SN There should be a greater attempt in this elementary T.M. work, to stick as close as possible to intuitive methods of learning. T.M. should be simply an attempt to explicate one's intuitive concepts.

E.G. In = we would like T.M. to realize that I and 0 are somehow related to =, but there is no reason for T.M. to pick such a remote word as "cause" when I and 0 are sufficient.

When = is given, again, we would like T.M. to realize I and 0 are assoc. with =.

While this is not too improbable, tetragrams are still probably easier (intuitively).

When ( = B) comes around, T.M. should be ready to realize that the isolated signs =, =, and isolated operations occurring near them — or even in R. some example.

At this point the fall. epist. rule comes up: If something happens that was unforeseen, then look for the things near the unforeseen event that are "unusual." They are good causal guesses for the "causes" of the unforeseen events. A good def. of "unusual" has been found.

before: essentially, an event is "unusual" if it doesn't conform to R. ass. That most examples have conformed to up to R. present time.

It might be well to formulate this in a form for use with the elementary Math T.M. Suppose one has a bunch of ass. that are very U. in prediction — that one's score is very low. Then one gets a few new examples in which predictions are
However, it may be said that if we have a sufficiently long or = problem, it will take less time for T.M. to learn tetragrams, than to carry on the "contagion process" of learning trigons like |

6 Say T.M. has learned tetragrams. (The formation of trigons gone thru on x61 to x63 is mainly for illustration of methods of developing tips in inversion, and the use of neighboring trigons.)

Next try --.(E V) (E +)

+x 0 0 0 + 0 0 0 + 0 0 0 + etc.

...x 0 0 0 + 0 0 0 + 0 0 0 + etc.

The monograms now become of some U, since it's occurs more frequently than O's.

If O's are put in the bottom line only, then becomes of hy U, for this particular problem = but it is of not much U, in -- so on the average, does much good.

The tetragrams

+ 0 0 0 0 0 0 0 are of hy U, with 100% correctness. They do not, however, fit all cases.

The tetragrams + 0 0 0 etc. are of usual lower U. However, they can be salvaged by adding As space, to obtain pentagrams:

+ 0 0 0 0 0 0 etc.
of these, only
\[ V, V', \{ V \} \] are of any use, so we retain them. They produce
\[ \varphi = \varphi \square \] etc.

Remarks: This fact strengthens the a priori of M. X D, as well as its U.

Similarly, we try neighboring strata to \[ \{ V \} \] et al. and obtain
\[ \varphi \square \] etc.

We get \[ \varphi \square \] and \[ \varphi \square \], etc.

by repeating similar operations each time.

[5N] At this point, we would like T.M. to learn that if we can do from it is desirable to go from
\[ \varphi \square \] to \[ \varphi \square \] to \[ \varphi \square \],

it should be much more immediately desirable to try \[ \varphi \square \] and \[ \varphi \square \].

This seems to make T.M. expedient,
T.M.'s ability to use sets of sets of sets.

Whether T.M. uses tetragons or trigons to predict \( \varphi \), will depend on the relative a priori attentuations assigned to the various operations used in forming them.
\[ \text{str.} \times \left[ \left( \sim, \sim \right) \times \left( \text{monogram set, cart. product digm set} \right) \right] \]

\[ S_1 \times \left[ M, \times D_1 \right] \]

\( S_1 \) is of hy aprip. as an initial str. that T.M. starts with.

Th. trigm set \( M, \times D_1 \) was obtained in R. follow way!

\[ \text{was mult. by R. cart product of} \]

\( \left( \begin{array}{cccc}
\text{are monograms, and} \\
\text{are always of hy U.} \\
\end{array} \right) \]

Th. result was a set of 128 trigrams, of these, only 32 occur; only 24 contain \( U \) of these 24, only 8 are of hy U. They are except for extraneous ambiguous trigrams.

\[ \text{Note: } 1 \begin{array}{c}
\end{array} \times \left( \begin{array}{c}
\sim, \sim \\
\end{array} \right) \rightarrow \left( \begin{array}{c}
\sim, \sim \\
\end{array} \right) \]

In general, if in multiplying a str. times an ntpl, th. result is ambiguous, then all of m. ambiguous elements a resultants should be retained.

Since \( S_1 \times \left( M, \times D_1 \right) \) is a monst of hy U, and \( S_1 \) is of hy aprip, then \( M, \times D_1 \) is of hy aprip.

We try strs. near \( S_1 \), and mult. them by \( M, \times D_1 \), to obtain trial trigrams.

Some strs. "near" \( 1 \begin{array}{c}
\end{array} \times S_1 \)

\[ 1 \begin{array}{c}
\end{array}, \begin{array}{c}
2 \end{array}, \begin{array}{c}
2 \end{array}, \begin{array}{c}
1 \end{array}, \begin{array}{c}
1 \end{array}, \begin{array}{c}
2 \end{array}, \begin{array}{c}
1 \end{array}, \begin{array}{c}
2 \end{array} \]
The in ability is due to the fact that e.g. \( \textcircled{1} \) and \( \textcircled{0} \) cannot be, in this method, members of the same categorization group. In the present case, it would be nice to have \( \textcircled{1} \) and \( \textcircled{0} \) members of the same category — since this would give something closer to the idea of “equality” but this is not essential yet.

It should be looked into soon however. We might make \( \square \) part of the specification of a str. — Then

\[
\begin{bmatrix}
\textcircled{1} & \textcircled{0} & \textcircled{1}
\end{bmatrix}
\]

\( \text{str. monogram set.} \)
\( \text{\# tuple set.} \)

with associated problems. This makes all digits useless. Trigrams are tried,

\[
\begin{bmatrix}
\textcircled{1} & \textcircled{0} & \textcircled{1}
\end{bmatrix}
\]

etc. are \( \square \).

However, most cases are not predictable by compact Trigrams.

b) Tetragrams work in every case:

\[
\begin{bmatrix}
1 & 0 & 0 & 1
\end{bmatrix}
\]

etc.

b) It is possible to develop trigrams like

\[
\begin{bmatrix}
\square & \textcircled{1}
\end{bmatrix}
\]

etc. in the following way:

This method involves more operations, but the result seems intuitively more desirable.

Th. mgm set

\[
\begin{bmatrix}
\square & \textcircled{0}
\end{bmatrix}
\]

et. al. can be factored into
from α58.40:

More detailed descriptions of T.M.'s response to a *typi. seq.* up to "+" with carry line.

1. T.M. is first fed a set of examples like

```
1 0 0 1  
1 0 0 1  
0 1 0 1  
1 1 1 0  
```

The problems are of the form

```
1 1 0 1  
1 0 0 1  
```

2. T.M. starts off by making digit freq. counts and using them for prediction.

```
Input array is 10 x 10

\frac{6}{10} \quad \text{is freq. of } 0
\frac{4}{10} \quad \text{is freq. of } 1
```

Monograms are denoted by

□. Frequencies determine a 4-component vector.

3. Predictions are very poor, so T.M. tries digrams next. Some digrams:

```
1 □, □, □, □, □, □, 0 □, □, □, □, etc.
```

Each digram gets a prediction vector assigned to its □, thru freq. counts.

Only the digrams □, □, □, □, □ are of any use in prediction. Since they have a prediction efficiency of 100%, no other digrams or trigrams are introduced, until many problems are run.

```
[5N] The use of □ or □, etc., of monograms or digrams, gives T.M. a serious inability, but we will let this ride for awhile. α61
```
Monte Carlo searches: from $X \times 974$ and $x_{46}$ also (previous) work.

On $x_{45}$ we seem to have worked cases 3 and 4 incorrectly. 
Case 3 is worked exactly and correctly on the bottom of $X_{972}$ and top of $X_{975}$. "n trials" is correct.

For 4) the method of the top of $X_{973}$ seems right, but it conflicts with the corrected method of $x_{45}$ for $x_{45}$: If the $i$th element in $v_{45}$ is correct, then $E_k$ expected no. of trials is

\[ 1 \cdot p_i + 2(1-p_i) p_i + 3(1-p_i)^2 p_i \ldots \]

\[ = p_i \times (1-p_i)^{x-1} = p_i \left( \frac{1}{(1-p_i)^x} \right) = \frac{1}{p_i} \]

Since the $i$th element is correct $p_i$ of the time, the expected no. of trials for all time is

\[ \sum_{i=1}^n \frac{1}{p_i} = n. \]

On the other hand: from $X_{972.02}$

The prob. of getting the right choice in the first trial is $p_i^2$.

The prob. of getting it on the second trial is

\[ (1-p_i^2) \times p_i \]

This is the error. The prob. distrib. for the second choice is not

\[ \frac{f_{x_{45}, \text{sum}}(x_{45}, \ldots, x_{45})}{x_{45}} \]

The first, since knowing we failed the first time makes it more likely that $x_{45}$ "true" case is more toward the lower $p_i$'s.
4) Shannon suggested that they be given a chance to pick their own problem for this T.M. to solve in a way different from the usual problem, and to do a hand simulation of it.

List imp. abs. that T.M. ought to have:

1. carry some problem soln.
2. set of set of sets ... etc.
3. to extrapolate = Th. stru. 1 2 3 4 ... 2, out into 4.
4. direction, an arb. distance.
5. Will T.M. think of trying commutativity on various operators? i.e. in 1+2=3 to give 2+1=3 etc. prior than 1+3=2.
6. Teaching T.M. to fill in < 1 0> ; G. commission.

[SN] On B.G.: Th. soln. That consists of taking the "Best" categorization group that an event belongs to, isn't too bad. In fact it is as general as soln. to a problem as any. Th. gp. that is Th. Boolean product of all R.

[SP] an event belongs to, will, in general, be et by apr. 0 and be the "best" gp. available. Th. trouble is, the evaluation of R. prediction params. of a gp. as a function of R.

[FP] params. of R. etam pps., tups, and strts. from which it was formed, is as difficult as the B.G. problem!

However, this latter would have to be done anyway, regardless of what sort of B.G. soln. was used.

Present approach: Just work on R. devising of new gp.

from old problem, with a little attention to the scheme expected amount of attenuation of apr. with the no. of operations performed.

A running account of T.M.'s battle with a typ. sequ.

is on X939.27 ff. Comments up to X957.22 where it is decided that I ought to be able to do up to +" with carry line, and I should "consolidate 8 ins".

A more detailed description of Th. Typ. sequ. up to +" will follow - on 260.
in which it occurs. If one has an infinitely large discrete universe, or a continuous one, there are, indeed, an \( \infty \) of possible states of the universe in which it occurs.

Also, I think that in the case of learning

\[
\begin{array}{cccccccccccccccccccccccccc}
1 & 1 & 1 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 & 1 & 0 & 0
\end{array}
\]

there is essentially, \( \mathcal{P} \) poss for an \( \infty \) of special squares between \( \mathcal{P} \) and \( \mathcal{R} \).

\text{**Tuesday, July 10, 1956**:} day of talk on T.M. to Dartmouth group:
Minsky, McCarthy, McCulloch, Shannon, attended - also partly Bill Schatz.

**Criticisms:**

1) They would like to do a hand simulation of this (Shannon in particular).
2) McCarthy expressed worry about T.M. length of some of the search processes that must be involved.
3) Minsky suggested ad-hoc soln. of T.M. for and seems quite happy to add all sorts of ad-hoc mechanisms; if they seem very useful. I am afraid that
   a) he is overly impressed by \( \mathcal{P} \) necessity of built-in mechanisms.
   b) He isn't aware that ad-hoc/specific since in using a T.M. one misses one more chance to pick up the clue of a general, very general mechanism, that may be the difference between "creative" and not.

The problem about setting up "classes of classes of classes" etc. in T.M. can, I think, be solved by giving NUMAN detailed instructions, in English, then coding them for T.M.

A class may be defined by a set of instructions telling one if an object is in its class or not - also, it may contain quick-try rejection methods that usually work.

1. List of class members
2. How class was constructed from other classes
3. How to test any object
We also have objects (e.g. permutations) which operate on sets to produce sets of sets. If $S_1, S_2, ..., S_n$ is a set of sets, then $G_z \equiv S_i \cdot N_i \ [i = 1/n]$ is a set of sets.

I think there are other useful sets of sets.

3) We probably can use higher order objects like sets of sets of sets.

The concept of "Number": We can use Russell's definition: $\exists \ z \equiv \ P_{\alpha}$. Set of all pairs of objects. We can have the "between-ness" relation: e.g. $<P_{\beta}, \text{set of all triples}>$ is "between" $P_4 <\text{set of all pairs}>$ and $<\text{the set of all quadruples}>

We must design T.M. so that this does turn out to be a "useful" concept, and that it is made likely that T.M. would "discover" it.

M. suggests that T.M. word "integer" might be difficult to define. This is a set of sets, but it has infinitely many members, which is a difficulty. However, many other of T.M.'s sets might be put of as having no members. E.g. by the diagram of 01, may be part of as including all situations.
How to get concepts like "number".

Consider that T.M. generally finds that 1 , 180 , 1980 , 18880 are all useful. Would he be able to extrapolate that 18880 was probably useful?

To start from R. beginning: relationships

1) we have sets of phenomena. These sets have the property that they group phenomena in ways that are useful for prediction. I.e. so that the occurrence of one element of a set can be used as statistical data for the occurrence of other members of the set.

2) members of the same set are "close" to one another in some way that is named by the set name.

2) 2 sets are "close" if the was empirical usefulness of one suggests that the other is probably useful. "Closeness" suggests sets of sets.

I.E. closeness suggests "neighborhoods". There is a closeness relation on a set of pts. is partly defined by a set of point sets, such that all the members of each point set are close together. Anyway, in general, the concept of set of sets is useful.

The factorization concept also gives sets of sets. Say a group is generated by str. $S_1 \cdot$ mult. by upst. N.

$G_1 = S_1 \cdot N_1$

If $S_2$ is close to $S$, then $G_2 = S_2 \cdot N_1$ is close to $G_1$.
Methodological Notes: It appears that sub-goals (and sec. recent) are only imp. in T.M. problems involving a search. E.g., in a well-defined problem of one kind — where one must invert some definite operator. There are many imp. T.M. problems that do not involve searching (this isn't absolutely certain). For this reason, it appears worth while to solve the non-search problems first, since they are simpler. A search problem involves all the techniques of a non-search problem, plus the extra problem of devising sub-goals. I think that once the non-search problem is solved, the search problem will be easy to do, using the techniques of abs. construction that were devised on the non-search problem.

This devising of abstractions sometimes involves searching. E.g., the inversion of certain X-forms is very often useful.

For preliminary work, however, we may simply assume that we can carry thru the search process successfully — say by means of a very large computer.
One also feels that if, in the past, this fact should be made much of.

A very good reason why I would want to be of by U: In R.W. language, it is the events that do occur, that get words assigned to them, not which of zero frequency.

A temporary way to solve this problem: (temporary in the sense of enabling me to continue with the review on T.M.). Th. "best" will have assoc. with it, e, U, and a vector, whose components give the prob. of various digits in this augst also has other prediction properties. Th. "Rough and dirty" R.G. soln. That will be used at first, is that we take th. augst of kyest U and a prop. (meaning? - since proby is a vector) or, more exactly, th. "best" augst, and use its proby vector as & soln. In the case of several augsts all about of goodness, average their vectors in some way (or correlate them?) - Anyway, if these "best" vectors differ much, give th. resultant a low variability.

It is very probable that discussion of what how to make WAMMA freq. counts in Math T.M. has looked into this point. X 858A has something to say about this, but not too clearly.

This new approach of suggesting that one take the "best" augst, and use its vector, isn't so bad. If several augsts should have their vectors combined to give the final result, perhaps augsts themselves could be combined in an optimal way to yield an even more optimal augst.

The trouble with this, I think, type representation, is that could not be members of this same set. This must not be imp. for very low level problems, but it is certainly imp. for complex problems.
In writing a review of work on Math T.M., Th. foll. prob. has come up:

We are trying to predict the square in
\[
\begin{pmatrix}
1 & 1 & 0 & 1 \\
1 & 1 & 0 & 1
\end{pmatrix}
\]

We want the rel. probability of
\[
\begin{pmatrix}
1 & 1 & 0 & 1 \\
1 & 0 & 1 & 1
\end{pmatrix}
\]

There are 24 possible binary digrams containing 0 assuming that the 3 possible completions could be 0, 1, or 0. We would think that the digrams of greatest U would be 0 and 1. However, if the digram 0 is used along with any other digram like 1,0, of non-zero frequency, prediction is made as good, or almost as good. 0 gives as good prediction as 0. For n examples, the freq. of occurrence of

\[
\begin{array}{c}
0 \text{ is zero; } 0 \text{ is } \frac{1}{2}; 0 \text{ null is } \frac{1}{4} \text{ (in fact } 0 \text{ null happens if and only if } 0 \text{ )}
\end{array}
\]

For this reason, 0, 0, 0, 1, and 1 get by U:

0 etc. get about the same U as most other digrams.

Intuitively, it would seem that 0, etc. ought to get by U, and that 1 etc. ought to get about zero U, yet 11 gets as much U as 1.

Also, Th. soln. to E.S. that is to be applied, isn’t at all clear. There is a kind of soln. on X 898.24, but it is still in too complex a form to understand what it with mean (even approximately) in more specific cases.

Also, it is felt that if one has statistical info. on 2 mutually incompatible n.m. types, like 1,0 and 1,0, one should make much of this info. At the present time, it is not too clear as to how this should be done.
T. M. This is because we have, because it appears that we can go extremely far, without having to alter the basic assumptions made very much—i.e., steps and mortalities. Certainly it seems that the usage line is possible, and very probably most hyperoperations with little basic modification.

[SAJ] At the present time, creation of new abstract entities by inversion, is being looked upon as a rather not very special case of the creation of a set by multiplication of a set of steps and mortalities. E.g., A g.p. \( \alpha \) is found to be of the form \( U \). \( \alpha \) is then found to be factorable into some structure \( S \), and the mortality, \( M \), so \( \alpha = S M \).

If \( U \) is of the form \( U \), then \( S \) becomes of the form \( U \) too of the form \( U \).

Actually, there is a rather serious search process involved. One can, mathematically, state that for any \( \alpha \) of the form \( U \), there for any \( \beta \) of the form \( U \), \( \alpha = S M = \alpha \), the a priori of \( S M \). However, often, one will not find this \( S M \) pair, since the computer available, with its very limited in capacity. The finding of such an \( S M \) pair will be a real discovery, and will not always occur, even when such a pair exists. The amount of time that one wants to spend looking for such pairs, will be sharply limited, but ingenuous for any \( \alpha \) of the form \( U \), many \( S M \) 's will be tried, to see if they yield something close to a g.p. of the form \( U \).

In some cases, the division of \( \alpha \) by \( M \) can be carried out to yield \( S \).
Some config. eval. methods will assign variances along with values. As $i$ increases, $1/$ variance decreases.

Actually, the variance of an evaluation will depend upon how many moves into the future one's opponent looks.

If one has these variances, one can know just how many moves to pursue each line of play. There is, however, the Q of whether at any time, it is better to spend time evaluating a move or to proceed with further analysis into the future. One may employ special quickly-applied evaluators, for preliminary determination of whether to use a hyper powered evaluation scheme, or simply continue tracing into the future.

Is it possible to make use of some of the work done in deciding on the previous move, to reduce the search process for the present move?

[SN] Math T.M. starts on T.M. X 729 with discussion of several kinds of probe to work. X730 has list of some fairly good probes. If they were solved by T.M., we would feel that it was really learning things. Also, it seems not too unreasonable that after learning all that math, T.M. should be able to learn to understand Q's posed in a simple language. Perhaps, eventually, English.

To understand English, however, it may be necessary to get T.M. to be able to predict analog events in R.W. Then translate them into English, and use the analog events (concepts) to extrapolate English - or to "understand it".

A somewhat more detailed "learning sequence" for the initial stages is on X774. From X774 to X780 is a discussion of whether or not Math T.M. is of much use - i.e., is it likely to demonstrate "creativity" as well as understand English?

As a final remark: A criterion of whether Math T.M. is really being "creative." If, after a long while, T.M. has been producing what would normally be regarded as "creative" results, and has had no new ad-hoc abstraction method inserted into its program for a very long time, then we will say that Math T.M. is really being creative.

At the present time, there seems to be reason to believe that we can expect creativity from
Just what would an efficient search procedure be in these games? Would there be much difference in chess or checkers?

2 random-even methods can always be compared: Simply take a large no. of past configurations that are of reasonable interest at p.r. and apply both methods. Then use a 3rd method (or one of 9a. 2) that has been "soaked up" by, say, 5 move exhaustive search, to decide which configurations are "really" the best. Compare the orher cgs of 2a. 2 methods to be compared, with the "true" or largest.

For optimum searching, at each point in the search, each move under evaluation will have a certain expected G and a variance about that G. Decisions to go ahead with a search line or drop it, can be then made on this basis. For example, consider game in which each player only has a possible move each turn, etc.

Black and white tokens. A dimensional board. Each player places token on North or South end of pile. A player wins if he gets some of his tokens to some occupy all the points of coordinates ni + 2 with \[ k = 1 \]

n, a = any integers, i may be 0, h must be \( > 0 \) if \( n \) must be \( > 2 \). for \( n = 2 \), this may be too easy.

This game may be trivial.

To figure out lines of play, one works backwards:

- white
- Black
- W
- B

First one evaluates all moves in the list row, in pairs. One discards the worst, then compares at next level, etc. Arrows show the choices.

Note: if 2 or more terminal pts. are really good, we will have some ambiguities.

Each position has a sat. Say one can linearly order all configurations. Then one can easily make all choices.

A config. has a sat. of values \( a, a_2 \ldots \)

\[ \ldots a_0 \]

\( a_i \) is the optimum value obtained after \( a \) play exhaustive search \( i \) moves into the future. \( a_0 \) is the true value of \( a \) move — usually is \( > 0 \) is \( < 0 \) or we may have a draw. Usually as \( a \) \( \rightarrow \), \( a_0 \) will not be even approx. monotonic — except if \( a_i \) becomes very low or very high — in which case an imp. move is won or lost, or the game needs its end.
and the method that uses the carry line.

The above method of "carry line" itself may not be so good. It seems that in addition problems, one does not carry's on one's head, and it may be expedient to have T.M. learn to do this.

Also, there are probably many processes in which

\[ A \rightarrow B \rightarrow C \rightarrow D \rightarrow E \rightarrow F \]

and it is only F that one cares about. To put it another way, one may go from "form e" to "form d". Take a b c d, so that a b c d is of reasonable prob. Then one really only wants a d.

This can be accomplished by a special operation on the trio of pairs: \( f((a, b), (b, c), (c, d)) = (a, d) \).

This is not as general as the set of xfrms:

\[ f(a, b) \rightarrow (c, d, e) \rightarrow (f, g, h) \] and it is only

\[ (f, g, h) \] that one wants. Well - the operation

\[ (f, g, h) \rightarrow (f, h) \] is certainly simple enuf, and is by no means a new xfrm or an ad-hoc one.

---

**SN**

Chess and Checker playing machines:

A fairly good checker player has been constructed by [A. Samuel](9).

It plays out all sequences to 5 moves, then counts men and uses out of "centrality" for tie breaking.

**A.** On evaluating a position:

Any 2 evaluation methods can be compared. Say method A takes t seconds to use, and B takes k t sec. (k > 1)

Suppose that there are m alternative moves on B, average.

Then play pool all positions log x m moves in advance and use method A. Compare win probability with B, when

B is used immediately.

It may, indeed, be true that few eval. methods are as "good" as a very simple method such as...
For example: I was worried about the learning of addition without carry line. Intuitively, the solution is as follows:

\[
\begin{array}{c}
+1101 \\
10101 \\
\hline
11110
\end{array}
\]

we can factor this in to:

\[
\begin{array}{c}
1110 \\
0101 \\
10100
\end{array}
\]

\[
\begin{pmatrix}
1110 \\
0101 \\
10100
\end{pmatrix}
\]

\[
\begin{pmatrix}
1234
\end{pmatrix}
\]

Tom has suggested that to simplify explanation, write Str as a vector, with each vector component giving the cart. record of the displacement of each component relative to the first component.

\[
X = \begin{pmatrix} 0,0 \\ 1,0 \end{pmatrix}, \begin{pmatrix} 0,0 \\ 0,-1 \end{pmatrix}, \begin{pmatrix} 1,0 \\ 0,0 \end{pmatrix}, \begin{pmatrix} 0,0 \\ -1,0 \end{pmatrix}
\]

In the last case, we illustrate the fact that _any_ constant 2 dim. vector can be added to all components of Str and leave Str invariant. The "str" is this part that remains invariant under such an xform.

\[
\begin{align*}
\begin{pmatrix} 1/2 \\ 1 \\ 2 \end{pmatrix} \times \begin{pmatrix} a \\ b \end{pmatrix} &= \begin{pmatrix} a \\ b \end{pmatrix} \\
\begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix} \times \begin{pmatrix} a \\ b \end{pmatrix} &= \begin{pmatrix} b \\ a \end{pmatrix}
\end{align*}
\]

There is still some trouble with the i.p. \( \begin{pmatrix} 0 \\ 1 \end{pmatrix} \) in our intuitive approach that is being outlined above.

To get

\[
\begin{pmatrix} 1010 \\ 10100 \end{pmatrix}
\]

we must

\[
\begin{pmatrix} 1110, 0101, 11110, 10100 \end{pmatrix}
\]

by

\[
\begin{pmatrix} 1 \\ 2 \\ 4 \end{pmatrix}
\]

Similarly, we can get 0101 + 1101 = 10100 with a little extra trouble. In fact this may be better to learn this way since there is some ambiguity.
choices toward bycast Pi.

The function $x^r$ is the "only" function that satisfies

Th. equ. $f(x) f(y) = f(xy)$. [r can be anything]

This means that if the P's are obtained by

Monte Carlo methods and for all P's,

$P_1 = P_1 / P_2$ ... then one can

suitably non-linearize the P_1 / P_2

P_1 and P_2 may, in turn, be obtain as random

variables. Eventually, at some low level, one may

obtain some random Monte Carlo variables

whose probles one can modify by

$p_i = p_i^r$, since the number of such

low-level variables is small—much smaller

than the no. of P's.

---

General Methodological note: In working, say,
on Math T.M., keep operations much closer to intuitive
processes than I have been doing. The definitions
of str's. and entities, and their cart. products,
are things that I feel that I understand intuitively, and so I should be able to do it
more exactly.

---


1) *post. of quasi-permanence (in time)* [but also insane]
2) *post. of separate causal lines* (a thing usually isn't caused by
   everything)
3) " " " special temporal continuity (no essential edit)
4) " " " common causal origin of similar str's
   grouped about a center. [a center in time or space] [may also be involved "post. of explanation"
   "cuff"—i.e., can one develop all rules
   of induction if one uses, from them.]
5) *post. of analogy* (similar to 1), but in both time and space.