MASSACHUSETTS INSTITUTE OF TECHNOLOGY

PROJECT MAC

Artificial Intelligence
Memo. No. 154.

January 1968.

THE ARTIFICIAL INTELLIGENCE OF HUBERT L. DREYFUS

A Budget of Fallacies

by

Seymour Papert

TABLE OF CONTENTS

(continued)

TABLE OF CONTENTS

Preface.

0.1 The Metaphysics of Ping-Pong.

In December 1965 a paper by Hubert Dreyfus revived the old game of generating curious arguments for and against Artificial Intelligence. Dreyfus hit top form in September 1967 with an explanation in the Review of Metaphysics of the philosophically interesting difficulties encountered in constructing robots. The best of these is that a mechanical arm controlled by a digital computer could not reasonably be expected to move fast enough to play ping-pong.

It is worth pondering the implications of this intrusion of the Review of Metaphysics into control engineering. Some philosophers may not know that it is a routine, technical problem to decide whether any particular mechanical arm could be moved fast and precisely enough to intercept a ball in flight. The reflections of the existential phenomenologists (Heidegger and Merleau-Ponty) cited by Dreyfus simply cannot be relevant. I do believe that Merleau-Ponty's writing bears on the grave social and psychological problems that would come in the wake of robots like those described by Isaac Asimov. Reading Merleau-Ponty might even be a valuable exercise for students of robotics. His insight into the inadequacies of crude behaviorist models of human behavior and perception could have given pause to some naive people who have written in the name of Artificial Intelligence. But his analysis would be relevant to the feasibility of robots only if computers were necessarily constrained to conform to these simple models.

They are not.

The following pages have at most an indirect bearing on how well robots will be able to play ping-pong. A much graver uneasiness led me to borrow time from the real problems of robotics to discuss the comments of metaphysicians.

The perturbing observation is not that Dreyfus imports metaphysics into engineering but that his discussion is irresponsible. His facts are almost always wrong; his insight into programming is so poor that he classifies as impossible programs a beginner could write; and his logical insensitivity allows him to take his inability to imagine how a particular algorithm can be carried out, as reason to believe no algorithm can achieve the desired purpose. Occasional errors of this sort might be enough to cast doubt on Dreyfus' competence, but would be tolerable if other parts of his text were more serious. The length of my essay is dictated by its purpose of showing that his errors are not occasional.

0.2 The Morality of Metaphysics.

Does it matter? I have been told that it is irrelevant to refute defamatory charges against Simon since other people really have made false claims about achievements in Artificial Intelligence. I have been told that it is a waste of time to show that there is nothing but muddle in Dreyfus' explanation of why machines can "play checkers" but cannot "play chess" since attempts to make a computer translate Russian really have encountered difficulties. I have been told that only a pedant would object to the technical nonsense that pervades every paragraph of Dreyfus' papers about Artificial Intelligence since his real purpose is to provide insight into the rich subtlety of human intelligence. I have been told that his argu-

ments must be read as literary conceits with deep "humanist" content.

I think it does matter. I sympathise with "humanists" who fear that technical developments threaten our social structure, our traditional image of ourselves and our cultural values. But there is a vastly greater danger in abandonning the tradition of intellectually responsible and informed inquiry in the futile hope of an easy resolution of these conflicts. The steady encroachment of the computer must be faced. It is cowardice to respond by filling "humanities" departments with "phenomenologists" who assure us that the computer is barred by its finite number of states from encroaching further into the areas of activity they regard as "uniquely human."

Our culture is indeed in a desperately critical condition if its values must be defended by allowing muddled thinking to depose academic integrity.

0.3 Reason and Rationalization.

My discussion is not a debate about Dreyfus' conclusions. It is of no concern to me whether he thinks machines will play ping-pong or does not believe they will play chess. What does affect me is that so many people praise his papers because they like his conclusions, and show no concern for the quality of his arguments. The conclusions are banal: the taxi driver and my maiden Aunt Agatha believe them as firmly and express them as well. The question is whether Dreyfus has contributed more serious reasons for the common dogma. On the contrary: Aunt Agatha honestly expresses her truly human grounds for believing that machines cannot be like her--they do not love and they do not make jokes; Dreyfus wanders in

search of arguments into areas about which he has neither sensitivity nor knowledge, so that his discussion acquires the superficial dullness of all rationalizations that conceal rather than express deeper problems. The essay I should like to have written would probe the problems we all have in integrating man and machine into a coherent system of thought. But a more direct intellectual duty demands that Dreyfus' papers be discussed as they have been written.

The next sections survey some of the issues I shall discuss later on.

0.4 Being-Human vs. Being-a-Computer.

The most insidious of the specters that haunt discussions on Artificial Intelligence is the arbitration of questions about machines by appeal to introspection. Being-human simply does not feel like being-a-computer--or rather, since we have never been computers, how we imagine being-a-computer to feel.

The specter is easily exorcised when it appears explicitly, as when Dreyfus asks

> "How can a determinate process give rise to experienced indeterminacy?" (Phenomenology, p. 39)

One could equally well ask: "how can small neural activity give rise to experienced largeness or blueness or anger?" and so reject neurology as well as Artificial Intelligence. But the more important reply is that the experienced indeterminacy of the human player does not, on any reasonable interpretation of these words, entail anything about whether machines can play ping-pong or chess. It might possibly pose a problem for the neurologist. It is quite irrelevant to the engineer.

Introspection is more persuasive when it is not explicitly mentioned. Hearing about a chess program irresistibly recalls the experience of playing chess, and some strength of mind is needed to prevent the comparison from militating against the plausibility of the program. One must firmly resist the arrogant suggestion that our own impression of how our minds operate tells us how machines must operate to obtain the same end result. It does not even tell us very much about the mechanisms of our brains. The unconscious motive is unconscious and my vision of the clear blue sky contains no sign of the discrete mosaic of my retina.

A related specter is the tacit transposition of questions about the feasibility of mechanical chess champions and secretaries into quite different questions about whether such entities, if feasible, would resemble us in the nature of their "inner experience" or relate to us as people do. Dreyfus very clearly addresses himself primarily to questions in the first class. However, I have noticed, sometimes with great amazement, that the transposition happens easily and frequently in the course of discussion. I can only warn readers of the temptation and ask them to resist it. They will find no comment in the following pages on whether computers experience indeterminacy.

0.5 The Experience of Determinacy.

It is more interesting to discuss the role played in determining people's attitudes to computers by their experience of the machine as a determinate being. Although life would be very different if friends and enemies were not predictably such, we take great delight in the subtle indeterminacy of human experience. The expression of this in symbol or

irony or in the joy of sharing an experience lived on many levels, can easily come to the fore as the most essentially human of human qualities. By stark contrast the computer's "experience" is seen as unambiguously determinate. It sees the world through the explicit "bits" it reads literally from those cards we humans (but not machines) take as symbols of the Anti-Human or of the Great Society.

Taken literally in a technical argument about what computers can do, none of this has the slightest force. Indeed it is based on a pun. The x can be perfectly determinate as a formal symbol and yet have many degrees of indeterminacy as a number. Programs do operate on one object in many different ways. But if we step back from the literal-minded pre-occupation with logic, it is easy to see that these considerations might have great force on a more human level.

0.6 The Simple-Minded Computer.

Weakness in formal thinking operates in two ways to make the idea of

a robot preposterous. Loose use of words like "infinite" permit us to say that our behavior is infinitely variable and our knowledge encompasses an "infinity of facts." Inability to imagine the kind of formalism that could describe certain aspects of human behavior leads us to say that this behavior cannot in principle be encompassed by formal theories. Thus we generate--by what I shall call the superhuman human fallacy--a description of ourselves that is even further removed than we really are from such primitive robots as have already been made. On the other side the popular conception of a program falls just as far short of what is routinely accomplished by mildly sophisticated algorithms. The combination of an exaggeratedly romantic image of himself with an exaggeratedly simplified image of the computer leaves the layman aghast at the suggestion that a robot could take dictation as well as his secretary.[1]

I shall say relatively little about Dreyfus' image of man. Too much verbal argument would be generated. But no argument is possible about the elementary poverty of his image of programming. I shall show this in three ways of which the first is most relevant to people (strangely, there are many) who think he has made insightful comments on current work in Artificial Intelligence, and the third most relevant to mathematicians and others with regard for elementary logical rigor.

(1) By looking at many examples of his comments on actual projects we shall see that he systematically misunderstands their purpose, their methods and their difficulties. The reason is simple. He knows nothing about the technical issues and barely understands the language used. In addition he is sufficiently suggestible to take people as meaning what

---

1. This is another example cited in the Review of Metaphysics.

he thinks they must.

(2) By examining his explicit descriptions of how he thinks programs operate, we shall see that his imagination does not extend beyond considering the most obvious and well-known algorithm for the problem being discussed. A very striking example is his pre-occupation with the time a computer will take to locate an item in a long list. He writes as if the only possible algorithm[1] is exhaustive item-by-item search, and bases general conclusions on his impression that this process necessarily consumes too much time. The example becomes even more striking when we observe that his conclusion covers all conceivable digital computers--not merely purely serial ones.

(3) Most serious of all is that failure of a particular algorithm systematically leads him to think that the problem is unsolvable.

0.6 Dreyfus Accuses.

Dreyfus' attack on Artificial Intelligence contains components of different degrees of intellectual pretension.

The simplest is a set of almost defamatory personal allegations. H. A. Simon and others are accused of making misleading and boastful claims of success while concealing the difficulties encountered in programming machines to play chess and prove theorems. Although it might be in better taste to ignore these abusively written charges I shall discuss them for two reasons. Dreyfus' argument on these very simple question of fact yields insight into his style of thinking; and I have learned from experience that questions about the honesty and reliability of workers in Artificial Intelligence hover in the background of many discussions on the less personal

---

1. Readers who do not understand a few technical terms will miss only a small part of what I shall say. Of course those who look in from the outside might miss the gravity of it.

aspects.

The charges against specific individuals are completely unfounded and the evidence cited for them can be seen as honest only by assuming that Dreyfus systematically misunderstands even elementary papers on the subject he sets out to denounce.[1] Other charges against unnamed "researchers in artificial intelligence" can hardly be answered. Of course many people have said many silly things about Artificial Intelligence. One could say the same of Existentialism and Phenomenology. But the acknowledged leaders in Artificial Intelligence, and particularly H. A. Simon who is singled out by Dreyfus for the fiercest criticism, could more easily be accused of under-estimating in public writing both what they have achieved and the potential developments of robotics. Simon's published papers are models of clarity and thoughtful self-criticism. It really is outrageous to accuse him of even the most indirect degree of cheating. This sense of outrage is further increased by the observation that much of Dreyfus' "penetrating analysis" (as A. Oettinger has called it) is generated by collecting the specific difficulties Simon and Newell report as technical problems for particular programs, and simply declaring them to be absolute obstacles for all possible programs.

0.7 Empirical and A Priori Components of Dreyfus' Theses.

Dreyfus makes three quite separate assertions about Artificial Intelligence:

(1) There is a limit, somewhere, to what can in principle be achieved. The argument for this is very general and does not explicitly refer to experiments with computers. A typical example is: a human can respond

---

1 I demonstrate this in Chapter 3.

to an indefinite number of situations; computers have a finite number of states; therefore, a human will sometimes deal with some situations better than any computer could.

(2) There is a state of stagnation in Artificial Intelligence. The typical pattern is "early success followed by unexpected difficulties."

(3) The boundary to what can be achieved in programming machines to play chess, recognize faces, translate languages, etc. is not very far beyond what has already been achieved. It "borders on self-delusion" to think otherwise.

These three theses constantly interact in Dreyfus thinking and so in my discussion. Readers must try to maintain the distinction in their own minds. The following very general comments are intended as a guide to later sections.

(1) Statements attributing "indefinite" and "infinite" properties to humans are intolerably vague. Attempts to make them precise either dissipate any semblance of truth, or turn the assertion into a logical truism. (Of course a computer has limits: it cannot draw a round square, nor can it be programmed to yield a decision procedure for the predicate calculus. Nor can humans.)

Moreover: knowing that limits of a more significant kind exist somewhere, would have no bearing on research in Artificial Intelligence without a statement of where the limits are. Dreyfus, as one might expect, finds himself in the most serious trouble when he tries to be specific.

(2) The assertion that there is stagnation is not as factual as it might seem. How is progress measured? How much progress is to be counted as refuting Dreyfus' statement? How does one assess the importance of work

on fundamental or specialized technical problems? Dreyfus does not even try to face these questions. He merely asserts pontifically that there is stagnation.

The most astonishing feature of Dreyfus' texts is their bibliography. His references to experiments on Artificial Intelligence are almost entirely confined to early work that has filtered into anthologies of "classical papers." He talks about chess programs as if the latest event was the victory, in 1960, of a child playing against a program that had not been completely debugged.

It is public knowledge that this impression of the computers' weakness was shattered when a machine effortlessly defeated Dreyfus himself even while his paper was in press.[1] The incident is special only in its irony: Dreyfus' idea of Artificial Intelligence falls as far short of reality in all other areas as it does in chess.

There is no stagnation. The crudely empirical criterion of observing performance of machines suffices to demonstrate steady progress. But even if Dreyfus had bothered to find out how well modern programs actually perform, he would have missed a far deeper point, which I shall introduce through an analogy with another branch of engineering. The innovators in Aviation at the beginning of the century worked by building whole airplanes and flying them. The problems of supersonic airliners and atomic aircraft are being solved now by people who could no more construct an airplane than fly themselves.

---

1 The incident occured after publication of a mimeographed version dated December 1965. The printed version published in 1967 contains some major changes but repeats the story of the ten years old child's victory. The story was gleefully quoted by the New Yorker and other popular magazines.

Artificial Intelligence follows this pattern like any other area of Science or Technology. The sign of its maturity is the emergence of specific technical problems. But the amateur observer sees this maturation as "stagnation."

The goal of Artificial Intelligence is to create intelligent automata so the amateur observer judges each scientist by the degree of intelligence he has personally created in a machine. Since Dreyfus assesses projects only by their finished product, and has no means of estimating how much time and effort they might reasonably need, it necessarily follows that he will see increasing stagnation as the pattern of work changes from limited experiments to serious developmental engineering and theoretical study of fundamental problems.

(3) The following chapters might have been less tedious had Dreyfus confined himself to argument on a theoretical level. Unfortunately, he draws important conclusions from his impression of "stagnation" and so forces me to expose his technical incompetence by examining examples with no deep intrinsic interest of their own.

The role of the stagnation thesis in Dreyfus' thinking is to provide support for a more fundamental thesis: that "intelligent activity" can be divided into separate classes, of which some can be programmed, while others simply cannot - neither on existing machines nor an any conceivable digital computer.

Dreyfus offers three kinds of justification for this claim that "intelligence" is "discontinuous": the supposed stagnation of achievement in Artificial Intelligence; the subjective impression that we sometimes

think in discrete steps and sometimes by holistic operations that defy analysis: and, finally, the assertion that some kinds of "activity" cannot in principle, be formalized or computed.

Although these arguments suggest subtle problems, some of which will be discussed later, Dreyfus' use of them is vitiated by a very simple aspect of his style of reasoning. He always poses questions in a binary, non-quantitative form. The question: "can computers play chess, yes or no?" comes close to begging the question by its very formulation. Some computers can play chess to some degree. A sensible assessment of the "limitations," if there are any, of possible chess programs must use a finer classification of computers and of play. There are situations (in elementary chess or advanced checkers), where people seem to use holistic reasoning, and yet computers can be programmed to compete very effectively. How far can this be carried? How can a reasonable answer even be formulated without quantitative or at least ordinal consideration?

Dreyfus' refusal to take quantitative matters into account appears even more simply in his failure to pay attention to the size of computers. Suppose it were true that failures in Artificial Intelligence did show that the tasks attempted were beyond the capacity of the computers used. One could still ask whether a computer with a thousand or a million times as much capacity could do better. Yet Dreyfus is perfectly willing to deduce general conclusions about the limitations of all digital computers from the performance of the puny machines of the first decade of computer technology.

The extraordinary aspect is not that he generalizes; the purpose of science is to do that. But he never even considers questions of computational

capacity. His theory of the process of computation takes account of no number between "twenty" and "enormous"!

1.0 Machines.

Almost everyone, whether he has seen a program or not, has strong convictions about what computers will never do. Ironically, the arguments used to show that they cannot emulate human intelligence follow, with machine-like regularity, a small number of standard patterns. Superficially, the effusions of my Aunt Agatha on this subject sound very different from those produced by sophisticated philosophers, biologists, or computer specialists. Intellectuals have the advantage of a technical vocabulary for the discussion of human thought and of machines. But, instead of helping them see the real problems posed by the project of creating intelligent automata, their greater sophistication often leads them all the more securely into the same misconceptions.

A good example is H. L. Dreyfus, associate professor of philosophy at M.I.T., who has devoted himself to writing articles complaining about computers, and warning against the pretentions of computer scientists. The resonance he has obtained in several quarters indicates that his difficulties must be sufficiently common to be worth examining thoroughly once and for all in public.[1]

---

1. I shall refer to the following articles and quote sufficiently to be intelligible to readers who have not studied them.

   Alchemy and Artificial Intelligence, RAND Paper P3244, 1965.

   "Phenomenology and Artificial Intelligence," Phenomenology in America, J. M. Edie, ed., 1967.

   "Why Computers Must Have Bodies in Order to Be Intelligent," Rev. of Metaphysics, September 1967.

1.1 What Is Dreyfus' Problem?

Dreyfus has metaphysical reasons to be worried about the potential capacities of computing machines. His entry into the debate about Artificial Intelligence is explained as follows [1]

> "Phenomenologists have thus far held aloof from these controversies, probably because they, like the parties involved in the debate, credulously assume that highly intelligent artifacts have already been developed. Indeed, if such artifacts exist or are about to be built, they are evidence for the truth of traditional empiricism in psychology and some sort of logical atomism in metaphysics. Such machines would certainly be acutely embarrassing to phenomenologists and existentialists." (Phenomenology, p. 32)

Dreyfus can easily reassure existentialists that no highly intelligent artifacts actually exist. But this gives little comfort if they are likely to be constructed. His conceptual difficulties arise when he tries to extend his assurance to the future.

There was a time when one could at least squarely state a difference between men and machines in terms of concepts like "the immaterial soul" and "the faculty of reason." The intellectual mood of our times does not allow this. We are too used to thinking of the material brain as a machine that might be extraordinary but needs no ghost to drive it, and of intelligence as a point in a spectrum that runs from the behavior of simple animals through the groping thought of the child to the subtle thinking of the intellectual adult and, in another dimension, from the tool-making of cavemen through the first beginnings of civilized culture to our own achievements.

An intellectually serious defense of any thesis of the form "machines cannot be intelligent" must, if it is to be more than an analytic matter of language, face tough conceptual problems on the level of definitions

---

1. Readers are not expected to know about phenomenology or metaphysics. Those who do might observe that each sentence of the following statement is questionable.

before it can even ask to be considered for judgment on its truth. For almost everyone qualified to hold a view will say: "some can and some cannot," and ask: what machines cannot have what kind of intelligence?

Dreyfus accepts this frame of discussion:

> "It is necessary first to say a word about computers. I am not trying to argue in this paper that no computer could produce intelligent behavior. It seems obvious to common sense that at least one information processing device, viz. the brain, does produce such behavior. What I am trying to argue here is that such behavior cannot be exhibited by a certain kind of computer: the digital computer, which is the only high-speed, all-purpose,[1] information processing device that we know how to design or even conceive at present, and therefore the device on which all work in artificial intelligence has been and must be done." (Phenomenology, p. 36)

He reasserts his assurance to existentialists in the form:

> " ... there are four distinct types of intelligent activity, only two of which do not presuppose these human forms of information processing and can therefore be programmed. Significant developments in artificial intelligence in the remaining two areas must await computers of an entirely different sort, of which the only existing prototype is the little-understood human brain." (Alchemy, p. iii)

Everything depends on how well he can maintain sharp distinctions between his two classes of computers and his four classes of intelligent activity.

To show that he fails could be quickly done. The length of this essay comes from following the steps that lead him to think he has succeeded. Its moral is: if you must discuss technical problems without actual knowledge, at least acquire enough philosophical sophistication to know what can be said without it.

---

1. ?

1.2 Which Cannot Do What?

I begin by describing his classifications in the starkest possible form. There will be time later to consider subtleties, qualifications and ambiguities. The distinction between computers is that the digital kind have a finite number of states, whereas brains and the brain-like computers of the future do not. The distinctions between kinds of intelligent activities are that some are formalizable whereas others are not, and some are "completely calculable" whereas others are not.

1.3 Infinite Brains?

The assertion that brains do not have a finite number of states might look like a profound challenge to modern physics and readers might fear that it will take us into arguments about the foundation of quantum mechanics of the theory of noise in physical systems. But Dreyfus shows no sign of awareness of such issues. His discussion is based on perfectly ordinary observations and introspections, which all readers, however unsophisticated, can share. Its premise is that people are capable of taking account of an "infinity of facts" and can "respond to an indefinite number of specific situations." This, in turn, is taken as obvious, or sufficiently supported by describing situations where it is not easy to enumerate all the factors that might be relevant to a decision. His esample is placing a bet at the races. Under the heading "The Infinity of Facts and the Threat of Infinite Progression" he explains that the winner cannot be selected on the basis of a restricted set of conditions, such as form, since (Alchemy, p. 68)

> "...there are always other factors, such as whether the horse is allergic to goldenrod or whether the jockey has just had a fight with the owner--which may in some cases be decisive."

It will probably seem, at this point, overly scholastic to discuss

whether it is to be taken as evident that the series suggested by "allergy, fight..." can _really_ be continued to infinity, or what this implies about the brain. So I shall leave this question until we have surveyed the more important distinctions between kinds of intelligence.

1.4 _Superhuman Intelligence_.

Dreyfus' preoccupation with the infinite does, however, cast a shadow on the classification of types of intelligence. Continuing his racing story, he points out, correctly, that a finite machine could not examine an infinite number of possible factors. But he then concludes:

> "If, on the other hand, the machine systematically excluded possibly relevant factors in order to complete its calculations, then it would sometimes be incapable of performing as well as an intelligent human." (_Alchemy_, p. 68)

That the machine should _sometimes_ fail where a human succeeds, seems to be taken here as a definite weakness, a bad trait that marks its sub-human nature. But the trait is not special to machines. Any man who is not superhumanly quick-witted, well-informed and objective will _sometimes_ be incapable of performing as well as some other intelligent human (or even some unintelligent machine).

Dreyfus is certainly being unfair to the machine in complaining that it has this very human trait. So much so, that we must stop to ask seriously whether his _entire_ complaint is of this kind. Is the intelligence machines cannot have, a superhuman variety of infinite infallible intelligence, or the ordinary sort we know in ourselves? It surely makes a difference. Even the existentialists and phenomenologists might feel cheated if Dreyfus' apparently re-assuring remarks turned out to mean merely that

there were no superhumanly intelligent artifacts. Which does Dreyfus mean?

The answer is: sometimes the one, sometimes the other and sometimes the two become confused. In his proof of the theorem "digital machines cannot be intelligent," intelligence means superhuman intelligence. By so interpreting it, he makes the theorem true and, although his proof is extremely weak, one would have no difficulty in providing a sound one.[1] But when it is applied to the criticism of Artificial Intelligence it is strengthened by taking intelligence to mean ordinary human intelligence. I turn next to what he has to say on this level.

1.5 Computers Can't Play Chess.

1.5.1 Nor Can Dreyfus.

I hope readers will pardon a digression to get a debating point off my chest. They will surely understand why I find it irresistible.

In Alchemy and Phenomenology Dreyfus discusses the weakness of chess-playing programs. He plainly gives the impression that they can typically be defeated by human novices. His twice recounted story of how a ten-year-old child defeated a program constructed by Newell, Shaw and Simon was gleefully quoted by the New Yorker and other popular magazines as demonstrating the futility of Artificial Intelligence. While Phenomenology was in press I had the pleasure of arranging for Dreyfus to play against Richard Greenblatt's chess program at M.I.T. and seeing him very roundly trounced. The newsletter SIGART reprinted the game with no comment beyond one phrase from Alchemy:

> "... no chess program can play even amateur chess." (p. 10)

Dreyfus indignantly replied that he was being misquoted: he was making a statement of fact, not a prediction.[2] But was it a mere statement of fact, or one with an intention and a significance?[3] Would he have made it had

---

1. With some suitable definition of superhuman such as infallible ability to know whether a given problem domain is programmable.

2. Note the mild pun in "no *** can ***": "there is none" vs. "cannot possibly." In the little interchange it is savorously unclear who misread whom.

3. For an elaboration of these concepts see any book on phenomenology.

he known that a strong chess-playing program was about to appear? Indeed, is his statement that intelligent behavior "cannot be exhibited by a certain kind of computer" to be taken as a "mere" statement of fact, i.e., if one does exhibit intelligent behavior tomorrow will Dreyfus say that his statement was nevertheless true at the time he made it? Since one cannot know, I shall continue to read what he says as if it were meant seriously.

1.5 Computers Can't Play Chess (resumed).

When Dreyfus formed his opinions about Artificial Intelligence he knew of a very good checkers program but imagined that all chess programs were fiascos. A sensible assessment of this situation (as it was in reality or as Dreyfus saw it) would naturally begin by disentangling a number of very different possible contributory factors. At one extreme the empirically observed difference could be an accurate reflection of fundamental differences between the games. At the other extreme it might reflect nothing deeper than the diligence or competence[1] of the programmers. Dreyfus does not give even passing consideration to the mundane kind of explanation. For him "failure" ineluctably suggests a profound source of difficulty; anyone who thinks otherwise is seen as a victim of self-delusion.

> "... workers in Artificial Intelligence--blinded by their early success and hypnotized by the assumption that intelligence is a continuum--will settle for nothing short of the moon." (Alchemy, p. 83)

> "Current difficulties, however, suggest that areas of intelligent activity are discontinuous and that the boundary is near. To persist in such optimism in the face of recent developments borders on self-delusion." (Alchemy, p. 84)

---

1. Since the burden of his writing is the extraordinary naivete of workers in Artificial Intelligence, it is remarkable that Dreyfus does not give more weight to this kind of explanation of "failures."

I recall that he feels enough confidence in this "suggestion" to predict that "significant developments" will have to wait for an entirely new kind of computer.

To maintain this position Dreyfus has to explain what features of chess and checkers are responsible for the different degrees of success obtained by programmers. I shall examine his explanations in more detail than would be warranted by the relative importance of these games in the total picture of Artificial Intelligence. My purpose is, of course, not to discuss chess and checkers but to show, through extended analysis of at least one case, how Dreyfus reasons. The choice of games for this purpose is partly determined by their relative clarity, but mainly by Dreyfus himself, who devotes more space to this example than to any other.

1.5.2 Can People Play Chess?

Dreyfus predicts that there will be no "significant developments" in programming digital computers to play chess, but gives no indication of the criterion of significance. Is Greenblatt's chess program "significant" or not?

The question places Dreyfus in a serious dilemma:

He may say "NO, Greenblatt's program merely plays creditably good tournament chess; it has not won the world championship and in any case does not 'think' like a human." But in this case his strictures ("self-delusion" and many, many others) are unjustified. The goals of Artificial Intelligence are not confined to creating superhuman intelligence. The content of his general attack on computers and computer scientists is radically changed if he is merely saying that machines will not be better than Baudelaire as translators or more perceptive than Picasso. Are we

self-deluded if our optimism extends only to endowing machines with ordinary human competence in translation or visual perception?

He may say "YES, I was mistaken about chess but my general theory is still valid." Anyone is entitled to admit a mistake. But the question is whether the mistake must be attributed to Dreyfus personally or to the method he uses. If it has already proved misleading in some cases, how are we to know where it is to be trusted or even what is being said?

Each answer to the question about Greenblatt seriously erodes the meaning of the "boundary is near." The statement can be saved in only two ways. Dreyfus can interpret "near" as compatible with being further away than most human performance; or he must admit that the kind of evidence he uses to determine its position is hopelessly unreliable. In neither case is he able to say where the boundary is, so that his "penetrating analysis"[1] becomes irrelevant to judging the degree of delusion and optimism in current research.

1.6 Rational Distinctions Between Chess and Checkers.

> "An alchemist would surely have considered it rather pessimistic and petty to insist that, since the creation of quicksilver, he had produced many beautifully colored solutions but not a speck of gold ..." (Alchemy, p. 86)

Artificial Intelligence would merit all the ridicule Dreyfus gives if its hopes were based on naive or opinionated neglect of demonstrable differences between areas where it has succeeded and those into which it tries to advance. Dreyfus' status as a commentator must ultimately be judged by the quality of his insights into such differences. One could dismiss complaints about looseness on general issues as pedantic if he

---

1. What could Oettinger have in mind?

showed perspicuity in detailed comments on the difficulty of particular problems. But he does not. On the contrary, confusions seen on the more general level re-appear in sharper form and are compounded with others due to sheer incompetence in formal and technical ideas. In this section I illustrate this by examining his attempts to explain the fundamental difference he perceives as separating checkers and chess.

Dreyfus attempts to find a sense in which "checkers" is "completely calculable" while "chess" is not. This is reflected in his definitions of the "areas of intelligent activity" to which the games are ascribed. The next two quotations show how he describes one programmable "area" (called Area III) and one unprogrammable "area" (called Area IV).

> "Area III on the other hand- is the domain of the esprit de geometrie. It encompasses the conceptual rather than the perceptual world. Problems are completely formalized and completely calculable. For this reason, it might best be called the area of the simple-formal.
>
> In Area III, natural language is converted into formal language, of which the best example is logic. Games have precise rules and can be calculated out completely, as in the case of nim or tic-tac-toe, or at least sufficiently to dispense with search-pruning heuristics (checkers)." (Alchemy, p. 78)
>
> "Area IV, complex-formal systems, is the most difficult to define and has generated most of the misunderstandings and difficulties in the field. The difference between the simple-formal and the complex-formal systems need not be absolute. As used here, 'complex-formal' includes systems in which exhaustive computation is impossible (undecidable domains of mathematics) as well as systems which, in practice, cannot be dealt with by exhaustive enumeration (chess, go, etc.)." (Alchemy, p. 79)

Dreyfus first says that "checkers" is completely formalizable and completely calculable and then adds the qualification "at least sufficiently to dispense with search-pruning heuristics." I begin by examining what

can be meant by the unqualified statement.

Obviously the legal rules of checkers are completely formalized and calculable in the sense that one can compute the set of legal moves for a given board situation. But this cannot be what Dreyfus means, since it is equally true of chess. He must mean that the strategy of play can be formalized and completely calculated. But this raises grave problems:

(1) There is no unique strategy for playing checkers. There are many strategies, some of which lead to optimal play and others to different degrees of ability. Some are more like and some are less like our impression of how humans play. What strategies can be said to be formalized and completely calculable?

(2) The best known optimal strategy for checkers cannot be carried out by a real physical computer since it begins by generating all sequences of moves and counter-moves to the very end of the game. Presumably this strategy belongs to Area IV. In any case exactly the same algorithm can be used (in a mathematical sense) for chess and is precluded (in a physical sense) for the same reason. So Dreyfus cannot base a difference between the two games on any superficial consideration of the combinatorial algorithm.

(3) I shall show later that Dreyfus' friend and admirer, Huston Smith, read him (on at least one occasion) as taking the difference to be that a slightly modified version of the combinatorial algorithm would be feasible for checkers but not for chess. But this will not do. First, the modified algorithm, described by Huston Smith as generating all possible sequences up to about the twentieth move, is itself not implementable. Second, even if it could be implemented, we do not have any reason to suppose that it

would generate perfect play, so that, at best, we are left with a difference of degree. Third, there is no semblance of a proof that a similarly modified algorithm would not do as well (or better!) at chess.

(4) The gravest problem concerns Dreyfus' reasons for thinking that "chess" is not "completely calculable." The statement must either be a tautologous repetition of the brute fact that no perfect chess program has yet been written or the result of ignoring elementary distinctions tacitly made in mathematical discussion. Articles on programming chess often begin by noting the practical impossibility of the obvious algorithm and go on to propose "heuristic" algorithms whose aim is to approximate perfect chess play. But no serious mathematician would read this as asserting that no exact and implementable algorithm is possible. We simply do not know one--as we do not know whether Fermat's famous conjecture is true.

Questions about properties of optimal chess algorithms are well-defined and difficult mathematical problems. Anyone who thinks he can pre-judge their answers on the basis of general philosophical or psychological observations really is in the category of circle-squarers and inventors of perpetual motion machines.

(5) At an opposite extreme one might be tempted to read Dreyfus as saying that the processes used by people in playing checkers can be "formalized" and "completely calculated". But it would be preposterous to take this as separating chess and checkers. Everything one can say about the subtlety and apparently holistic character of chess thinking applies prima facie with equal force to checkers. MOreover Samuel's checkers program makes no attempt to simulate human thinking in other respects than

the end result of winning the game. Indeed the tremendous scientific value of Samuel's work consists precisely in showing that a very simple algorithm can sometimes obtain the same resluts as the "holistic", "intuitive" human mind. Naturally this leaves open a host of questions about whether this is due to special features of checkers. But these questions are not resolved, one way or the other, by declaring that "checkers is completely calculable."

Thus it is far from clear what Dreyfus is saying. Worse muddles appear when we examine the qualification he uses to bring "checkers" into the class of programmable "areas of intelligent activity."

1.5.3 Can a Machine Examine 64 Squares One After Another?

To explain the qualification some technical terms must be introduced. This section could be passed over by readers who are already convinced, as everyone should be, that Dreyfus must be precluded, for lack of mathematical skill, from saying anything specific on this score.

His qualification stems from the observation that in chess the average number of legal moves per board position is greater than in checkers. He does not, as usual, mention numbers and the concept of "average" has a certain vagueness, but one can accept the point and take 30 and 7 as plausible values. The most important quantity in the picture is the "exponentially increasing" number of move sequences, 30^n and 7^n respectively, generated by algorithms based on complete look-ahead, i.e., on considering all legal moves, all counter moves, all replies to these and so on through n stages.

The simplest practical chess programs limit the number of sequences generated in two ways: "in-depth" by giving n a small value, such as 6,

and "in breadth" by considering sequences of all "plausible moves" rather than all "legal moves." A separate sub-program, called "the plausible move generator", is used to decide which of the legal moves is to be considered "plausible."[1]

Dreyfus' discussion recalls his man at the races. If the machine "considers" only some moves it might leave out the best one.[2] He says:

> "We cannot run through all the branching possibilities even far enough to form a reliable judgment as to whether a given branch is sufficiently promising to merit further exploration. Newell notes that it would take much too long to find an interesting move if the machine had to examine the pieces on the board one after another." (Alchemy, p. 19)

Now this cannot be taken literally. The machine can, does and must "examine all the pieces on the board one after another," if only to decide which of the legal moves is to be taken as "plausible." There is, of course, a problem: to "examine" the piece is not enough, the examination would have to use an infallible algorithm to avoid any danger of passing over good moves. But this problem is not a specific consequence of "breadth limitation." Even if all sequences to depth n were considered, it would still pass over some good moves unless the evaluation of the hypothetical future situations were infallible. Nor is it specific to machines--people often fail to choose the best move. But I shall pass over thispoint here.[3]

---

1. Plausible move generation is related to the tree search somewhat as the preliminary survey of the board by a human player ("zero-ing in" in Dreyfus' language) is to checking out the hypothetical consequences of making the move (Dreyfus' "counting out"). Human players will be discussed later. Analogies must not be taken too literally.

2. The SUPERHUMAN HUMAN FALLACY involved in taking this as a proof that machines are necessarily inferior to (or even different from) people will be discussed elsewhere. In this section I am concerned only with Dreyfus' technical distinctions between programs for chess and for checkers.

3. I return in an appendix to show that Dreyfus does take the absurd statement literally and to discuss what Newell really meant.

Now Dreyfus tries to distinguish between chess and checkers by saying that simple-formal programs use "search-limiting" procedures, but can dispense with "search-pruning" procedures. These terms are not clearly defined, but most probably "pruning" refers to limitations of breadth and "limiting" to limitations of depth alone. If this is so (and most likely even if the words mean something else) the difference is simply quantitative and does not establish a fundamental qualitative dichotomy. Programs can obviously use a mixture of the two kinds of procedures (or procedures of an altogether different sort) to a degree that depends on the ever increasing availability of computational capacity. In fact, Greenblatt's chess program can be made to vary the threshold of "plausibility" as a function of level so as for example to include all moves on the first k levels. This means that it will never fall into "traps" that might catch a novice who fails to see a disastrous pin or capture two moves ahead. Dreyfus' impression of a qualitative difference comes from his literal-minded and rigid understanding of the literature: inability to think in terms of programs forces his concept of a chess program into the exact image of descriptions he has read.[1] He never considers the possible effects of even minor modifications, unless these have been spelled out in the very papers on Artificial Intelligence which he holds in such contempt.

The second point bears on the checkers side of the opposition. So long as Samuel's program is not actually invincible, it cannot be said to show that programs for checkers can dispense with search-prunig (or anything else). At best it shows that a certain level of play can be so achieved--but if we look more carefully we see that this level has improved

---

1. Of course: as he understands them!

gradually as the reward for more dedicated work than has been devoted by any equally gifted programmer to any chess project.[1]

In short, Dreyfus has not done very well in providing penetrating insight into the difference between chess and checkers. That there are differences is a mere statement of the universal principle that no two things are alike. That we cannot formulate them clearly is a reflection of the undeveloped state of the young science of Artificial Intelligence. But progress towards doing so will come from hard mathematical analysis and careful experiments. Dreyfus has made no contribution.

---

1. Though Greenblatt's chess program is catching up--but so is its performance.

Before leaving chess and checkers I shall comment on one more passage, presented in another slightly rhetorical digression, as seen through the eyes of one of his principal intellectual admirers.

1.6.1 How Dreyfus Gets Huston Smith Into Trouble.

Prof. Huston Smith, also of M.I.T., has used Dreyfus' "work" in his own campaign for the recognition of fundamental differences between "natural and artificial intelligence". To understand how Dreyfus has achieved authoritative status in some circles, it is revealing to note that Prof. Smith's perception of some very gross errors of logic does not protect him from following Dreyfus to disaster on other points. The next passage is quoted from a typescript manuscript given to me by Prof. Smith.

> "The problem facing a chess plaing program is the one inevitably connected with very large choice mazes; namely, exponential growth. Some games are simple enough to be programmed to win or draw every time: nim and tic-tac-toe are obvious examples. Other games are too complicated for this, yet admit of machines that can be programmed to play them very well. Here checkers is perhaps the best example. In checkers only two kinds of moves are possible, captures are forced, and pieces block one another. Under such restrictions it is possible to explore all possibilities to a depth of twenty moves or so, which is sufficient to play a very good game. Already there exist machines which very few human players can beat, so a world's champion checker player that is mechanical seems likely. In chess the situation is radically different. 'It has been estimated that there are about 10^{120} different paths through a complete chess maze.' A maze of this complexity does not permit one to run through all the branching possibilities far enough even to form a reliable judgment as to whether a given branch is sufficiently promising to merit further exploration. Consequently 'it is beyond the limits of plausibility that a computer will ever be able to play "optimum" chess by ... brute-force computing,' by counting out."

Smith thinks it is "possible to explore all possibilities to a depth of twenty or so." Well it is not. Any of my freshman students would have been glad to show him how to see the absurdity of the idea. To calculate the number of possibilities one must first know how many moves can be made on the average. If the student were informed about checkers he would know the number seven. Otherwise he would probably estimate five as reasonable

--though even three is enough to show the impossibility of looking ahead as far as twenty moves. Now $5^{20} + (5^4)^5 = 625^5 > 512^5 = 2^{45} = 2^5 2^{20} 2^{20} >$ 32,000,000,000,000. This number of microseconds is a year, so that even on the exceedingly optimistic assumption that the computer could analyze each possibility in a few microseconds it would take centuries to play a game.

Unfortunately, instead of asking the advice of a student with some facility in arithmetic and elementary knowledge of computers, Prof. Smith quite naturally relied on his colleague Dreyfus, who is reputed to be an expert on computers. The origin of his extraordinary mistake is revealed on p. 19 of Alchemy:

> "... one can explore all possibilities to a depth of as many as twenty moves ..."

Dreyfus' statement has a mild ambiguity. He may have meant to say what Smith thought or he may have meant "as many as twenty" in the shopkeeper's sense of "from $10", i.e., "twenty or fewer". On the first interpretation he is plainly wrong. But the second fails even more superficially to establish a difference between checkers and chess, and this is, no doubt, what made Prof. Smith choos the other. For, while a checkers program may pursue twenty possibilities in exceedingly rare situations of long forced sequences, this is true of chess as well. In neither game can this happen often enough to be significant.

1.7 "Success and Subsequent Stagnation".[1]

Dreyfus has not succeeded on either empirical or on rational grounds in showing us how chess and checkers differ in relation to programming. The implications go far beyond these games: they show that his classification of intelligent activity into "distinct areas" is founded on muddle rather than insight. He has given no reason to suppose that scientists are self deluded when they take their success in limited areas as indications of how to go forward rather than as proof that anything beyond these limits simply cannot be programmed.

His last line of defense is to declare that, as a matter of brute fact, Artificial Intelligence is in a state of stagnation. This appears to be an empirical claim. But, the matter is not so simple, What one sees

---

1. Alchemy, p. 84.

in the world is a function of what one knows about it. The evaluation of the rate of progress needs technical judgment and the greatest sensitivity to differences between the "difficulty" that puts an entire conception in doubt and the kind that must mark the normal route to its realization.

Before examining the quality of Dreyfus' technical judgment I recall that the assertion of stagnation is not secondary to a more theoretical and fundamental critique, but the keystone of his theoretical structure. For everything is postulated on the theory of discontinuity in intelligence and this in turn is based on the fact of stagnation.

1.7.1 What the Eye Does Not See ...

I begin with a typical example of Dreyfus' style of comment on a project in which, as is usual in Artificial Intelligence, the engineering component is as important as the formal mathematical ones. Commenting on an experiment in which a computer is linked to an electronic eye and a mechanical hand, he says:

> "At present, it takes the machine minutes just to pick up a block. A more flexible arm endowed with more degrees of freedom will involve calculations requiring even longer computations. If one adds to this the fact that, in the case of any skill which takes place in real time (such as playing ping pong), these calculations must be completed before the ball arrives, the outlook is not very promising.
>
> In the light of these difficulties, what encourages researchers to devote their research facilities to such a project? Simply the conviction that since we are, as Minsky puts it, "meat machines" and are able to play ping pong, there is no reason in principle or in practice why a metal machine cannot do likewise." (Bodies, p. 27)

This is a perfect example of how the "amateur scientist" is unable to distinguish between essential and accidental features of the situation he sees before him--again because, like the poor machine at the race track, he is unable to take account of all the relevant factors, such as the intentions of the experimenters.

The number of minutes taken to pick up the block is a meaningless quantity with no significance in principle. As a matter of fact it was reduced by a factor of ten while Dreyfus' paper was in press! But this reduction has no significance either. The reason the experimenters were unperturbed by the time taken is not the analogy with "meat machines," but the firm (and elementary) knowledge that the use of slightly different equipment could reduce the time to a fraction of a second!

It is perhaps worth explaining why slow machinery was quite consciously chosen. The speed of operation was determined by the eye used, rather than by the arm. Now there are many kinds of "electronic eye". A relevant difference is that some, like a TV camera, systematically scan the scene while others, like the image dissector used in this experiment, can be directed at will to particular points. This property makes it easy to write and modefy programs for the image dissector. The price paid is slowness of operation under certain conditions. It is reasonable as a research strategy to use the slower eye in the phase of the project in which many programs are being studied and constantly modified. When the time comes to settle on one and build a working robot, not only will a faster eye be used but many operations carried out by the computer will be frozen into special "hardware" devices.

It is a matter of routine (and dollars) to transform programs developed for one kind of eye to another; the fundamental research bears on the development of algorithms suitable for machine vision.

Computational time is actually an important factor in these experiments. But the relevant time is not the "minutes" observed by Dreyfus as an amateur onlooker, but the milliseconds observed by people involved in the experiment as the time taken by various algorithms to analyze data once it is in the computer.

A propos of the reference to ping pong:

(1) The movement of the arm in real-time is certainly not the problem. This same robot was in fact programmed to catch a ball thrown to it. It could succeed because a ball is a visually simple object.

(2) It is ridiculous to suggest that metal arms cannot be moved as fast as meat arms. Indeed making a metal ping-pong player is not necessarily a project in Artificial Intelligence at all. It could be successfully (but dumbly) done by miniaturizing existing military technology.

(3) Incidentally, the ping pong story is not due to Minsky, but was started by me as a joke. It is amusing that many disclaimers have not succeeded in dislodging it from the public image of the robot project.

1.7.2 Sordid Dollars.

Dreyfus cites as an example of "the disparity between prediction and performance which is characteristic of artificial intelligence" the slow development of automatic reading devices. Ten years ago, he says,

> "flight to the moon was still science fiction and the print reader was around the corner. Now the moon project is well underway while, according to Oettinger in 1963, no versatile print reader is in sight."
> (Alchemy, p. 16)

To bring out the extent of his bad judgment in engineering matters[1] I note:

(1) The construction of a print reader is in principle so straightforward as hardly to merit being classified as Artificial Intelligence.

1. The next sections up to 1.10 merely document this already obvious point most readers should skip.

(2) The difficult problems are entirely economic: to make a machine read at less cost per word than human transcribers. This has become gradually more realistic as the price of computing machinery falls. One can now buy several dozen different kinds of print reader.

(3) On the other hand, space travel is under no comparable constraint of economic competition, and the 10,000 to 1 ratio of research budgets for the moon project and for print-readers cannot be simply ignored!

1.7.3 The Invisible Bug.

Many programs dismissed by Dreyfus as failures were shown by later developments to be fundamentally correct in their logical structure but degraded in performance by bugs. A good example is the chess program. The Greenblatt program that defeated Dreyfus may be the first to have been sufficiently well debugged on a programming level for its ability at chess to be visible for study and improvement. The early programs mentioned by Dreyfus played so few games that one can be almost sure they were riddled with bugs. Yet Dreyfus is happy to use their performance as prime evidence for very general conclusions.

1.8 The Amateur Scientist Syndrome.

The previous examples show how the amateur scientist is drawn inexorably into a literal-minded empiricism by his inability to see beyond the surface facts. In other cases he is not able to see even the surface. His access to research is usually through secondary sources and early texts written at a stage when the subject was simple enough for work

to bedescribed in lay terms. His knowledge is consequently always out of date and over-simplified.

A survey of the development of physics from Newton to modern times might not suffer much if its author were unaware of work done in the last decade. The same lag is obviously intolerable in a survey of a subject that began in the early nineteen fifties! But the programs discussed by Dreyfus are invariably the earliest and simplest in each category; in fact most of them are drawn from an anthology published in 1963, of papers reporting the very earliest work on the subject. This alone might account for his impression of "dramatic early success" followed by "stagnation"! Yet he considers this meagre historical evidence sufficient for a comparison with the history of Alchemy and for his theory of stagnation. In fact he cannot claim to have begun to make a case until he has examined the work done since 1960 at least sufficiently to show that it is worthless.

1.9 Inspiration of Phenomenology Vs. Organization of Science.

A more subtle and important form of the amateur scientist fallacy comes from failure to see the dynamics of scientific development.

The amateur scientist starts from the premise that the purpose of Artificial Intelligence is to make intelligent artifacts; and then judges each piece of work by whether it has, in itself, actually produced intelligence--as if one were to judge workers in aerodynamics by their ability to fly. This tendency is related to the preoccupation with early work. At the beginning of the century many people engaged in research on aviation did in fact make and test whole airplanes. It is a mark of maturity that fundamental contributions to aviation might be made by a metallurgist or

even a mathematician who could no more construct or fly an airplane than fly themselves.

Dreyfus' behaviorist criterion of success leads him to see the beginnings of similar maturity in Artificial Intelligence as stagnation. The goal of the earliest experiments had to be the construction of a program to display intelligence in some area--for example to play chess. Dreyfus complains that instead of playing perfect chess, they ran into difficulties. Of course they did. The purpose of the experiments was to find the obstacles and difficulties and the characteristic of later and more mature stages of development is that research can be directed to technical problems. Dreyfus is unable to see this for two reasons: his style of thinking generally prevents him from looking; and even when, occasionally, he does look his technical ignorance prevents him from understanding studies whose very definition is technical.

Some examples follow.

Example 1

Dreyfus' comments on the time the robot takes to pick up a cube is a striking case. The purpose of this experiment was not to lift cubes as quickly as possible but to study certain algorithms and pieces of equipment.

Example 2

Some work has been done on specialized problems in chess. For example, Simon devoted a study to the possibility of using algorithmic rules to discover combinations. In commenting, Dreyfus does not even consider the technical question of whether these rules do generate combinations, but as we shall see later, pours scorn on Simon for the limited scope of the study.

Example 3

Greenblatt's work on chess is no more to be measured by the level of play he achieves (though this is indeed impressive) than the flight at Kitty Hawk by the twelve seconds the Flyer remained aloft. One of the great contributions of the Wrights was to abandon the previous goal of an inherently stable aircraft and, instead, make the machine sufficiently unstable to be controlled by the pilot. Naturally, this aspect was as invisible to the public (those who cheered and those who jeered) as the details of a program to Dreyfus. Greenblatt made a similar contribution by designing his program so as to maximize the ability of the programmers to understand and modify its operation. Dreyfus does not see this kind of "progress." Nor does he see as a source of progress the battery of programming aids (including assembler programs, debugging programs, program-editing programs) Greenblatt was able to use. Neglecting these is like failing to see the importance of machine-tools in the development of aviation.

Example 4

Dreyfus complains again and again that computers cannot handle large quantities of information because they take so long to search lists. It is apparent that his image of information retrieval is exhaustive item-by-item search through a list--as if the machine would have to read the entire telephone directory to find a number. One of the areas of specialized research in computer science is the study of techniques for handling large quantities of information. Some of these are as simple as the use of alphabetical ordering in the telephone directory. One would indeed have to read the whole book (or rather, on the average, half of it) if the names were arranged randomly or if the "association" were "number-name"

instead of "name-number."[1] Others, like "hash-coding" range from a mild degree of mathematical sophistication in most uses, to extreme ingenuity in the way it is used by W. Martin in his recent Ph.D. thesis (M.I.T. 1967). Others again depend on special hard ware ("associative memories" etc., etc.). All this Dreyfus ignores. Yet he has the simple-minded audacity to announce, as if it were a deep personal discovery, that any digital computer would take too long to search "enormous lists" (and, of course, without saying how big is enormous!).

These examples provide no more than a glimpse of the extent of Dreyfus' incompetence. No page, and hardly a paragraph of Alchemy is free of some piece of tehcnical nonsense. To pursue them all would be boring. Yet Dreyfus' guerrilla warfare style of argument--throw many bombs, some might do damage, too bad if others fizz--allows his admirers to maintain (as experience shows they will) that whatever muddles I draw attention to he is nvertheless right on the whole. There is nothing I can do about that. But from another point of view just these few examples should suffice for, after all, there are some things no one with any understanding of a subject could say.

---

1. Note for philosophers and psychologists. Dreyfus complains that the "knowledge" incorporated in a program is "associationist." Question: is the use of the number-name rather than name-number association, itself an "association"? Isn't it precisely such structural knowledge that is least "associationist" in people?

With this I leave these technical issues and return to more general ones.

1.10 The Size of Intelligence.

In a certain relevant sense a little intelligence is not intelligence at all but stupidity. Any program that does just one thing well is at best more like an idiot savant than like an intelligent man. However well the machine plays chess, one can say that it is not intelligent because it did not learn but was programmed. And if it did "learn" (as Samuel's checkers program does in a limited sense) one could say that it learned under special, formalized conditions, not by free communication in English. And if ... But the principle of modus post ponens can be used indefinitely to refuse the conclusion by always demanding one more line of proof.

The dictum also has a less analytic sense. Even if we confine ourselves to the limited goals of making machines capable of playing chess or of using metal hands and eyes to operate the simple tools of a carpenter's shop, one might have to give them access to very much more logic and knowledge than any program has had up to now. To "see" the world, the machine needs not only eyes but also (in some explicit or implicit sense) knowledge of light and shade and shadows and perspective and physical laws governing stability and forms of representation of geometric configurations and so on ... not to infinity but perhaps beyond the bounds of what one programmer could achieve with one IBM 7094.

A sensible discussion on the prospects of robotics must recognize at least four kinds of barriers which could be called Turing Barriers, Theory Barriers, Economic Barriers, and People Barriers. Turing Barriers are

absolute: there are things no finite machine can do. Theory Barriers are set by our current state of knowledge: the limitation is in us, not the machine. Economic Barriers are in Washington: they prevent us from carrying into practice well-formulated or tentative ideas that might need a hundred man years of programming or a billion bit addressable memory. People Barriers are the most frustrating and least metaphysical: the number of talented people with appropriate knowledge and interest available to develop new ideas and implement existing ones.

Where would a sensible person place the project of building a mildly intelligent robot in relation to these barriers? Two kinds of extremists are closely related: on the right, the Dreyfusians who believe that the project is impossible, i.e., who see a Turing Barrier; and on the left, the romantic kind of cyberneticist who expects intelligence to emerge from properties of special machines such as being randomly conncected, self-organizing, capable of growth or merely unintelligible, i.e., who see a pure Theory Barrier.

The extremists are united in holding a "faculty theory" of intelligence as opposed to an "epistemological theory." The faculty theory sees human intelligence as the manifestation of a special (usually otherwise unspecified) property of the brain. The epistemological theory sees it as emergent from knowledge, and abstractly definable operations on this knowledge. It follows almost necessarily from the epistemological theory that a truly intelligent system would have to be a large one, so that the Economic and People Barriers are seen as paramount in assessing the magnitude of the problem and the reasons for limitations of past experiments. In particular it would not expect a one-man effort with a limited access to a computer center designed for other purposes to produce a high level

of general intelligence.

It seems to me almost perverse of Dreyfus to attach any importance to "failures" (granting, for the sake of argument, that they are such from his perspective) of very small experiments. What does he expect? I could understand this attitude in someone who was romantically convinced that the task must be easy. But someone who claims to see that it is infinitely difficult should also admit that it is more difficult than, say, creating a new programming language or a big time-sharing system. All the experiments discussed by Dreyfus, invested far less effort in programming than is needed for these relatively trivial jobs. One would, therefore, expect him to wave them off as irrelevant petty surface-scratching nonsense rather than to take them seriously as suggestive of general and profound conclusions.

There is only one explanation: he does not think in these technical terms. Someone who does not ask himself questions about arithmetic simply does not see it as absurd to say that a checkers program can search all possibilities to a depth of "twenty or so" (H. Smith). Someone who does not ask himself questions about the act of programming does not see it as absurd to deduce from the effects of bugs in a chess program that the boundary of what finite-state machines can do is near (H. Dreyfus).

Believers in Artificial Intelligence should not be angry, but delighted. One scarcely needs the example, but one could not have a better one to illustrate the contrast between faculty theories and epistemological theories. Dreyfus is as silly in his comments on programs as is the poorest chess program in the kind of game that needs real knowledge. And for exactly the same reason.

2.0 People.

If computers cannot, and people can, go beyond the programmable level of competence at chess or Russian translation, it follows that people have some very special property. Dreyfus gives two answers: they have bodies[1] and their brains can perform three[2] "Uniquely Human Forms of Information Processing." I shall concentrate on the second answer.

2.1 Uniquely Human Forms of Information Processing.

The three UHFIPs are:

Fringe Consciousness.

> "The ability to retain this infinity of facts on the fringes of consciousness allows human beings access to the open-ended information characteristic of everyday experience..." (Alchemy, p. 69)

Ambiguity Tolerance.

Essence/Accident Discrimination.

Now it is certainly true that an intelligent robot would need to distinguish between the essential and the accidental features of a situation. For otherwise, when it sees another robot take several minutes to lift a cube it will be driven to false conclusions by its inability to distinguish between accidental features of the particular experiment and the essential difficulties of programming computers to use visual information.

One can recognize the "essential" only to the extent that one has appropriate knowledge and more elementary analytic skills. Dreyfus could say that we need UHFIPs as well. Man and computer fail in the absence of knowledge but knowledge will not avail the computer. How can Dreyfus

---

1. This idea is attributed to S. J. Todes.

2. Sometimes he says four (Alchemy, p. 18).

know this? His empiricism comes out very strongly when he draws such conclusions from observation of the most rudimentary programs. Does the program play badly at chess because it is a computer or because it is an ignorant computer?

What can Dreyfus do? He could compare computers and people under conditions of relatively closer equality of knowledge. For example he might compare a chess program with a human beginner, or a theorem-proving program with a high school sophomore. But under these rules the computer wins the competition.

This chapter is devoted to Dreyfus' attempts to describe certain sectors of human intelligence, which have not, as far as he knows, been the subject of experiments in programming, in such a way as to make it unplausible that any program could imitate them. Unfortunately the plausibility of this judgment is a function of the judge's knowledge. Dreyfus' discussion of chess and checkers acquired some contact with reality from a slight familiarity with actual programs. His discussion becomes truly bizarre when he has to rely on his own resources to imagine what programs are possible.

I recall that Dreyfus classifies "areas of intelligent activity" by two dichotomies: formalizable or not; and completely calculable or not. Where an actual program exists, as in chess, he is, for obvious reasons, inclined to call the area "formalizable" and attribute difficulties to its not being completely calculable. Where he knows of no program he is inclined to say the area is not formalizable. All the problems about degrees of ability and about the rational and empirical basis of the classification re-assert themselves in this new context. I shall not, however, spell them out in detail but concentrate on some new sources of confusion.

Most of the examples I shall discuss are apparently intended as evidence for the existence of UHFIPs. Indeed they are given by Dreyfus as examples to illustrate the appropriate UHFIP. I am not quite sure whether the logical structure of the argument is intended to be: people use UHFIPs in these situations therefore computers cannot emulate them; or computers cannot emulate people, therefore people have UHFIPs. The second is more reasonable, but I shall not find it necessary to make a decision. Dreyfus' arguments are so bad that it scarcely matters what conclusion he wishes to draw.

Another question I shall not try to resolve is the sense in which UHFIPs are "forms of information processing." I know what "information processing" usually means when used in the context of the everyday discourse of computation centers. I even understand, though less clearly, the meaning of the term in biological or psychological contexts when the speaker clearly has in mind a model based on computation or information theory. But I am at a loss when a speaker explicitly denies the relevance of computers as models and the basic assumptions of information theory, but nevertheless uses words whose technical meaning derives from computational information theory. But again, I am under no obligation to make sense of Dreyfus conclusions. His discussion of concrete situations is specific enough to be analyzed without doing so.

2.2 The Unthinkable Fallacy.

The theme of the next examples is Dreyfus' readiness to take his inability to think of an algorithm as reason to believe that there is no algorithm.

2.2.1 Contexts.

I begin with a very clear case. Dreyfus believes that computers would have trouble in situations where the meaning of a symbol depends on

the context. The source of this idea is probably the failure of some simple-minded attempts to resolve ambiguity for mechanical translation. Dreyfus offers the following remark as an explanation not only of past failures but as support for the contention that no program ever will succeed.

> "Since the meaning of each term contributes to the meaning of the context, every word must be made determinate before any word can be made determinate, and we find ourselves involved in a circle." (Alchemy, p. 72)

If this is a "circle," it is not a vicious one, for one could equally well say that the equations:

$$x + y = 101$$
$$x - y = 99$$

cannot be solved because y must be made determinate before x can be made determinate and vice versa.

In another place Dreyfus expresses the same difficulty more generally.

> "It is hard to imagine how a computer, which must operate on completely determinate data according to strictly defined rules, could be programmed to use an underdetermined expectation of the whole in order to determine the elements of that whole." (Bodies, p. 19)

Dreyfus is punning. There is a sense in which the data on which a program must operate is always determinate, i.e., in the sense that "x + y" is perfectly determined as a formal symbol. But, of course, it is perfectly indeterminate as a number. I suspect that part of Dreyfus' difficulty comes from a rigid way of thinking (like his binary approach to being able vs. not being able to play chess) that contrasts singularly with a sophisticated programmer's fluent handling of the multi-valent meanings of objects that appear now as binary numbers, now as instructions, now as naming symbols and so on. But whether this diagnosis is right or wrong, there is obviously no difficulty whatsoever in making a program use "an underdeter-

mined expectation of the whole in order to determine the elements of that whole."[1] Every good two-pass Assembler does just this! Of course, Dreyfus will say he meant it in a more subtle sense. But he neither says what sense he intends, nor, what is more important, does he give any reason for his assertion beyond the subjective fact that he, who is not a programmer, finds it difficult to imagine how to write such a program.

2.2.2 Another Example from Chess.

Dreyfus uses an example taken from chess to elaborate his idea of a circle in mutual determination of indeterminates. He sees a similar circle in the attempt to define "danger":

> "Clearly, for a more and more refined definition of danger, a larger segment of the total situation will have to be considered. Moreover, at some point the factors to be taken into account, such as tempo or possibility of a forced mate, will themselves have to be defined in terms which involve the determination of whether pieces are in danger." (Alchemy, p. 73)

> "The reason a human player does not go into a corresponding loop is that ... he is able, for example, to define 'danger' as precisely as necessary to make whatever decision the situation requires, without at the same time being obliged to try to eliminate all possible ambiguity ... The digital computer by definition lacks this ambiguity tolerance." (Alchemy, p. 74, my emphasis. The dots represent long passages.)

There are several levels of naivete in this argument. The definition of a concept in "terms of itself" does not create a vicious circle unless it is done badly. For example, a standard definition of the function "n factorial" in a typical programming language could be transcribed into English as:

> If n = 0, FAC(n) = 1
> Otherwise, FAC(n) = n × FAC(n-1).

---

1. Unless Dreyfus gives this phrase a truly esoteric meaning for which there is no indication in his texts.

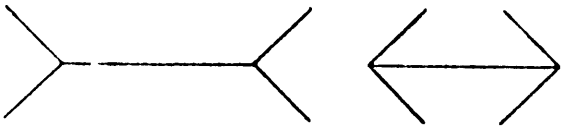
Such "recursive definitions" are the stock in trade of all sophisticated programmers. One would very naturally use them in defining "danger." If, however, one had technical reasons for not doing so, one would use exactly the kind of partial determination Dreyfus declares to be "by definition" outside the scope of the computer.

This might fail to meet the specification if "as precisely as necessary" carries the implication that the machine should never be mistaken in deciding how precisely to define the term. But on this reading there would be no reason to suppose that humans can meet it. So, depending on the interpretation, Dreyfus' text commits one or both of two fallacies: the superhuman fallacy or the UNTHINKABLE FALLACY: declaring a formal procedure to be impossible for no better reason than that one cannot think how it could be carried out.

2.2.3 The Funniest Example.

The next passage is an extremely curious expression of the difficulties Dreyfus has in coping with concepts like "determinate" and "explicit."

> "If this assertion (that human information processing is explicable in discrete terms) claims to be based on a description of human experience and behavior, it is even more untenable. Certain forms of human experience and behavior clearly require that some of the information being processed not be made perfectly explicit. Consider a specific example from gestalt psychology: When presented with the equal line segments in the Muller-Lyer illusion (Fig. 1), the subject cannot help but see the upper line as shorter than the lower. The lines at the end of each segment (which are not considered explicitly, but which rest on the fringes of the perceptual field) affect the appearance of the lines on which attention is centered. Now suppose a machine with some sort of electronic perceptors perceives these lines by scanning them explicitly point by point. It will simply perceive the lines as equal, with no suspicion of illusion." (Alchemy, pps. 56-57)



The statement is rather more piquant to those who know that certain real programs are actually subject to an illusion that goes in the same direction as the human one--though, of course, one cannot say it has the same cause, for there is no satisfactory theory of these illusions in people.

The explanation of the computer illusion has an interesting epistemological overtone. It is related to the fact that a "line" is not simply reducible to the set of points in it: to know it as a line one (man or machine) must look also at the surrounding space and is liable to be influenced by what one sees there in possibly unexpected ways. Naturally, one can make more complex programs that take account of any particular effect of this kind. It is conceivable that no program could take account of all such disturbances and so be completely free of illusion. By analogy with the racing story one might be tempted to deduce that machines are therefore inferior to people. But Dreyfus has obvious reasons for not drawing this conclusion in this case.

The aspect of this statement that most emphatically disqualifies Dreyfus as a commentator on computers is the insensitivity of "will simply perceive". But more insight into deeper muddles is given by pausing for a moment on the use of "explicit" and its relation to "discrete."

In a "discrete" system, he seems to be saying, "information" must be made "perfectly explicit" or completely ignored. What can this mean? Suppose that a discrete computer reads values of light intensity in the

form of an average over a region, or of a fourier transform or more generally as a function f(R), where R is the set of actual light values at certain points on its retina. One might very well say that the computer does not explicitly have the value at a point P in R. Dreyfus must, if he is saying anything, have a sense of "explicit" that excludes this, so as to make it true that "discrete" systems must deal only with perfectly explicit information. If he does, he goes to no trouble to explain it. It is more likely that the statement reflects the same confusion we saw in 2.2.1 in connection with "determinate data." [1]

2.2.4 Penetration Or Muddle?

I must pause to make it perfectly explicit that I am not saying there are no problems.

There are deep technical mathematical problems about computations involving many mutually dependent variables. Even the simple example of simultaneous linear equations is not yet completely understood from a computational point of view. Every student of mathematics learns an algorithm for solving systems of equations but no one knows which algorithms are optimal for computers.[2] This means that we do not know how big a linear system can be solved by a given computer in a given time. The situation is even worse when the relation between the variables is less analytic. Almost nothing is known about the following elementary problem with crucial implications for "information retrieval": what is the minimal time needed

---

1. I shall return, later, to speculation about what is really in Dreyfus' mind.

2. It is only in the last two years that explicit progress has been made towards solving this important problem.

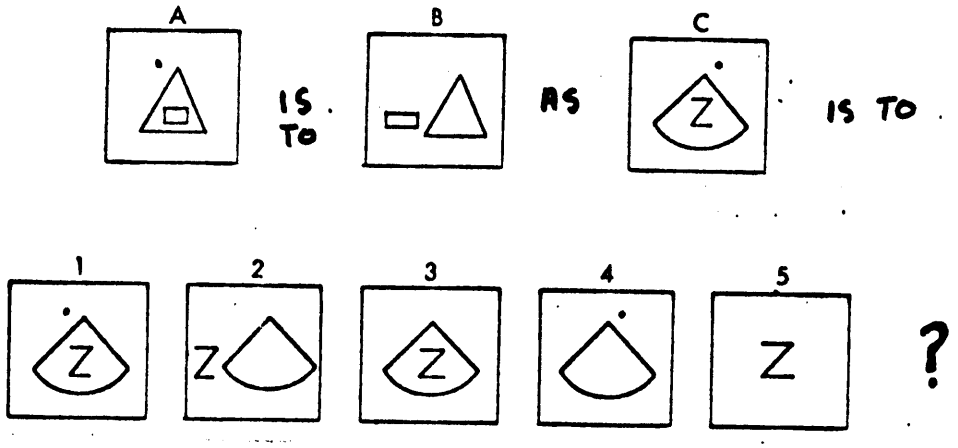
by a serial computer with addressable memory to find, for a given point in binary n-space, its nearest neighbors in a large list of points in the same space. A truly penetrating analysis of the limitations of digital computers would have to solve problems of this kind. Titillating comments by phenomenologists about their personal difficulty in imagining suitable algorithms are irrelevant.

Philosophers might have something to contribute by bringing more clarity on a conceptual level. Although I find it hard to imagine how they could do so without understanding technical problems, I do not say it is impossible. But I do say that Dreyfus' discussion of the Muller-Lyer illusion does not clarify the key concepts used. On the contrary it shows that he is no less confused than the rest of us and lacks even the awareness that there is a problem. I would dearly love to see a well-developed theory of the kinds of "explicitness" and "determinateness" "data" can have. But there is less than no evidence that Dreyfus is the man who can make one.

On the practical level the resolution of local ambiguity is a serious problem in programming computers to listen to speech, see objects or extract meaning from texts in natural languages. Everyone knows this without Dreyfus. The question is whether his analysis casts any light on the nature of the difficulties. I say that he does not show the slightest sign of seeing where the difficulties lie. This is not surprising when we note that he makes no reference to any serious study of these problems, although many people who live within ten miles of him have worked on them, e.g., D. Bobrow, T. Evans, A. Guzman, M. Minsky, S. Papert, R. Quillian, L. Roberts. I am not suggesting that Dreyfus would learn the answers to all the fundamental problems by talking to these people.

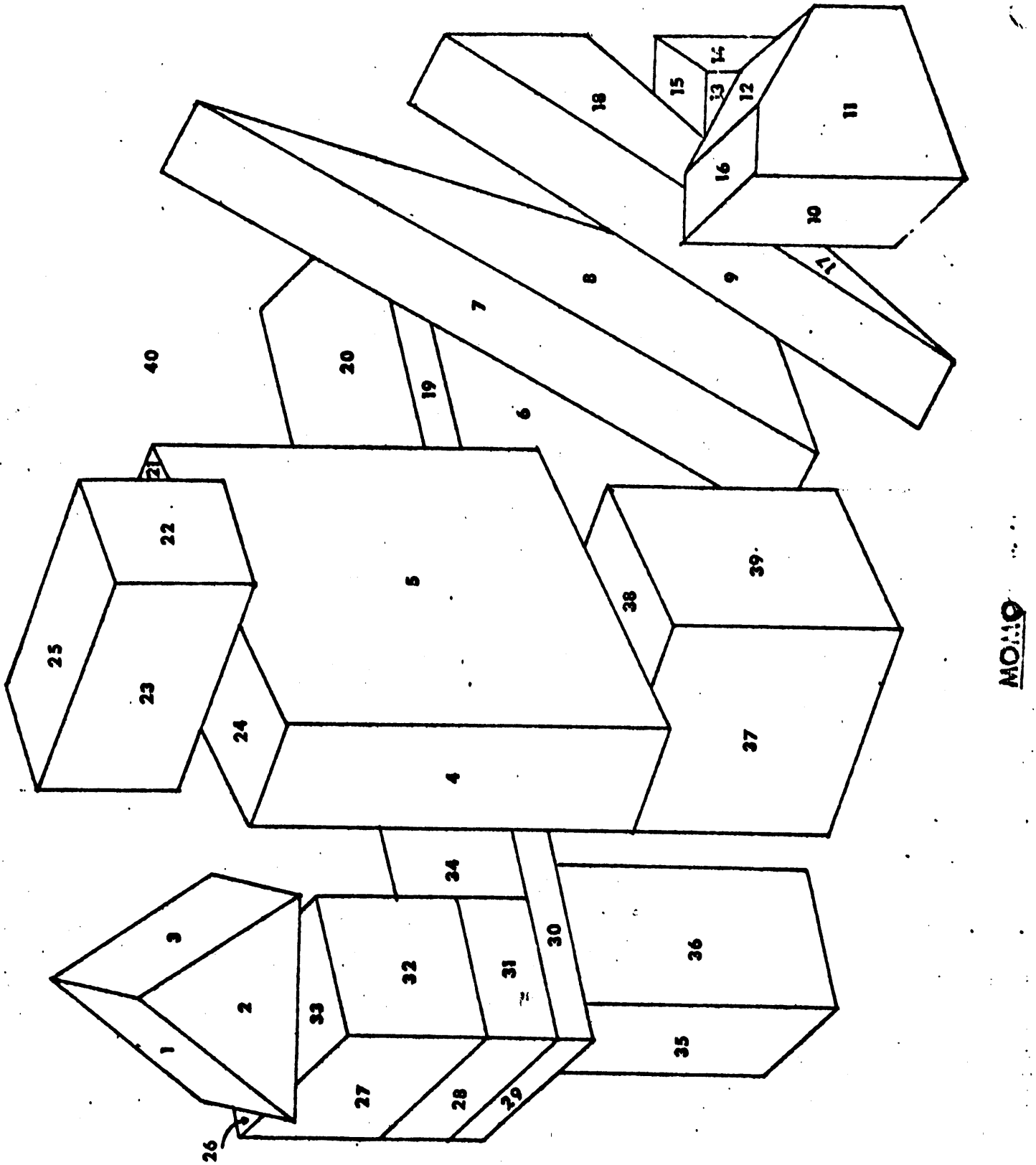
I am saying that a critical analysis of their work might lead to a more realistic penetration of the difficulties than uncontrolled fantasy about the Muller-Lyer Illusion. I am saying that Dreyfus shows deep irresponsibility in pretending to discuss Artificial Intelligence without reference to the relevant modern work.

For readers who know these studies or can imagine the details I shall comment very briefly on their relevance to the problems of ambiguity, context and underdetermined expectations. Bobrow's program solves algebra word problems like: "Jane is three times as old as Mary was when ...". Its ability to "understand English" comes from its use of the underdetermined expectation that the story is describing an algebraic relationship. Evan's program solves geometric analogy problems like:



A context and ambiguity problem arises when the program has to decide how to subdivide the complex figures and how to pair off their components. The program operates by constructing descriptions of the figures (e.g., X inside Y, Y next to Z) and of possible transformation (e.g., X is dilated, Y moved into Z, etc.). It picks the answer pair whose transformation admits a description most like a description of the transformation of the data-pair. All these "descriptions" are highly indeterminate and underdetermined.

Guzman's program analyzes a "scene" like that shown on the next page into separate objects. It makes use of local configurations like "T-joins" (where an edge runs into another like a T, although not necessarily at right angles). These are highly ambiguous taken individually--for example because the program uses no local information to know which of the three regions that meet at a T-join are "figure" and which, if any, are "background". Its cleverness is its combination of local features to make a global interpretation of the whole as a complex of three dimensional objects.


40
20
19
6
7
8
9
17
18
15
14
13
12
11
16
10
22
5
39
25
23
38
24
4
37
24
3
2
1
33
32
31
30
36
35
27
28
29
26

MOUTH

The program adds to the air of silliness of the next passage. Dreyfus is discussing the outline figure of a transparent cube.

> "But the machine could not interpret the figure three-dimensionally as a cube first in one, then in the other of these orientations. Such an interpretation would require the machine to focus on certain aspects of the figure while leaving others in the background, and the machine lacks precisely this figure-ground form of representation. For it, every point of the figure is equally explicit; thus the figure can only be interpreted as an ambiguous flat projection. To say that now one, now the other orientation was being presented would make no sense in such a program, although this alternation of perspectives could easily affect human behavior. Such phenomena challenge the possibility of totally formalizing human information processing." (Alchemy, p. 58)

Guzman's program could be used very trivially to do what Dreyfus innocently declares to be impossible.

A theoretical study by Minsky and Papert makes a detailed mathematical analysis of simple algorithms for combining local measures to make decisions about global figures. Rigorous impossibility proofs in certain cases and surprising positive results in others show that one cannot trust an uninformed intuition about what can and cannot be done in this area.

None of these examples are mentioned here to impress readers with the degree of intelligence of any program. I have already remarked that this is not the purpose of experiments in Artificial Intelligence. Their proper use is the one I have just made: as material for discussion of particular techniques and theoretical issues.

2.2.5 UHFIPs.

If the Muller-Lyer Illusion and the ability to use underdetermined expectations of a whole to determine elements of the whole are evidence for UHFIPs, then machines have them. But the examples taken up to now from Dreyfus are very abstract. I turn next to one of his prime examples of a real human ability. In this case I shall not be able to show that the machine can do what he says requires a UHFIP. But his attempt to argue that it cannot, reveals some new aspects of his style of analysis.

2.2.6 Wittgenstein and the Inexactly Similar Brothers.

Dreyfus gives as an example of the uniquely human our ability to recognize family resemblances between objects with no "exactly similar trait" in common. I shall begin by taking "family resemblance" quite literally as physical similarity between people of common ancestry. This is obviously a special case: but what is true in general should also be true in particular. In 2.2.8 I shall take the wider interpretation.

2.2.7 Digression: Some Background Material.

It will help the discussion to describe the kind of program that might be used in studies on facial recognition. The simplest associates with the face a number obtained by combination of local measures such as distance between eyes, length of nose made on a two-dimensional projection of the face (e.g., a photograph). Experiments with programs of this sort have not yielded results good enough for practical application, say to identifying a wanted criminal. Analysis of the errors made by the program shows that it is often led astray by confining attention to "flat" properties and ignoring three-dimensional information such as "protruding chin" or "sunken cheeks" or "bulging forehead." To appreciate how difficult this is, one has to recall seeing photographs taken under very uniform illumination, so that shadows and shading do not appear. The photograph appears so flat and characterless that even humans cannot see much in it. However, there is no reason to confine the computer to "flat" properties. It can use stereoscopic vision or take account of shading to construct a three-dimensional representation of the face. Prof. Bledsoe at the University of Texas has begun to build programs to do this. It is too soon to predict

exactly how well the program will perform. I mention it only to illustrate the open-ended approach of research in Artificial Intelligence: it is quite clear that Bledsoe's new program will be better than the earlier ones: and if it is still not good enough, new programs will be given more knowledge of a yet more structured kind. For example a preliminary phase of the program might make hypotheses about the expression of the face. This could enter the total program in many ways. The simplest is to refrain from using a measure of lip protrusion to compare a pouting man with a smiling photograph. More subtly the program could use the kind of smile as a property for classification of the face.

Now to compare the potential performance of programs with the actual performance of people we need to make an informed guess about how much improvement is likely in programs and to know something about how well people are able to recognize faces or classify them by "family resemblance." I do not have much information about studies of human performance. The ones I do know indicate that our subjective impression of responding to the "whole face" is misleading: classification of errors shows that untrained people often have a predominant preference for certain features. Moreover, training people to look systematically at specific features seems to improve their ability at recognition. But, as I have said, I do not know very much about the topic and will not rely on these points of fact in my discussion. I mention them as methodological examples: one ought to consider that kind of hypothesis before holding strong views about what people can and cannot do. Subjective impressions are sometimes wrong.

2.2.6 Wittgenstein and the Inexactly Similar Brothers (resumed).

I must ask readers to bear with me by reading a passage from Dreyfus very carefully. They will recall having seen the first sentence.

> "It is hard to imagine how a computer, which must operate on completely determinate data according to strictly defined rules, could be programmed to use an underdetermined espectation of the whole in order to determine the elements of that whole. But workers in AI might answer that, even if people do use some sort of holistic approach based on context which no one now knows how to program, there is no reason in principle why some alternative approach could not be discovered which would do the same job. One could, for example, deal more efficiently with a large number of specific traits, or one could develop a sort of anticipation which on the basis of certain traits in the context would assign an object to a class defined in terms of a large number of traits, which would then serve as hypotheses. This answer however, ignores a unique feature of human pattern recognition: our ability to recognize family resemblances where, as Wittgenstein poinyd out, two individuals recognized as belonging to the same family need have no exactly similar traits in common. We can nonetheless recognize such similarities by picking out a typical case and introducing intermediate cases. This use of paradigms and context rather than class definitions allows our recognition of patterns to be open-textured in a way which is impossible for any recognition based on a specific list of traits." (Bodies, p. 19)

Well, is he saying anything?

What is an "exactly similar trait"? If two brothers have, as one says, their father's big nose, does this count as an exactly similar trait in common? Must the noses be exactly similarly big? If so it is obviously true that family resemblances cannot be judged by "exactly similar traits"--even two views of the same person would not have any in common. But then not even the crudest form of program is fairly described as judging by "exactly similar traits." To give Dreyfus' statement a semblance of sense we must interpret "exactly similar" more loosely, e.g., as meaning "inexactly similar." This might allow the program to recognize that both

faces have larger than average noses or even to correlate quantitative measures. However, in this case how does Dreyfus know that it will not perform better than people?

And even if we grant that he could know this, we may still ask why a program could not follow his description of the human method by using intermediate cases? The analogy program by Evans mentioned earlier finds sets of transformations that take one given figure into another. Why does Dreyfus think we cannot extend this idea to transforming faces into one another?

There is an almost serious argument Dreyfus might use. If the program transformed the faces it would do so by applying elementary operations drawn from a fixed set: "enlarging the nose" or "flattening the cheeks" and so on. On the other hand, he might say, people are free to find intermediate cases along <u>any</u> dimension, with completely unrestrained liberty.

Before answering this argument I recall that we are discussing whether a program "would do the same job," possibly using "some alternative approach" quite different from the "holistic" one used by people. We are <u>not</u> discussing what kind of program would use the "same approach" as a man.

Now I see no reason to suppose that people really are free to use more kinds of transformation than the machine. Admittedly they are free in the logical sense that makes it not quite self-contradictory to imagine my Aunt Agatha using a Fourier transform. But people usually do, as Lady Lovelace might have said, what other people have done before them.

But even if people <u>are</u> freer than the machine, it does not follow that they will do the job better. When Dreyfus played chess against Mack-Hack 6 he was free to do many, many things the machine had never dreamed of. But

to no avail. The machine did the job of playing chess better than Dreyfus' UHFIPs. It is irrelevant to this point that some people are stronger players than the machine. The incident is merely a counter-example to the suggestion that consideration of human open-ended liberty can be sufficient justification for assertions about human superiority on the job. But it hardly needs a counter-example.

I contend that Dreyfus has not said and cannot say, what "unique feature of human pattern recognition" is ignored by the "workers in AI." Furthermore he has not said why features of human pattern recognition should not be ignored. He has not even described in any vaguely intelligible language the activity he believes people can but machines cannot perform. He has certainly given no reason to support the belief that there are any.

2.2.8 Can the Behavior of Computers Be Formalized?

The more abstract and less literal sense of family resemblance brings to mind puzzles about the meanings of complex words like "game". In this section I shall digress again to indicate how easy and how misleading it is to say that these "meanings" cannot be "formalized" or "programmed." To make the connections between these ideas clearer I note that in Alchemy Dreyfus elaborates his problem about family resemblances and words like "game" by declaring that the use of these words is not "rule-like." The intelligent activity involved in their use is placed in the area of "non-formalizable" activities, which are declared to be the most intractable of all to programming. The following paragraphs will examine Dreyfus'

reasons for believing this. I emphasize once more that I shall discuss his explanations of the difficulty he claims to see. I am not saying that there are no difficulties, but that they are not what he says they are.

It is well-known that dictionary definitions are inadequate for words like "game." Webster says "contest according to set rules undertaken for amusement or for a stake." But solitaire is usually called a game, though there is no contest, and aerobatic flying usually is not called a game, though there is contest. People sometimes go on from this observation to say: games have no definite set of characteristics in common--rather, they have a family resemblance to one another. In some contexts saying this would serve a definite purpose. For example one might use it to bring home the logical facts of life to someone who insisted on being given a definition of "game" in a stereotyped form such as a list of conditions formulated by combining some set of words like "team," "competitive," "fun," etc. In other contexts it would merely invite the comment: of course they have characteristics in common--"they are all games" or "they are listed in my excellent encyclopedia" or "after a little thought one sees that they are games." My first complaint is that Dreyfus' discussion is too vague to decide clearly what reply would be relevant. HOwever, I shall continue.

Questions like "how do people decide what is a game" easily generate an air of mystery. But this is not simply or mainly because we do not have complete answers. The sense of mystery arises when all hypothetical answers are unacceptable. And this will often happen if the only answer is so long and complicated that the questioner will not stay to listen or worse yet, will not accept as an answer so complex a story or that kind of story. To illustrate the point I shall consider a different question. A young lady who knew no programming asked me how the robot that cannot

play ping-pong recognizes the building blocks out of which it can make a tower. Some people will accept sketchy answers like "it looks for the edges of the cube and computes the position by projective geometry." This girl really wanted to know. No answer like "all parallelipipeds have the exactly similar trait of nine visible edges from the most frequent point of view" would do. This is part of the story; but only part, for example because if the program sees only six it might still recognize the block, or if it sees four it might look again. But this is vague: when does it look again? What does "look" mean? What happens if the block is seen end on?

The young lady never understood. She decided it was beyond her. The point of the anecdote is that she did not decide that the criteria used by the computer cannot be formalized or that the behavior of the machine can be understood only by postulating a special uniquely mechanical form of pattern recognition for the resemblances of blocks. I suspect that she might have made the opposite decision if the discussion had been about how people recognize blocks or games. I suspect she might have done so even if I understood how people do this as well as I understand how the machine operates.

The joke is that in a certain important sense the criteria used by the computer really cannot be formalized. Is the program a formalization of the criteria? In some senses yes, in some no. An important negative case is exactly the sense in which people sometimes say that a man's criterion for "being a game" is not formalizable. When we ask for a formalization of "being a game" we are not asking for a description of neural events--but for a description of a relation between characteristics of games,

non-games, players, prizes, etc. But the program of the machine is more analagous to the neural description; its elements look like "MOVEM FOO BAR" and nothing like "if six edges are seen then ...".

Look at the situation more concretely through a gedanken experiment. The computer is in front of us busily sorting cubes into two stacks. We have no copy of the program. The machine is sealed. We have to "formalize" its criterion. Surely the outcome will be this: we will easily find approximate descriptions but perhaps never understand exactly how the poor machine decides what is a cube and which goes on the left. How easy it is to imagine an observer who did not know what was in a computer exclaiming: "computers cannot be simulated by programs."

2.2.9 Family Resemblances (concluded).

What does Dreyfus' theory of human recognition of family resemblances say about the family of games?

At best he has observed that the human use of the word "game" is not easily formalized by constructing a set of rules of the kind he believes to be programmable. I wonder how much time he devoted to looking for a formalization: it took some centuries for mathematicians at least as clever as Dreyfus to find a good formalization of integration. But let us grant that he tried hard enough to be sure that there is no formalization of the kind he was seeking. The next question is whether the impossibility of this kind of formalization implies that programs cannot imitate the use of the word "game" to within the range of variation observed amongst people?

Far from having an answer, he has not even begun to see the problems he must face in giving one. Can he define his concept of "formalizable"

sharply enough to ensure that the activities of computers are but those of people are not? To do this he would need a better characterization of what computers can do than saying that they use lists of traits. What is a trait? In what ways can they be "used"? How far towards human ability does he think computers can go? All the questions I asked about chess and computability are equally appropriate here. The difference is that "completely computable" at least has a meaning, while the concepts "family resemblance," "formalization" and "rule-like" are left so vague, so unformalized, so un-rule-like that this alone would be enough to conclude that Dreyfus has not even stated a case to prove.

3.0 Liars?

Dreyfus' mission does not end with showing that people associated with Artificial Intelligence are wrong or even foolish. He is called to expose them as obscurantists and liars. Many of the charges he makes are too generalized and vaguely formulated to be answerable. But others are directed clearly at individuals and particularly at Prof. H. A. Simon.

3.1 Did Simon Hide the Unexpected Difficulties?

The background to the charge against Simon is his well-known prediction of 1957 that a program would be the world's chess champion within ten years. Dreyfus does not directly criticise Simon for this optimism. His major charge is that instead of acknowledging the unexpected difficulties encountered by subsequent attempts to make chess machines and describing the low level of their play, Simon wrote an article in 1962 which "gives the impression that the chess prediction is almost realized."

I cannot understand what possessed Dreyfus to say this. I have read the article (which I shall call S) many times and find no such claim. Indeed as I shall explain in a moment, it does not directly discuss chess programs at all. Of course, I cannot argue with Dreyfus about the subjective question of what Simon's words might have "suggested" to him. The really astonishing thing is that when, in a later chapter of Alchemy, he wishes to cite authority for the poor level of chess programs, he quotes "Newell, Shaw and Simon themselves,"[1] referring to an article published four months after S. As he says, this article (dated February 1963) gives

---

1. Alchemy, p. 10.

a very grim description of the "mediocre" level of machine chess. Although its authors remain optimistic, nothing could serve better than their sober statements to disillusion anyone who imagined that programs were already half-way to the world championship. But, if we turn back four pages in Alchemy to the chapter on the charges of misrepresentation, we read

> "While their program was losing its five or six poor games--and the myth they had engendered was holding its own against masters in the middle game[1]--Newell, Shaw and Simon kept silent. When they speak again, three years later (i.e., in S, October 1962--S.P.) they do not report their difficulties and disappointments." (Alchemy, p. 6)

Since the critical paper was to appear four months later, the allegation that Simon was not frank about his difficulties would be unreasonable even if S were an article about programming machines to play chess. To berate Simon for not mentioning these difficulties in an article on a different subject verges on irresponsibility.

3.2 Did Simon Imply That the Prediction Was Almost Realized?

Dreyfus makes the following statement about the article, S:

> "... as if to take up where the myth had left off, Simon published an article in Behavioral Science announcing a program which will play 'highly creative' chess end games ...it is misleadingly implied that similar simple heuristics would account for master play even in the middle game... Thus, the article gives the impression that the chess prediction is almost realised. With such progress, the chess championship may be claimed at any moment." (Alchemy, p. 7)

I begin by dissipating an ambiguity in the words: "announcing a program which will play ..." The article is not about computers, but about experiments on the use of a set of rules to guide human subjects to find chess combinations. This set of rules, Simon explains, is called a "pro-

---

1. The sarcasm refers to the statement by Norbert Wiener discussed in the next section.

gram" by analogy with the use of this word in computation. Thus the article does not "announce a program" in the literal sense of these words.

The article presents evidence to show that by following the rules one can indeed find the correct moves in situations taken from chapters on combination play in books on chess. These facts do, of course, give encouragement for the possibility of mechanical chess by indicating that in some situations that seem difficult to people, a set of rules can lead to the proper move. In this sense the experiment could be taken as a partial confirmation of the theoretical ideas on which Simon's original optimism was based. But confirming the theory is a very far cry from making a champion chess machine. The actual rules used in the experiment could, of course, be incorporated in the machine's program. But they would be a very small component of it.

In short, no one with a reasonable idea of what is involved in making a chess program could possibly read Simon's article as suggesting that a champion program was very near. Simon certainly does not say it is.

A minor point remains for discussion. Dreyfus accuses Simon of "misleadingly implying" that similar rules could be used in "middle games." I am not sure what the charge is. First, the situation is confused by the fact that the whole experiment was on middle-game mating combinations. I have no idea what made Dreyfus think it had to do with end games--unless he thinks that a mating combination is an end game because it ends the game. Second, I am not sure what "misleadingly implies" can mean. Simon does conjecture that similar rules exist for a wider class of situations. But all the facts are openly and plainly presented. Simon clearly sets out his analysis and his reasons: his writing is a model of honest clarity.

The onus is on Dreyfus to give a reason to call it "misleading." He gives only one clue:

> "That the program restricts these end games to dependence on continuing checks ... is mentioned but not emphasized." (Alchemy, p. 7)

Of course Simon mentions nothing about end games. The role of checks is "mentioned" with perfect clarity at four separate places in the article!

3.3 Norbert Wiener Exaggerates.

The specific charge against Simon is part of a general tirade alleging a tendency to exaggerate the achievement of Artificial Intelligence by making boastful claims but never mentioning the difficulties and obstacles. "Fact" says Dreyfus, "had ceased to be relevant." The chess machine was in "the realm of scientific mythology." (Alchemy, p. 6). This is illustrated by accusing Norbert Wiener of saying:[1]

> " 'chess-playing machines as of now will counter the moves of a master game with the moves recognized as right in the text books, up to some point in the middle game.' " (Phenomenology, p. 33)

Dreyfus does not say that Wiener's very next sentence was:

> "It is true that when they go wrong they will go very wrong and commit absurdities." ("The Brain and the Machine" in Dimensions of Mind, S. Hook, ed., p. 110)

But even this omission is bland compared with his failure to say that Wiener's statement, placed in its context, is about the contrast between the strength of checkers programs "whose plays up to the end game are already recognized to be better than those of a checker master" and the relative weakness of chess programs whose play is good only at the beginning of the game.

The incident has no importance since Wiener is not a leader in programmed Artificial Intelligence. But it is flavorful and revealing.

---

1. Dreyfus' indignation might be conditioned by the popular impression that programs should be strongest in end games Not so. Their forte is tactical play in the middle game.

3.4 The Pons Asinorum.

Simon also predicted that computers would prove important new mathematical theorems. Dreyfus again alleges that false claims have been made about progress toward this goal. In Phenomenology he says

> "We do not have time to go into the deliberate confusions surrounding the supposed proof of an important theorem." (p. 32)

This is a strong charge. But at whom is it aimed? Aspersions made in this way against unnamed individuals show dubious taste and dubious justice. In Alchemy he makes a more specific charge:

> "Recent publications suggest that the first of Simon's forecasts has already been half-realized and that considerable progress has been made in fulfilling his second prediction." (Alchemy, p. 4)

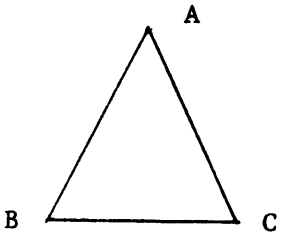
This is followed by a statement that Ross Ashby said a computer had found a "bewilderingly simple" proof of the pons asinorum (i.e., base angles of an isosceles triangle are equal). But this is a very mild claim indeed and carries no implication that computers are anywhere near discovering important new theorems.[1] Moreover, careful reading of Ross Ashby's article reveals no hint of such a suggestion.

Some history will help readers see how Dreyfus transmutes Ashby's simple metal to a shining example of deliberate confusion.

---

1. Whether the pons is important is a matter of mathematical taste.

The relevant history began at a Summer Program on Artificial Intelligence held at Dartmouth in 1956. At this meeting, Marvin Minsky proposed a simple set of heuristic rules for a program to prove theorems in Euclidian Geometry. Before any program had been written these rules were tried (by "hand simulation") on the simple theorem (sometimes called the "pons asinorum") which asserts that the base angles of an isosceles triangle are equal. To everyone's surprise and pleasure these rules led to an elegant proof quite different from the one normally taught in high school courses. Instead of constructing the perendicular bisector of the base, this proof proceeded by observing that the given triangle could be represented as ABC and as ACB. Since (AB, $\hat{A}$, AC)



matches (AC, $\hat{A}$, AB), the triangles ABC and ACB are congruent. Hence B = C !

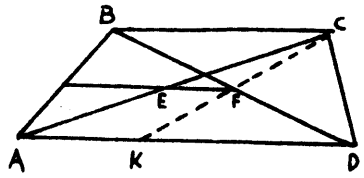
Although not entirely new, this proof is not well-known and strikes anyone with mathematical taste as extremely elegant. So it is not surprising that the story spread and very soon transformed itself, as stories do, from "Minsky's rules suggested ..." to "a program generated a new proof" to "a computer invented an elegant new proof."

In the meantime, H. Gelernter, justifiably encouraged by this little success, used Minsky's heuristics in a program which, by 1958, was proving

theorems with a facility some high school students could justly envy. A typical example of the kind of problem studied by Gelernter is:

> If the segment joining the midpoints (E,F) of the diagnals of a trapezoid (ABCD) is extended to intersect a side of the trapezoid, it bisects that side.

To test the difficulty of the problem, readers might like to prove it, as did the program, with the following hint: construct the line CFK.



We now return to Dreyfus. We see another example of text in which he attempts to foist on us the same triad of elements that constitute his impression of the chess episode: a prediction by Simon--that machines would prove original mathematical theorems; by contrast, a description of an allegedly meagre achievement; and a pitiful image of Artificial Intelligence experts blowing the trumpets of victory to hide shameful defeat. A new element is Dreyfus' need to invent the experts. If, as Dreyfus claims, progress is being oversold, he would surely be able to find a statement by Simon or Newell or Minsky or McCarthy or some other well-known leading figure in Artificial Intelligence. But he cannot. The best he can offer is a hair-splitting quibble about a remark made by a man who made fundamental contributions to the logical foundations of cybernetics but has not participated directly in work on the use of computers. This does not inhibit Dreyfus from calling this man a "leading authority on Artificial Intelligence."[1] Dreyfus writes:

---

1. It is fair to say the same also about Wiener.

> "In a review of Feigenbaum and Feldman's anthology, Computers and Thought, W. R. Ashby (one of the leading authorities in the field) hailed the mathematical power of the properly programmed computer: 'Gelernter's theorem-proving program has discovered a new proof of the pons asinorum that demands no construction.' This proof, Professor Ashby goes on to say, is one which 'the greatest mathematicians of 2000 years have failed to notice ... which would have evoked the highest praise had it occurred.'" (Alchemy, p. 4)

He goes on to complain:

> "The theorem sounds important and the naive reader cannot help sharing Ashby's enthusiasm. A little research, however, reveals that the pons asinorum, or ass's bridge, is the first theorem to be proved in Euclidian geometry, viz., that the opposite angles of an isosceles triangle are equal ... The first announcement of the 'new' proof 'discovered' by the machine is attributed to Pappus (300 A.D.) [37:284]. There is a striking disparity between Ashby's excitement and the antiquity and triviality of this proof." (Alchemy, p. 5)

This short passage contains a truly remarkable number of intellectual atrocities.

(1) To begin with Dreyfus' tendency to distort quotations shows itself almost as directly as in the quotation from Wiener. The real quotation from Ashby reads:

> "Similarly Gelernter's theorem-proving program has discovered a new proof of the Pons Asinorum that demands no construction and uses only a direct application of Euclid's immediately preceding Proposition. The proof is bewilderingly ingenious yet so simple that it can be written in about three lines! The fact that the greatest mathematicians of two thousand years have failed to notice this proof (which would have evoked the highest praise had it occurred) must be regarded as settling forever the question whether a machine can really produce something new." (My emphasis)

The suppressed sentence is critically important in judging the charge made by Dreyfus against Ashby. Dreyfus claims that Ashby leads the "naive reader" to believe that the machine proved a very complicated theorem; but

Ashby says, as plainly as anyone can, that the proof was very, very simple. Indeed, for Ashby, the virtue of the proof is precisely its simplicity ... which Dreyfus, in his innocence of mathematical taste, sees as triviality.

(2) In any case, it is ludicrous to allege that Ashby misleadingly exaggerates the importance of the theorem. He calls it by its common name--and as an educated Englishman he would naturally assume that readers know the meaning of "pons asinorum." That "naive" readers might be misled is undeniable. But scientific writing is not addressed to the ignorant and the naive (even if they are professors of philosophy).

(3) A more serious demonstration of Dreyfus' poor scholarship is his failure to observe that Ashby is misrepresenting Gelernter. Dreyfus quotes Gelernter's papers in his bibliography to Alchemy and discusses them as if he had not only read them, but understood them well enough to pass judgment on their value. It is, therefore, to say the least, remarkable that he does not notice that Gelernter makes no mention of the pons asinorum! It is of questionable honesty to let readers think that Gelernter's program had proved a very simple theorem (which is not even mentioned by Gelernter) while not mentioning the more complex theorems it did prove. If Dreyfus were acquainted with the literature in this area he might have noticed Minsky's account of the incident in Mechanization of Thought Processes.