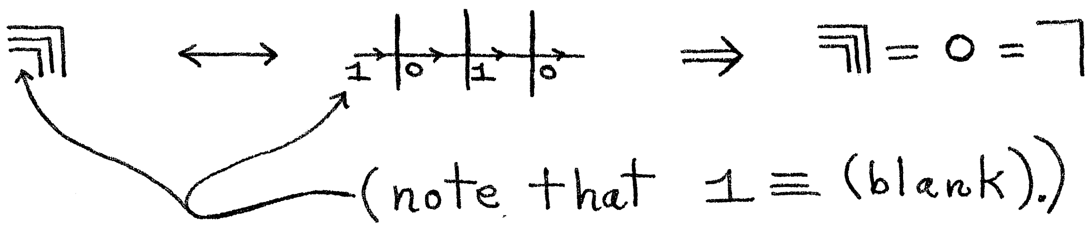
(1.)

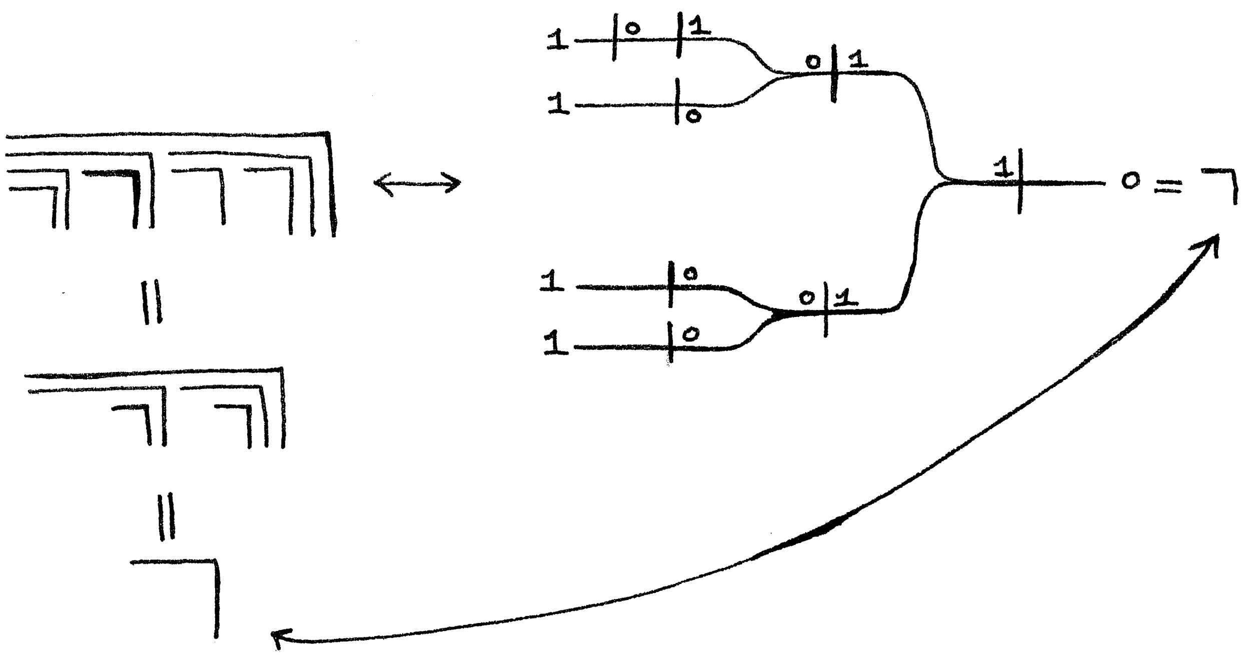
(1.)

You can play lots of calculational games following these rules.

Its a very nice exercise to prove that every expression reduces uniquely to ⏋ or (blank). The problem of <u>uniqueness of value</u> (that you can't get from ⏋ to (blank) via steps of type 1 and 2) is a nice prototype for many recursive situations. We have a method of calculation that can lead to the same result by many different paths. What is needed is a <u>standard method</u> whose result does not change under moves of type 1 and 2.

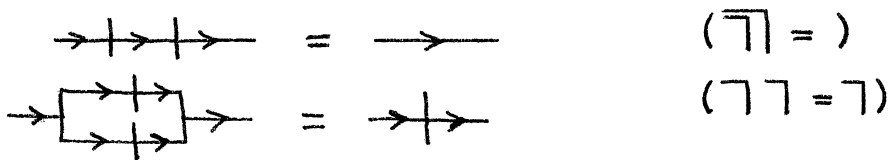
In this case we just view the expression as an <u>inverter net</u> and process the signals!





Since, in a net

and



we see that uniqueness of valuation follows from the net-work behavior.

III) Linear and Non-Linear Language

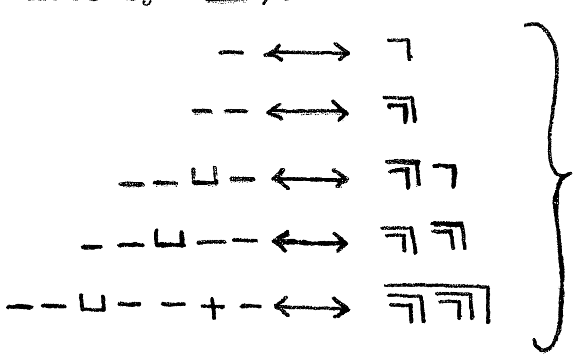
The language involving ⌉ is non-linear in that it involves two planar dimensions:

I set the following exercise to my class of computer science students: Design a linear, parenthesis free language that describes elements of the ⌉-language. Your linear language should be suitable for use in a graphics program that draws ⌉-expressions.

———————— o ————————

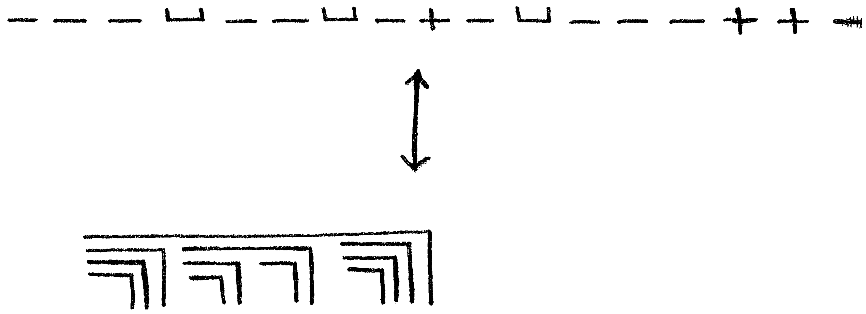
There are many possible answers to this. The best was invented by Sharon Saluski. She actually did write a <u>Plato</u> graphics program and this is available and working somewhere in SEO.

<u>Her solution</u>: Use symbols - , + and (space) (which we denote by ␣).



+ joins expressions separated by a blank. Iterated +'s cause a search for corres # of ␣'s for joining up.

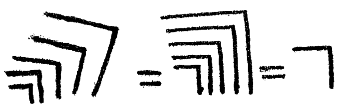
Thus



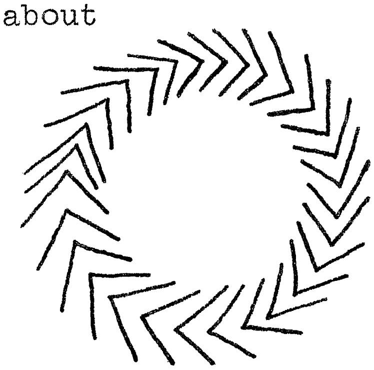
In any case, there is ample room for concrete discussion of various cases of language and language representation.

It is also interesting to see how adept one becomes at visually computing values in ⌉-language. This leads to various quite practical discussions about pattern recognition and parallel processing.

IV) Conventions

 OK.

What about

e =  ?

If you search for the bottom of e you get in a loop.

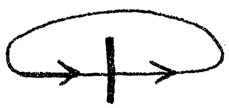
If you send a signal thru e, then it <u>oscillates</u> just as in

$0 \to 1 \to 0 \to 1 \to 0 \to 1 \to 0 \to \cdots$

Examples like this can be used to discuss conventions (and tacit conventions) about the limits of use of the formal system.

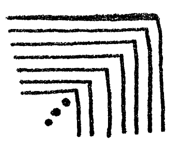
Anyway, self-contradicting loops lead to:

IV) <u>The Liar</u> (This statement is false.)

 is like $X = \overline{X}|$

$X = 0 \Rightarrow X = \overline{0}| = 1 \Rightarrow X = 0 \Rightarrow X = 1 \Rightarrow \ldots$

Two ways out:

A) $X =$  (ad infinitum) .

Then <u>geometrically</u> $X = \overline{X}|$ and we need to figure out how to deal with it <u>algebraically</u>.

B) Let the <u>process</u> be the solution:

i : ...010101010101O1...

j : ...101010101010lO...

These are the two sequences that come from $X = \overline{X}|$ depending on whether you set $X = 0$ or $X = 1$ to start. <u>In what sense do we</u>

have $\overline{\imath}\rceil = \dot{\imath}$? Answer: Reinterpret $\rceil$ for sequences as ordinary inversion plus a half-period shift.

$$s = \ldots a\ b\ a\ b\ a\ b\ a\ b \ldots$$

$$\Rightarrow \overline{s}\rceil = \ldots b'\, a'\, b'\, a'\, b'\, a'\, b'\, a' \ldots$$

Then $\overline{\imath}\rceil = \dot{\imath}$ $\quad$ $\overline{\jmath}\rceil = \dot{\jmath}$

$$\dot{\imath}\,\dot{\jmath} = \rceil \,.$$

This leads naturally into some non-standard boolean algebras (De Morgan Algebras).

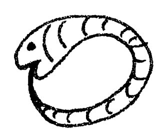
Both A) and B) can be expanded on considerably.

A) leads to studying fractals (self-similar forms, space filling curves...).

B) leads to non-standard logics and algebras.

VI) Fractals and Self-Similar Forms.

Let f =  . The little hook indicates that the form "re-enters" itself.



Notation due to Spencer-Brown, Francisco Varela and Ancient Mythology.

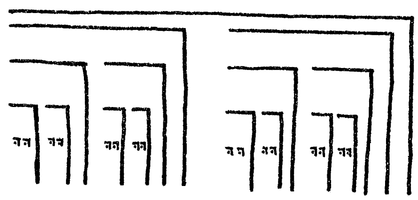
Here equality just means as a geometrical (infinite) nest of rectangles. In this section we explore such infinite constructions but assume no extra structure (such as $\rceil\,\rceil = \rceil$).

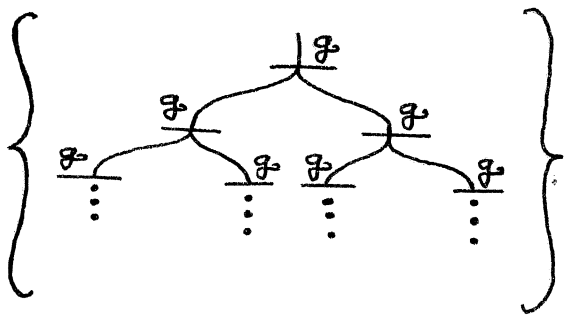
a) $g = \overline{g\,g|} = \overline{\llcorner\!\lrcorner|}$

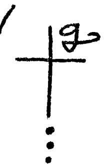
$\Rightarrow \quad = \overline{\overline{g\,g|}\;\overline{g\,g|}\,|}$ (since $g = \overline{g\,g|}$)

$= \overline{\overline{\overline{g\,g|}\;\overline{g\,g|}\,|}\;\overline{\overline{g\,g|}\;\overline{g\,g|}\,|}\,|}$

Whence

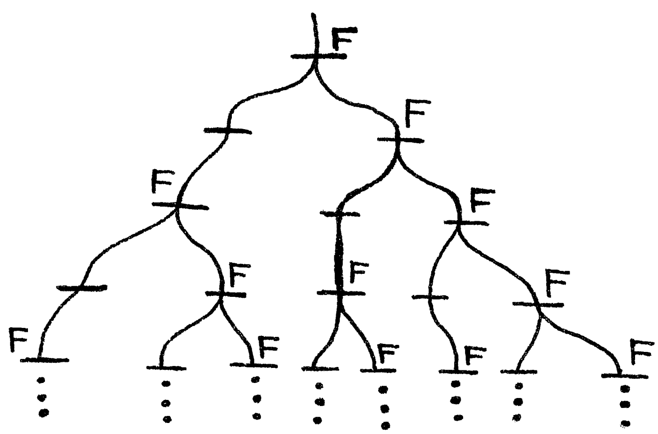
$g =$  (ad infinitum)

or 

(

denotes everything <u>below</u> it in the tree.)

b) $F = \overline{\overline{F|}F|} = \overline{\llcorner\!\lrcorner|}$



1
2
3
5
8
⋮

You might call this one the <u>Fibonacci</u> <u>Fractal</u>!

If f_n denotes the number of divisions of F at depth n, then we see at once from $F = \overline{\overline{F}|\,F}|$ that $f_n = f_{n-2} + f_{n-1}$. Let the <u>growth rate</u> $\mu = \mu(F)$ be defined (when defined) by

$$\mu = \lim_{n\to\infty} \left(\frac{f_n}{f_{n-1}}\right).$$

Thus here $(f_n/f_{n-1}) = \left(\frac{f_{n-2}}{f_{n-1}}\right) + 1 = 1 + 1/\left(\frac{f_{n-1}}{f_{n-2}}\right)$

$$\Rightarrow \boxed{\mu = 1 + 1/\mu}$$

So $\mu = 1 + \cfrac{1}{1 + \cfrac{1}{1 + \dots}} = \frac{1+\sqrt{5}}{2}.$

VII) <u>Non-Standard Algebras and Logics</u>

i : ...01010101...

j: ...10101010...

$\overline{i}| = i$, $\overline{j}| = j$, $ij = \overline{\ \ }|$ (See V).

In general, let B = any boolean algebra. Let $\hat{B} = \{[a,b] \mid a,b \in B\}$

Define $[a,b][c,d] = [ac,bd]$

$[a,b] + [c,d] = [a+c,bd]$

but $\overline{[a,b]}| = [b',a']$. (Invert and shift.)

Let $1 = [1,1]$, $0 = [0,0]$

$i = [0,1]$, $j = [1,0]$.

Think of $[a,b]$ as representing ...abababab... (or ababab... if you like the order made explicit.). $\hat{B}$ is an example of a

De-Morgan Algebra. Note that since $\dot{\imath} = \overline{\dot{\imath}}$ and $\dot{\jmath} = \overline{\dot{\jmath}}$, the algebra is not boolean.

$$\widehat{\{0,1\}} = \{0,1,\dot{\imath},\dot{\jmath}\}$$

The subalgebra $\{0,1,\dot{\imath}\}$ is also a D-M Algebra and corresponds to some people's notions of a 3-valued logic.

VIII) Complex Numbers

Transpose the ideas of section VII over the real numbers R, and we get a new construction for the complex numbers:

$$\hat{R} = \{[a,b] \mid a,b \in R\}$$

$$\left\{\begin{array}{l} [a,b][c,d] = [ac,bd] \\ [a,b] + [c,d] = [a+c,b+d] \\ \overline{[a,b]} = [b,a] \quad \text{(conjugation)} \\ 0 = [0,0],\ 1 = [1,1] \\ \kappa = [+1,-1] \end{array}\right\} \begin{array}{l} k^2 = [+1,-1][+1,-1] \\ \quad = [+1,+1] \\ \therefore k^2 = 1 \end{array}$$

So far $\hat{R} = \{A+\kappa B \mid A,B \in R\}$ and $\kappa^2 = +1$. This is not the complex numbers. We want $\sqrt{-1}$ to correspond to κ!

$$[X^2 = -1 \Rightarrow X = \frac{-1}{X} \Rightarrow \text{If } X = 1 \Rightarrow X = -1 \Rightarrow X = 1 \Rightarrow \ldots]$$

Exercise: Show that if $\alpha,\beta \in \hat{R}$ and we define

$\alpha*\beta = \frac{1}{2}(\alpha\beta + \bar{\alpha}\beta + \alpha\bar{\beta} - \bar{\alpha}\bar{\beta})$ then $*$ defines a mult on $\hat{R}$ s.t.

i) $*$ is commutative, associative and distributes over +.

ii) $\kappa*\kappa = -1$.

This reconstructs the complex numbers!

Incidently, $\hat{R}$ with the first multiplication is useful in <u>special relativity</u>.

$$Z = X + kY$$

$$\Rightarrow Z\bar{Z} = (X+kY)(X-kY)$$

$$\boxed{Z\bar{Z} = X^2 - Y^2}$$

This gives the space-time metric when (speed of light) = 1, X = space coord, Y = time coordinate, for the <u>space-time plane</u> (Minkowski plane, $\mathfrak{m}$)

$$k(X+kY) = Y + kX$$

k = orthogonality operator for $\mathfrak{m}$.

Null lines: $Z\bar{Z} = 0$

$\Leftrightarrow$

$X^2 - Y^2 = 0$

$\Leftrightarrow$

$X = Y$: $X(1+k)$

$X = -Y$: $X(1-k)$

$\therefore$ Let $p = \frac{1+k}{2}$

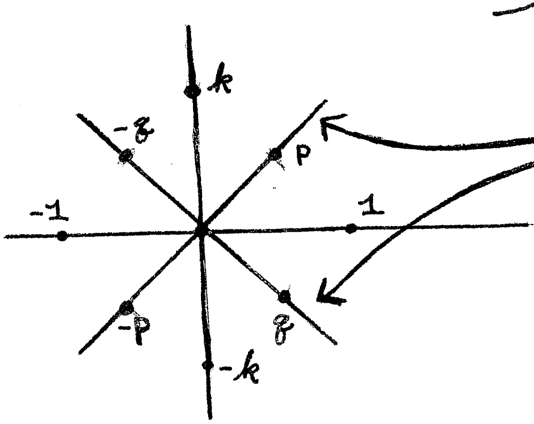
$q = \frac{1-k}{2}$

$p = [1,0]$

$q = [0,1]$

$$\left\{ \begin{array}{l} p^2 = p \\ q^2 = q \\ p+q = 1 \\ pq = 0 \end{array} \right\}$$

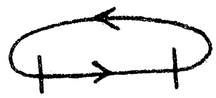
$\boxed{p-q = k}$

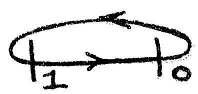


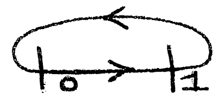
Light cone thru the origin is spanned by $\{p,q\}$ forming a mini-boolean algebra!

IX) <u>Finite (Deterministic) Automata</u>

Return to <u>inverter nets</u> (see section I):

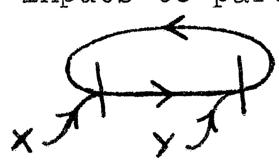
 has two

<u>balanced states</u>: 

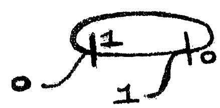
and 

Two inverters placed "back-to-back" form an elementary computer memory element (or flip-flop).

We can include inputs to part a):

$\mathcal{M}$ = 

and analyze:

<u>e.g.</u> X = 0, Y = 1 

Let X $\rightsquigarrow$ 1. Then $\mathcal{M} \rightsquigarrow$ 

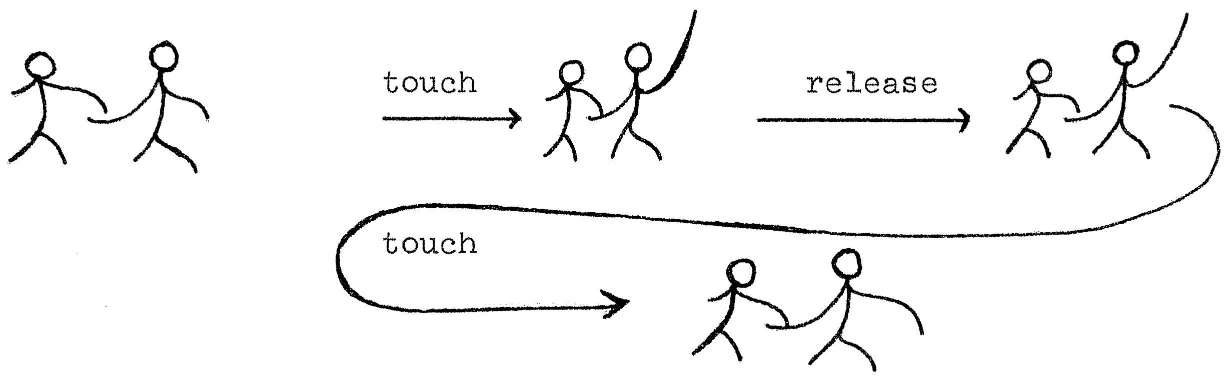
. $\mathcal{M}$ <u>remembers</u> the state (1,0).

It is easy to formalize these notions and present a simple mathematical model for inverter nets. Transitions can be followed by marking the net with (e.g.) Go stones. Imbalances are sequentially re-set (possibly creating new imbalances) until the net reaches a new stable state.

There are relationships between this sort of net-analysis and De-Morgan Algebras (see (VIII)).

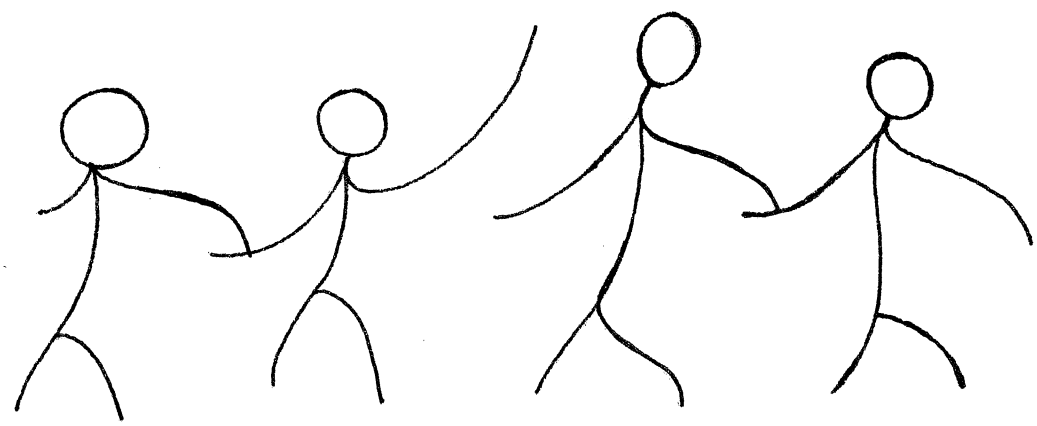
X) Digital Chorus Line

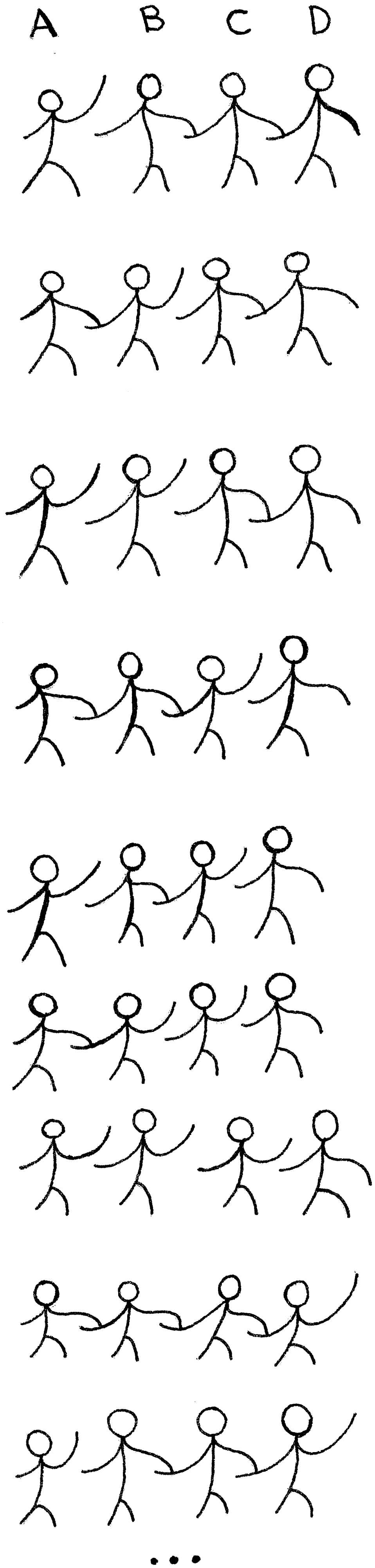
Human counting circuit. (Have students act this out.)



Each person becomes a binary frequency divider alá above indications.

Here is what might happen for four protagonists. (The leftmost person drives the assembly):





In this transition, A touches B causing B to touch C, and C raises his/her arm.

Here A touches B, making B touch C, making C touch D and D raises arm.

<u>Finis</u>