July 4, 1956

A FRAMEWORK FOR AN ARTIFICIAL INTELLIGENCE

By Marvin Minsky Paper I

The following is a general description of certain components of a machine which is expected to work at the solutions of problems in a reasonably intelligent manner. Its construction is not imminent -- the system described here is intended only to provide an heuristic framework in which certain general procedures can be examined in some detail. The machine is described in terms of a diagram with several "blocks." None of these blocks are described in anything approaching complete detail; on the contrary, each of them raises its own set of theoretical problems. The value of this approach, I feel, is that such a framework makes it possible to work in the direction of relatively hard solutions to specific problems. The danger is that of being trapped into making unnecessarily elementalistic distinctions in an effort to maintain the boundaries of the boxes. But this may be a smaller risk than that involved in evading separation; _then_ one is liable to find that one has n boxes, n-1 of which contain a few relays or valves, and the last behaves intelligently! I claim, at the least, that no _one_ of my boxes will behave intelligently.

The idea of constructing the framework, and many details concerning the "Characterizer," the "Method box," and the "Clean-up box," originated or were agreed upon in discussions between the author and John McCarthy. Some details of the "evaluator" originated in discussion with Ray Solomonoff. Many ideas came out in meetings of the Artificial Intelligence Group as a body.

For concreteness, I will often talk as though we had a particular interest in building a machine to find proofs for theorems in mathematical logic. Actually, the machine is intended to operate over much broader classes of problems, including the making of empirical perceptions and judgments. The use of the logic-proving example is dictated not only by the convenience of the clarity of the problems involved, but by the

It would seem that this device is suitable mainly for "well defin[illegible]" problems of the [illegible]

EXPLORATION SYSTEMS AND SYNTACTIC PROCESSES

By Marvin Minsky   Paper II

A "syntactic process" is a formal system (with a set of "atomic" or "kernel" sentences and a set of transformations which with the atomic sentences generate other sentences), together with a process which controls the sequence in which new sentences are generated. That is, a process whose past history determines, in a specified way, which transformation is to be next applied to which expression(s). The process itself is not here regarded as "formal;" its behavior is involved with semantic and, frequently, empirical contingencies.

In many artificial intelligence situations the syntactic process may involve in an essential way what may be called an "exploration graph." "Problem solving" seems to usually involve such a structure. Other parts of the process may involve the use of special procedures such as "interpretations" and "examples" which may use special computers or "models." The ideas of "memory," design and performance of "empirical experiments," and computation of "characters" and "abstractions" are likely to be required. And a program for self-improvement will likely be part of any important intelligence.

These various divisions might reflect actual divisions of the machine program into parts, or even separations of hardware, or they might be just a heuristic way of thinking about a relatively interpreted operation. By the use of the term "syntactic" we wish to indicate that the emphasis is on programs in whose operation expressions are used in a manner more or less suggestive of a humanoid language.

Two topics will be examined here. The first topic will be a discussion of the exploration process for problem-solving machines. The second topic is a discussion of possible ways to construct the syntax of a (meta)language so that a formal deductive system can be used, together with a semantic, and a set of empirical verification methods. We will in fact be talking about a (meta)language which in turn can talk about a logic plus a set of interpretations, plus non-logically derived properties of the models on which the interpretations are grounded.

NOTES ON THE GEOMETRY PROBLEM

Paper III

M. L. Minsky

1. The language.

One of the reasons Plane Geometry might be a rewarding domain for artificial intelligence is that there is a good chance that we could find a language that was simultaneously suitable for machine use and human use. A rather small vocabulary is required, and very little grammar outside that of propositional calculus would be needed. A little bit of lower functional calculus and some special grammatical states or "moods" would be all that need be added. The "moods" are sentences which bear on the various parts of the master organization: they assign to a proposition the status of fact, assumed premise, question to be answered, order ("work on this subgoal:___") etc. I think that incorporation in the object language of terms which can describe machine orders will cost little and lead to valuable observations, but I would concede that it might be overambitious on a first study. These terms will appear in some form in the pre-assembled program, in any case.

Nouns: "$\underline{AB}$", "$\angle\underline{ABC}$", "$\underline{P}$", C(P,r) etc. represent line segments, angles, points, circles, etc. Other terms like "square ABCD" can be a priori or later defined.

Relation terms: Equality: $\underline{AB} = \underline{CD}$ (length equality)

$\angle A = \angle B$

Congruence $\cong$

Similarity $\sim$

Parallelism $\parallel$

Perpendicularity $\perp$ etc.

2. Logic.

Most of elementary geometry can be handled through propositional calculus, or at least most of the formal proofs. The formalization of "construction" may be a little tricky.

ON THE MEANING OF "THE READER WILL EASILY VERIFY"

By John McCarthy

Logicians occasionally complain that the proofs given by mathematicians are incomplete almost to the point of non-existence. Mathematicians sometimes retort that a proof that would satisfy the logician would have to be fifty times as long as is customary. It seems desirable to find a way of writing proofs which will be completely rigorous in the sense that they can be algorithmically verified and yet will be as concise or even more so than the proofs customarily given by mathematicians.

If the theory T in which the theorem $\mathcal{T}$ occurs has a decision procedure P the proof can simply say $\mathcal{T}$: by the decision procedure P or more simply $\mathcal{T}$: P. However, the more interesting theories do not have decision procedures. Nevertheless, perhaps any theory whose proofs are made from a finite number of axioms and rules of inference can be divided into a number of subtheories $T_1, \ldots, T_k$ with decision procedures $P_1, \ldots, P_k$ such that any proof can be written:

$$\begin{array}{l} S_1 : P_{i_1} \\ S_2 : P_{i_2} \\ \quad\vdots \\ \mathcal{T} = S_n : P_{i_n} \end{array}$$

where the jth step means that S_i follows from the axioms and rules of inference of the subtheory T_{i_j} and the $S_{1_j}, \ldots,$ S_{j-1} by means of the decision procedure P_{i_j}.

If one admits <u>pseudo-subtheories</u> on which the length of proofs are bounded then it is trivial that the decision prodedures can be found, in fact if the pseudo-theories consist of a single application of a rule of inference, then any proof is of that form.

The interest of the ideas depends on two things:

1. That the decision procedures should be good enough so that they will decide whether a conjecture follows from an