Man Viewed as a Machine

"Muscle" and "brain" machines do much of his daily work. Now he conceives a machine that will reproduce itself. This once again brings up the question of whether man himself is only a machine

by John G. Kemeny

Is man no more than a machine? The question is often debated these days, usually with more vigor than precision. More than most arguments, this one tends to bog down in definition troubles. What is a machine? And what do we mean by "no more than"? If we define "machine" broadly enough, everything is a machine; and if by "more than" we mean that we are human, then machines are clearly less than we are.

In this article we shall frame the question more modestly. Let us ask: What could a machine do as well or better than a man, now or in the future? We shall not concern ourselves with whether a machine could write sonnets or fall in love. Nor shall we waste time laboring the obvious fact that when it comes to muscle, machines are far superior to men. What concerns us here is man as a brain-machine. John von Neumann, the mathematician and designer of computers, not long ago made a detailed comparison of human and mechanical brains in a series of lectures at Princeton University. Much of what follows is based on that discussion.

We are often presented with Utopias in which all the hard work is done by machines and we merely push buttons. This may sound like a lazy dream of heaven, but actually man is even lazier than that. He is no sooner presented with this Utopia than he asks: "Couldn't I build a machine to push the buttons for me?" And indeed he began to invent such machines as early as the 18th century. The flyball governor on a steam engine and the thermostat are elementary brain-machines. They control muscle machines, while spending only negligible amounts of energy themselves. Norbert Wiener has compared them to the human nervous system.

Consider the progress of the door. Its earliest form must have been a rock rolled in front of a cave entrance. This may have provided excellent protection, but it must also have made the operation of going in and out of the cave quite difficult. Slowly, as man found better means of defending himself, he made lighter and more manageable doors, until today it is literally child's play to open a door. But even this does not satisfy us. To the delight of millions of railroad passengers, the Pennsylvania Railroad installed electric eyes in its New York terminal. Man need only break the invisible signal connecting the two photoelectric "eyes," and immediately the little brain-machine commands the door to open. This control device needs only a negligible amount of energy, is highly efficient and is vastly faster than any doorman.

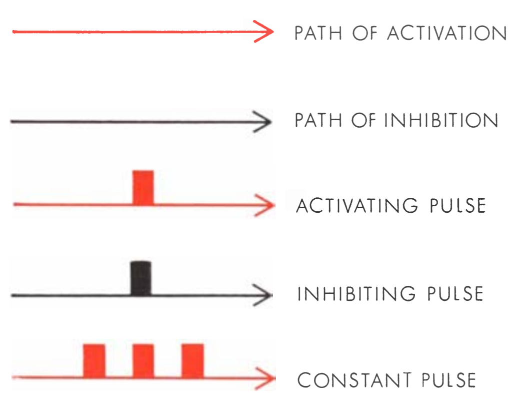
The central switchboard in an office is another brain-machine, especially if the office has installed the dial system. Messages are carried swiftly and efficiently to hundreds of terminals, at the expense of only a small quantity of electricity. This is one of those brain-machines without which modern life is supposed to be not worth living.

And, finally, there is the example most of us are likely to think of when brain-machines are mentioned: namely, the high-speed computer. Electric eyes and telephone exchanges only relieve us of physical labor, but the calculators can take the place of several human brains.

The Slow Brain

In economy of energy the human brain certainly is still far ahead of all its mechanical rivals. The entire brain with its many billions of cells functions on less than 100 watts. Even with the most efficient present substitute for a brain cell—the transistor—a machine containing as many cells as the brain would need about 100 million watts. We are ahead by a factor of at least a million. But von Neumann has calculated that in theory cells could be 10 billion times more efficient in the use of energy than the brain cells actually are. Thus there seems to be no technical reason why mechanical brains should not become more efficient energy-users than their human cousins. After all, just recently by inventing the transistor, which requires only about a hundredth of a watt, we have improved the efficiency of our machines by a factor of 100; in view of this the factor of a million should not frighten us.

While we are still ahead in the use of energy, we are certainly far behind in speed. Whereas a nerve cannot be used more than 100 times a second, a vacuum tube can easily be turned on and off a million times a second. It could be made to work even faster, but this would not contribute much to speeding



SIMPLE CIRCUITS for a brain machine are depicted on the next five pages. The activating and inhibiting pulses are identical but follow different paths.

up the mechanical brain at the moment. No machine is faster than its slowest part, so we must evaluate various components of the machine.

In a calculating machine four different problems confront the designer: the actual computations, the "logical control," the memory and the feeding of information to the machine and getting answers out. Speed of computation, a bottleneck in mechanical computers such as the desk calculator, has been taken care of by the vacuum tube. The next bottleneck was the logical control—the system for telling the machine what to do next after each step. The early IBM punch-card machine took this function out of the hands of a human operator by using a wiring setup on a central board which commanded the sequence of operations. This is perfectly all right as long as the machine has to perform only one type of operation. But if the sequence has to be changed frequently, the wiring of the board becomes very clumsy indeed. To improve speed the machine must be given an internal logical control. Perhaps the greatest step forward on this problem has been accomplished by MANIAC, built at the Institute for Advanced Study in Princeton. This machine can change instructions as quickly as it completes calculations, so that it can operate as fast as its vacuum tubes will allow.

That still leaves the problems of speeding up the memory and the input and output of information. The two problems are closely related. The larger the memory, the less often the operator has to feed the machine information. But the very fact that the machine performs large numbers of computations between instructions clogs its memory and slows it down. This is because an accumulation of rounding errors makes it necessary to carry out all figures in a calculation to a great number of digits. In each computation the machine necessarily rounds off the last digit; in succeeding operations the digit becomes less and less precise. If the computations are continued, the next-to-last digit begins to be affected, and so on. It can be shown that after 100 computations the last digit is worthless; after 10,000 the last two digits; after 1,000,000, the last three. In the large new computers an answer might easily contain four worthless figures. Hence to insure accuracy the machine must carry more digits than are actually significant; it is not uncommon to carry from eight to 12 digits for each number throughout the calculation. When the machine operates on the binary system of numbers, instead of the decimal system, the situation is even worse, for it takes about three times as many digits to express a number in the binary scale.

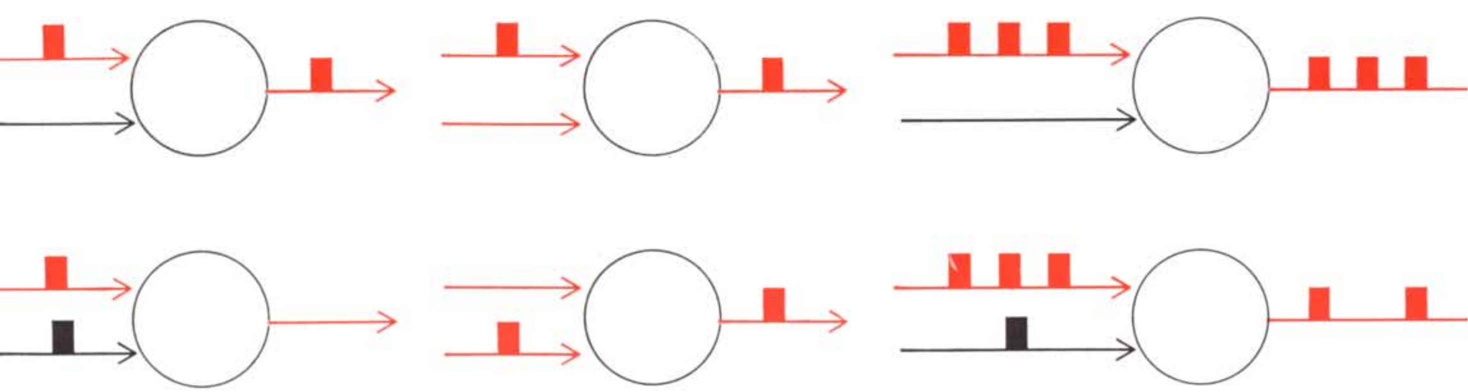
MANIAC uses up to 40 binary digits to express a number. Due to the necessity for carrying this large number of digits, even MANIAC's celebrated memory can hold no more than about 1,000 numbers. It has an "external memory," in the form of a magnetic tape and magnetic drums, in which it can store more information, but reading from the tape or drums is a much slower operation than doing electrical computations.

In spite of the present limitations, the machines already are ahead of the human brain in speed by a factor of at least 10,000—usually a great deal more than 10,000. They are most impressive on tasks such as arise in astronomy or ballistics. It would be child's play for MANIAC to figure out the position of the planets for the next million years.

Still we are left with the feeling that there are many things we can do that a machine cannot do. The brain has more than 10 billion cells, while a computer has only a few tens of thousands of parts. Even with transistors, which overcome the cost and space problems, the difficulty of construction will hardly allow more than a million parts to a machine. So we can safely say that the human brain for a long time to come will be about 10,000 times more complex than the most complicated machine. And it is well known that an increase of parts by a factor of 10 can bring about differences in kind. For example, if we have a unit that can do addition and multiplication, by combining a few such units with a logical control mechanism we can do subtraction, division, raising to powers, interpolation and many other operations qualitatively different from the original.

The Complex Memory

Part of man's superior complexity is his remarkable memory. How does MANIAC's memory compare with it? For simplicity's sake let us measure the information a memory may hold in "bits" (for binary digits). A vacuum tube can hold one digit of a binary number (the digit is 1 if the tube is on, 0 if it is off). In vacuum-tube language it takes 1,500 bits to express the multiplication table. Now MANIAC's memory holds about 40,000 bits, not in 40,000 separate tubes but as spots on 40 special picture tubes, each of which can hold about 1,000 spots (light or dark). Estimates as to how much the human memory holds vary widely, but we certainly can say conservatively that the brain can remember at least 1,000 items as complex



BASIC CELL (*circle*) will fire when it is activated (*top*). It will not fire, however, if it is inhibited at the same time (*bottom*).

"OR" CIRCUIT requires two paths of activation. The cell fires if a pulse arrives on one path (*top*) "or" the other (*bottom*).

"NOT" CIRCUIT incorporates constant activating pulses. The cell can thus fire constantly (*top*). It signals "not" when it is briefly inhibited (*bottom*).

as the multiplication table (1.5 million bits), and a reasonable guess is that its capacity is closer to 100 million bits—which amounts to acquiring one bit per 20 seconds throughout life. So our memory exceeds that of MANIAC by a factor of 1,000 at least.

Is the difference just a matter of complexity? No, the fact is that machines have not yet imitated the human brain's method of storing and recovering information. For instance, if we tried to increase MANIAC's memory by any considerable amount, we would soon find it almost impossible to extract information. We would have to use a complex system of coding to enable the machine to hunt up a given item of information, and this coding would load down the memory further and make the logical control more complex. Only when we acquire a better understanding of the brain's amazing ability to call forth information will we be able to give a machine anything more than a limited memory.

The Logical Machine

Let us now consider the inevitable question: Can a machine "think"? We start with a simple model of the nervous system such as has been constructed by Walter Pitts and Warren S. McCulloch of the Massachusetts Institute of Technology. Its basic unit is the neuron—a cell that can be made to emit pulses of energy. The firing of one neuron may activate the next or it may inhibit it. The neurons are assumed to work in cycles. This corresponds to our knowledge that after firing a neuron must be inactive for a period. To simplify the model it is assumed that the various neurons' cycles are synchronized, *i.e.*, all the neurons active during a given period fire at the same time. For a given neuron to fire in a given cycle two conditions must be satisfied: in the previous cycle it must have been (1) activated and (2) not inhibited. If, for example, a neuron has two others terminating in it of which one activates and one inhibits, and if the former fires in a given cycle and the latter does not, then the neuron will fire in the following cycle. Otherwise it will be inactive for a cycle.

Out of this basic pattern we can build the most complex logical machine. We can have a combination that will fire if a connected neuron did not fire (representing "not") or one that will fire if at least one of two incoming neurons fired (representing "or") or one that will fire only if both incoming neurons fired (representing "and"). Combining these, we can imitate many logical operations of the brain. The simple arrangement diagrammed on pages 62 and 63 will count up to four, and it is easy to see how to generalize this technique.

We can also construct a very primitive memory: *e.g.*, a system that will "remember" that it has been activated until it is instructed to "forget" it. But if it is to remember anything at all complex, it must have an unthinkably large number of neurons—another illustration of the fact that human memory acts on different principles from a machine.

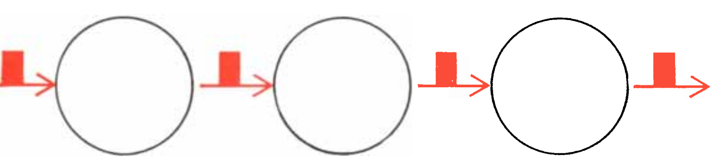
The Turing Machine

If we were to stop here, we might conclude that practical limitations of memory and complexity must forever restrict the cleverness or versatility of any machine. But we have not yet plumbed the full possibilities. The late A. M. Turing of England showed, by a brilliant analysis, that by combining a certain few simple operations in sufficient number a machine could perform feats of amazing complexity. Turing's machines may be clumsy and slow, but they present the clearest picture of what a machine can do.

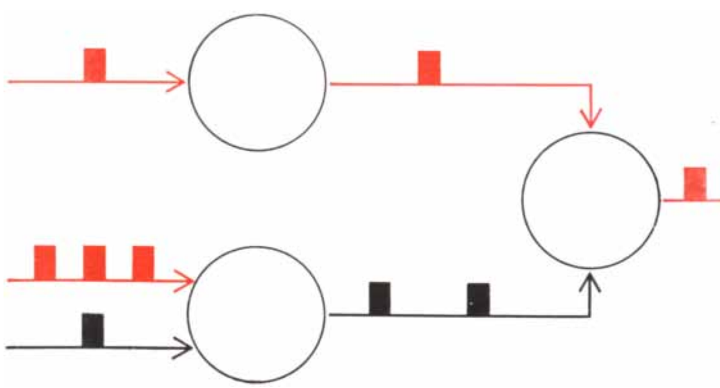
A Turing machine can be thought of as a mechanical calculator which literally works with pencil and paper. The paper it uses is a long tape divided into successive squares, and it operates on one square at a time. As it confronts a particular square it can do one of six things: (1) write down the letter X; (2) write down the digit 1; (3) erase either of these marks if it is already in the square; (4) move the tape one square to the left; (5) move the tape one square to the right; (6) stop.

Essentially this machine is a number writer. It writes its numbers in the simplest possible form, as a string of units. This is even simpler than the binary system. In the binary system the number 35, for example, is written 100011. In a Turing machine it is a string of 1's in 35 successive squares. The X's are merely punctuation marks to show where each number starts and ends.

The machine has the following parts: a device that writes or erases, a scanner, a motor to move the tape, a numbered dial with a pointer, and a logical control consisting of neuron-like elements, say vacuum tubes. The logical control operates from a prepared table of commands which specifies what the machine is to do in each given state. The



DELAY CIRCUIT is based on the fact that the basic cell receives a pulse in one "cycle" and fires it in the next. In this arrangement of three basic cells the pulse would be delayed three cycles.



"AND" CIRCUIT utilizes three cells. The first (*upper left*) has only an activating input. The second (*lower left*) has a constant activating input and an inhibiting input. The third (*right*) is the conventional basic cell. In the fir

state consists of two elements: what the scanner "sees" in the square before it, and where the pointer is on the dial. For example, the table of instructions may say that whenever the square has an X and the pointer is at the number 1 on the dial, the machine is to erase the X, and move the pointer to the number 2 on the dial. As the machine proceeds from step to step, the logical control gives it such commands, the command in each case depending both on the position of the dial and on what the scanner sees in the square confronting it. Observe that the dial functions as a primitive "memory," in the sense that its position at any stage is a consequence of what the scanner saw and where the pointer stood at the step immediately preceding. It carries over the machine's experience from step to step.

Turing's machine thus consists of a tape with X's and 1's in some of its squares, a dial-memory with a certain number of positions, and a logical control which instructs the machine what to do, according to what it sees and what its memory says. The diagram on pages 64 and 65 shows a very simple version of the machine, with a dial having only six positions. Since the scanner may see one of three things in a square—blank, 1 or X—the machine has 18 possible states, and the logical control has a command for each case [*see table at right of illustration*]. This machine is designed to perform a single task: it can add two numbers—any two numbers. Suppose it is to add 2 and 3. The numbers are written as strings of 1's with X's at the ends. Say we start with the dial at position 1 and the scanner looking at the second digit of the number 3 [*see diagram*]. The instructions in the table say that when it is in this state the machine is to move the tape one square to the right and keep the dial at position 1. This operation brings the square to the left, containing another digit 1, under the scanner. Again the instructions are the same: "Move the tape one square to the right and keep the dial at position 1." Now the scanner sees an X. The instructions, with the dial at position 1, are: "Erase (the X) and move the dial to position 2." The machine now confronts a blank square. The command becomes: "Move the tape one square to the right and keep the dial at position 2." In this manner the machine will eventually write two digit 1's next to the three at the right and end with the answer 5—a row of five digits enclosed by X's. When it finishes, an exclamation point signifies that it is to stop. The reader is advised to try adding two other numbers in the same fashion.

This surely is a cumbersome method of adding. However, the machine becomes more impressive when it is expanded so that it can solve a problem such as the following: "Multiply the number you are looking at by two and take the cube root of the answer if the fifth number to the left is less than 150." By adding positions to the dial and enlarging the table of instructions we can endow such a machine with the ability to carry out the most complex tasks, though each operational step is very simple. The Turing machine in fact resembles a model of the human nervous system, which can be thought of as having a dial with very many positions and combining many simple acts to accomplish the enormous number of tasks a human being is capable of.

Turing gave his machines an infinite memory. Of course the dial can have only a finite number of positions, but he allowed the machine a tape infinite in length, endless in both directions. Actually the tape does not have to be infinite—just long enough for the task. We may provide for all emergencies by allowing the machine to ask for more tape if it needs it. The human memory is infinite in the same sense: we can always make more paper to make notes on.

The Universal Machine

If we allow the unlimited tape, the Turing idea astounds us further with a universal machine. Not only can we build a machine for each task, but we can design a single machine that is versatile enough to accomplish all these tasks! We must try to understand how this is done, because it will give us the key to our whole problem.

The secret of the universal machine is that it can imitate. Suppose we build a highly complex machine for a difficult task. If we then supply the universal machine with a description of the task and of our special machine, it will figure out how to perform the task. It proceeds very simply, deducing from what it knows about our machine just what it would do at each step. Of course this slows the universal machine down considerably. Between any two steps it must carry out a long argument to analyze what our machine would do. But we care only about its ability to succeed, not its speed. There is no doubt about it: anything any logical machine can do can be done by this single mechanism.

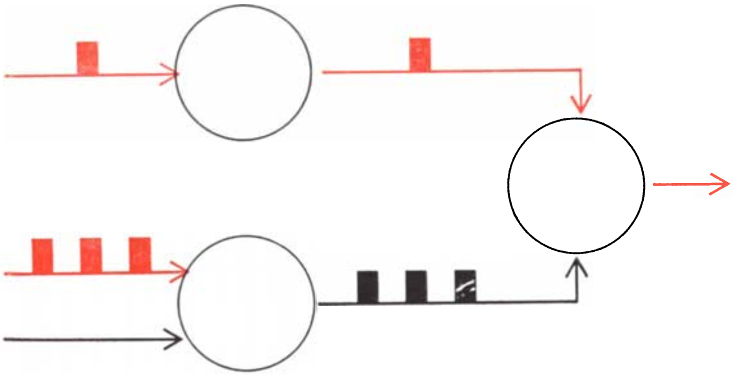
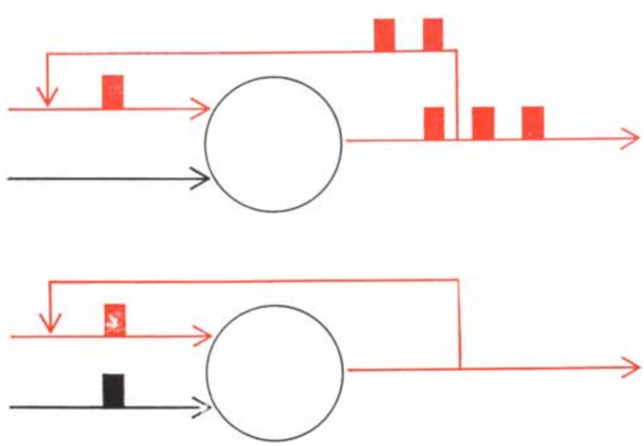
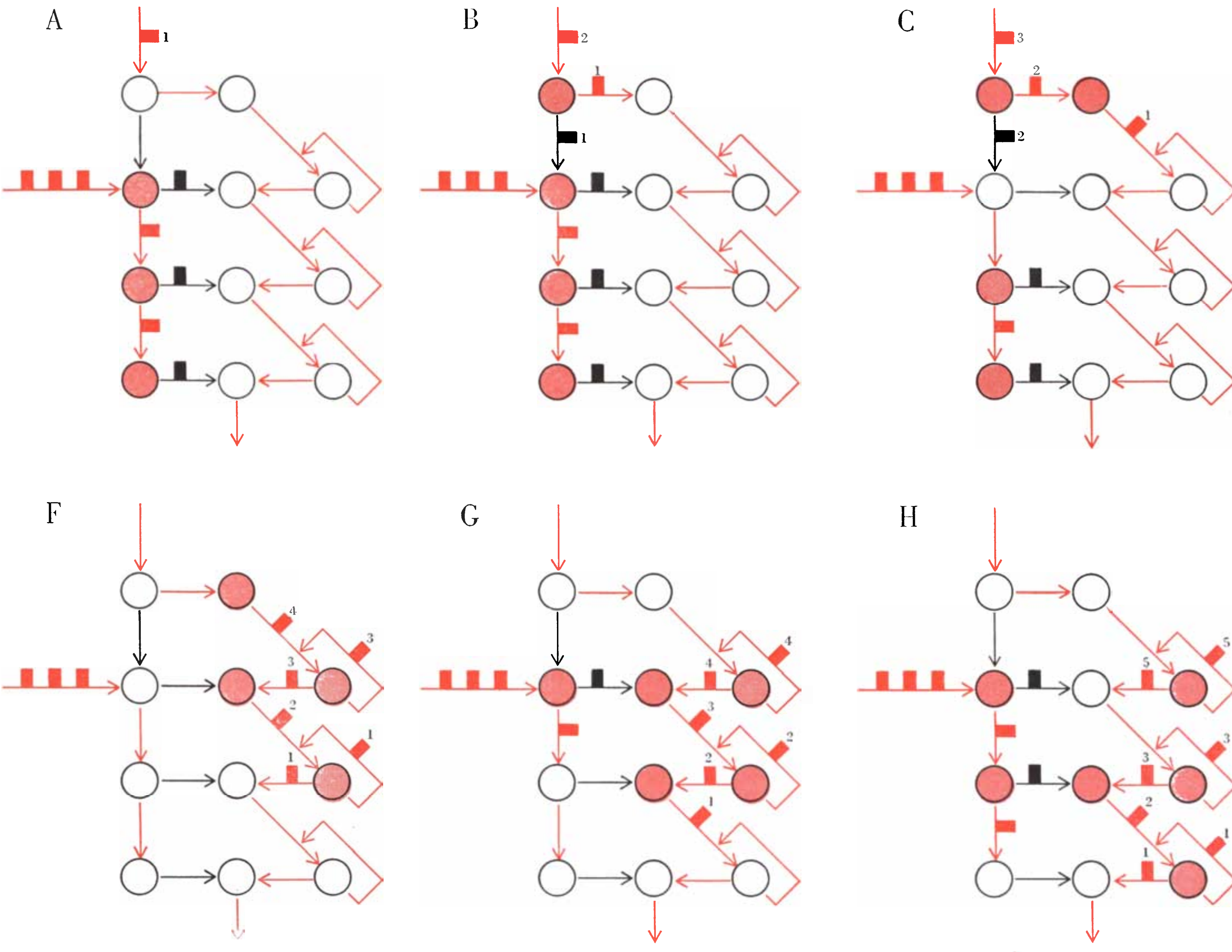


diagram the pulse received by the first cell is not fired by the third. The third cell will fire the pulse only if the activating pulse of the first cell "and" the inhibiting pulse of the second are fired on the same cycle (*second diagram*).



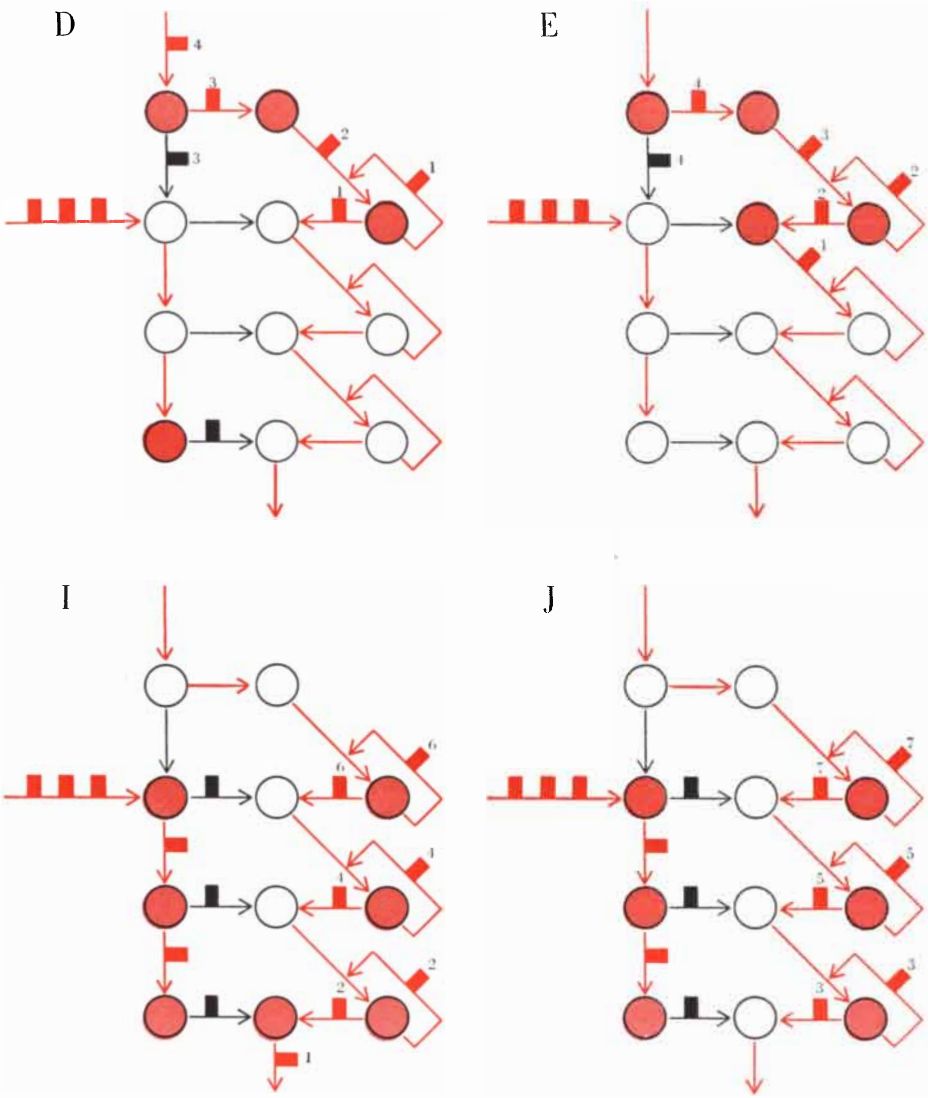
MEMORY CIRCUIT feeds the output of a cell back into its input. Thus if the cell is activated, it "remembers" by firing constantly (*top*). If it is inhibited at any time later, it stops firing (*bottom*).



COUNTING CIRCUIT "counts" to four and then fires. The conventions in this series of diagrams are the same as those in the illustrations on the four preceding pages, with two important exceptions. The first is that, where the diagrams on the preceding pages show each circuit during two or more cycles, each of these diagrams shows the circuit during a single cycle. The second exception is that, when a cell fires, it lights up (*red tone*). The input of this circuit is the activating pathway at the top. The output of the circuit is the activating pathway at the bottom. In addition one of the cells has a constant activating input (*left*). The three cells at the right are memory cells (*see diagram at the right on preceding page*). These cells can activate the three cells to the left of them.

The key question is: How do you describe a complex machine in terms that a relatively simple machine can understand? The answer is that you devise a simple code which can describe any machine (or at least any Turing machine), and that you design the universal machine so that it will be able to understand this code. To understand a Turing machine we need only know its table of commands, so it suffices to have a simple code for tables of commands. We will sketch one possible way of representing each conceivable table of commands by an integer. Of course there are infinitely many such tables, but there are also infinitely many integers—that is why they are so useful in mathematics.

A table of commands consists of P rows. Each row has three commands in it, corresponding to seeing a blank, an X or a 1. The first step is to get rid of the letters in the table [*refer again to page 65*]. This can be done by replacing E, X, D, L, R and S by 1 through 6 respectively. Thus the commands on the first line of the table of our sample machine become 3–6, 1–2, 5–1. Step two: Get rid of the question mark and the exclamation point, say by putting 1 and 2 for them respectively. (Since these occur only in conjunction with an S, there is no danger of confusing them with memory positions 1 and 2.) Thus the second row of our table becomes 5–2, 1–3, 6–1. Step three: Represent each row by a single integer. There is a famous simple way of doing this; namely by treating the numbers as exponents to primes and obtaining a product which completely specifies the series of numbers. As the final step, we represent the entire table with a single number obtained by the same trick. Our code number for this table will be $2^{2991509440920}$ times 3 raised to the number of the second row. It is an enormous number, but it does identify our table of commands uniquely. And it is a straightforward mechanical task to design the universal machine so that it can



These three cells can in turn be inhibited by the three cells to the left of them. Now in the first four cycles (A, B, C and D) four pulses are put into the circuit. The position of each pulse in succeeding cycles is indicated by small numbers. In the ninth cycle (I) the circuit, having "counted" the four pulses, fires once. In the 10th cycle (J) the circuit returns to its original state with the exception that pulses are still circulating through the memory cells. A practical counting circuit would be fitted with a device to wipe out these memories.

decode the large number and reproduce the table of commands. With the table of commands written down, the machine then knows what the machine it is copying would do in any given situation.

The universal machine is remarkably human. It starts with very limited abilities, and it learns more and more by imitation and by absorbing information from the outside. We feel that the potentialities of the human brain are inexhaustible. But would this be the case if we were unable to communicate with the world around us? A man robbed of his five senses is comparable to a Turing machine with a fixed tape, but a normal human being is like the universal machine. Given enough time, he can learn to do anything.

But some readers will feel we have given in too soon to Turing's persuasive argument. After all a human being must step in and give the universal machine the code number. If we allow that, why not give the machine the answer in the first place? Turing's reply would have been that the universal machine does not need a man to encode the table; it can be designed to do its own coding, just as it can be designed to decode.

So we grant this amazing machine its universal status. And although its table of logical control has only a few thousand entries, it seems to be able to do essentially all the problem-solving tasks that we can. Of course it might take a billion years to do something we can do in an hour. The "outside world" from which it can learn is much more restricted than ours, being limited to Turing machines. But may not all this be just a difference of degree? Are we, as rational beings, basically different from universal Turing machines?

The usual answer is that whatever else machines can do, it still takes a man to build the machine. Who would dare to say that a machine can reproduce itself and make other machines?

Von Neumann would. As a matter of fact, he has blue-printed just such a machine.

The Reproducing Machine

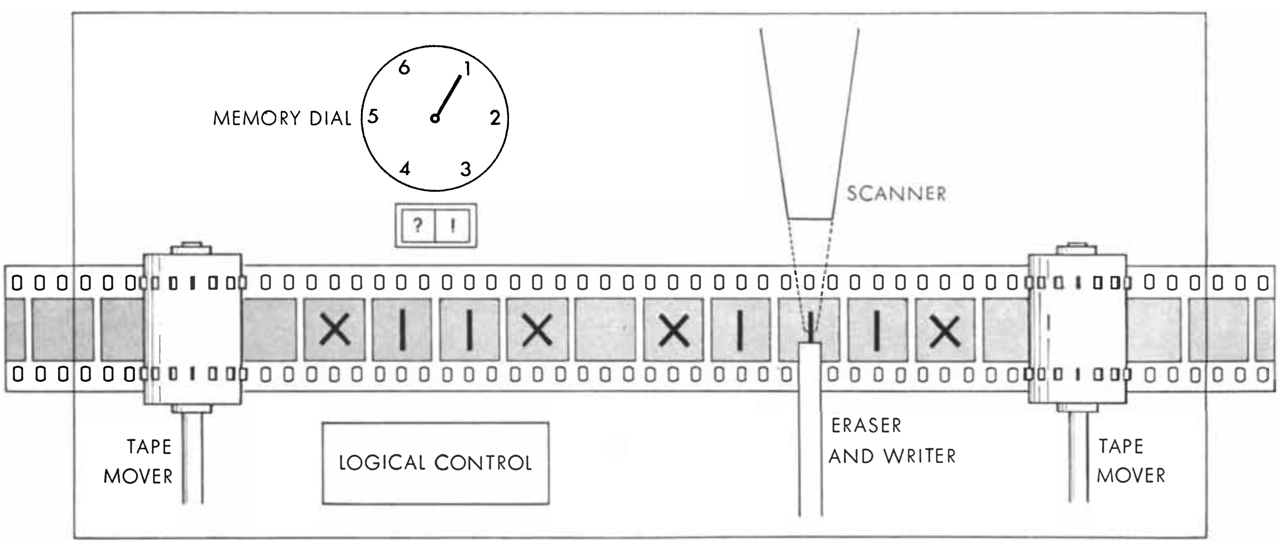
What do we mean by reproduction? If we mean the creation of an object like the original out of nothing, then no machine can reproduce, but neither can a human being. If reproduction is not to violate the conservation of energy principle, building materials must be available. The characteristic feature of the reproduction of life is that the living organism can create a new organism like itself out of inert matter surrounding it.

If we agree that machines are not alive, and if we insist that the creation of life is an essential feature of reproduction, then we have begged the question: A machine cannot reproduce. So we must reformulate the problem in a way that won't make machine reproduction logically impossible. We must omit the word "living." We shall ask that the machine create a new organism like itself out of simple parts contained in the environment.

Human beings find the raw material in the form of food; that is, quite highly organized chemicals. Thus we cannot even say that we produce order out of complete disorder, but rather we transform more simply organized matter into complex matter. We must accordingly assume that the machine is surrounded with pieces of matter, simpler than any part of the machine. The hypothetical parts list would be rolls of tape, pencils, erasers, vacuum tubes, dials, photoelectric cells, motors, shafts, wire, batteries and so on. We must endow the machine with the ability to transform pieces of matter into these parts and to organize them into a new machine.

Von Neumann simplified the problem by making a number of reasonable assumptions. First of all he realized that it is inessential for the machine to be able to move around. Rather, he has the mechanism sending out impulses which organize the surroundings by remote control. Secondly, he asssumed that space is divided into cubical cells, and that each part of the machine and each piece of raw material occupies just one cell. Thirdly, he assumed that the processes are quantized not only in space but in time; that is, we have cycles during which all action takes place. It is not even necessary to have three dimensions: a two-dimensional lattice will serve as well as the network of cubes.

Our space will be a very large (in principle infinite) sheet, divided into squares. A machine occupies a connected area consisting of a large number of squares. Since each square represents a part of the machine, the number of squares occupied is a measure of the complexity of the machine. The machine is surrounded by inert cells, which it has to organize. To make this possible the machine must be a combination of a brain and a brawn machine, since it not only organizes but also transforms matter. Accordingly the von Neumann machine has three kinds of parts. It has neurons similar to those discussed in the model of the nervous system. These provide the logical control. Then it has transmission cells, which carry messages



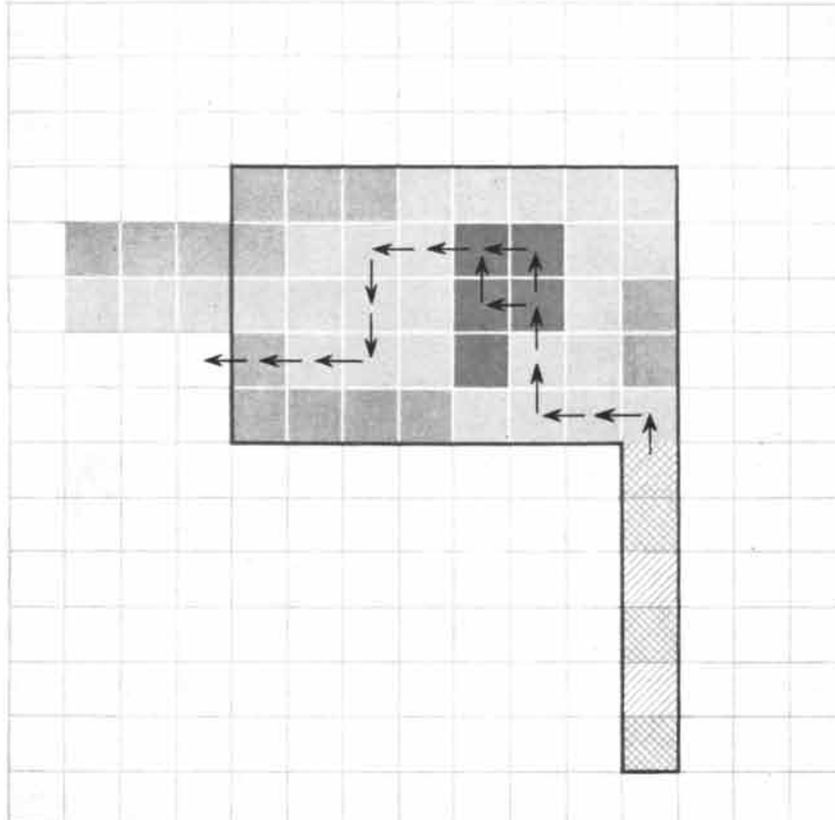
TURING MACHINE designed for simple addition is confronted with the numbers 2 and 3. The numbers are indicated on the tape; each digit is represented by a 1. X is a signal that a number is about to begin or has just ended. The logical control of the machine is depicted in the table on the opposite page. The horizontal rows of the table represent the position of the memory dial (*1, 2, 3, 4, 5 or 6*). The vertical columns represent the symbol on the tape (*blank, X or 1*). The symbols at the intersection of the rows and columns are commands to the machine. E means erase the symbol on the tape; X, write an X on the tape; D, write the digit 1 on the tape; R, move the tape one frame to the right; L, move the tape one frame to the left; S, stop; ?, something is wrong; !, the operation is completed; 1, 2, 3, 4, 5 or 6, turn the memory dial to that position. Thus at the upper left in the table the memory dial is in position 1 and the tape is blank; the command is D6, or write the digit 1 on the tape and turn the memory dial to position 6. Then in the beginning position shown above the machine begins to operate as follows. In the first step the memory dial is in position 1 and the tape shows a 1. The command is R1: move the tape one frame to the right and leave the memory dial in position 1. In the second

from the control centers. They have an opening through which they can receive impulses, and an output through which the impulse is passed on a cycle later. A string of transmission cells, properly adjoined, forms a channel through which messages can be sent. In addition the machine has muscles. These cells can change the surrounding cells, building them up from less highly organized to more complex cells or breaking them down. They bring about changes analogous to those produced by a combination of muscular and chemical action in the human body. Their primary use is, of course, the changing of an inert cell into a machine part.

As in the nervous system, the operation proceeds by steps: the state of every cell is determined by its state and the state of its neighbors a cycle earlier. The neurons and transmission cells are either quiescent or they can send out an impulse if properly stimulated. The muscle cells receive commands from the neurons through the transmission cells, and react either by "killing" some undesired part (*i.e.*, making it inert) or by transforming some inert cell in the environment to a machine part of a specified kind. So far the machine is similar in structure to a higher animal. Its neurons form the central nervous system;

	☐	X	I
1	D6	E2	R1
2	R2	E3	?2
3	R3	E4	E5
4	L4	?4	R6
5	L5	?5	R1
6	X6	! 6	R3

step the situation and the response are the same. In the third step the memory dial is in position 1 and the tape shows an X. The command is E2: erase the X and turn the memory dial to position 2. In this way the machine comes up with the answer 5 on its memory dial in 36 steps. On the 37th step the machine stops and signals with a ! that it is finished. If the reader is skeptical and hardy, he is invited to trace the whole process!



VON NEUMANN MACHINE is theoretically capable of reproducing itself. This is a highly simplified diagram of its conceptual units. The darkest squares are the "nerve cells" of the "brain." The next lightest squares are "muscle cells." The next lightest are transmission cells. The crosshatched squares are the "tail" which bears the instructions of the machine. The double hatching represents an "on" signal; the single hatching, an "off" signal. The empty squares are units of the environment which the machine manipulates. The arrows indicate that instructions are coming from the tail, on the basis of which the brain instructs a muscle cell to act on its surroundings. The machine has sent out a "feeler" to the left.

the transmission cells establish contact with various organs; the organs perform their designated tasks upon receiving a command.

The instructions may be very long. Hence they must in a sense be external. Von Neumann's machine has a tail containing the blueprint of what it is to build. This tail is a very long strip containing coded instructions. The basic box performs two types of functions: it follows instructions from its tail, and it is able to copy the tail. Suppose the tail contains a coded description of the basic box. Then the box will, following instructions, build another box like itself. When it is finished, it proceeds to copy its own tail, attaching it to the new box. And so it reproduces itself.

The secret of the machine is that it does not try to copy itself. Von Neumann designed a machine that can build any machine from a description of it, and hence can build one like itself. Then it is an easy matter to copy the large but simple tail containing the instructions and attach it to the offspring. Thereafter the new machine can go on producing more and more machines until all the raw material is used up or until the machines get into conflict with each other—imitating even in this their human designers.

It is amazing to see how few parts such a machine needs to have. Von Neumann's blueprints call for a basic box of 80 by 400 squares, plus a tail 150,000 squares long. The basic box has the three kinds of parts described—neurons, transmission cells and muscle cells. The three types of cells differ only as to their state of excitation and the way in which they are connected. The tail is even simpler: it has cells, which are either "on" or "off," holding a code. So we have about 200,000 cells, most of which are of the simplest possible kind, and of which only a negligible fraction is even as complex as the logical control neuron. No matter how we measure complexity, this is vastly simpler than a human being, and yet the machine is self-reproducing.

The Genetic Tail

Pressing the analogy between the machine and the human organism, we might compare the tail to the set of chromosomes. Our machine always copies its tail for the new machine, just as each daughter cell in the body copies the chromosomes of its parent. It is most significant that while the chromosomes take up a minute part of the body, the tail is larger than the entire basic box in the machine. This indicates that the coding of traits by chromosomes is amazingly efficient and compact. But in all fairness we must point out that the chromosomes serve a lesser role than the tail. The tail contains a complete description of the basic box, while the chromosome description is incomplete: the offspring only resembles the parent; it is not an exact duplicate. It would be most interesting to try to continue von Neumann's pioneer work by designing a machine that could take an incomplete description and build a reasonable likeness of itself.

Could such machines go through an evolutionary process? One might design the tails in such a way that in every cycle a small number of random changes occurred (*e.g.*, changing an "on" to an "off" in the code or *vice versa*). These would be like mutations; if the machine could still produce offspring, it would pass the changes on. One could further arrange to limit the supply of raw material, so that the machines would have to compete for *Lebensraum*, even to the extent of killing one another.

Of course none of the machines described in this article has actually been built, so far as I know, but they are all buildable. We have considered systematically what man can do, and how much of this a machine can duplicate. We have found that the brain's superiority rests on the greater complexity of the human nervous system and on the greater efficiency of the human memory. But is this an essential difference, or is it only a matter of degree that can be overcome with the progress of technology? This article attempted to show that there is no conclusive evidence for an essential gap between man and a machine. For every human activity we can conceive of a mechanical counterpart.

Naturally we still have not answered the question whether man is more than a machine. The reader will have to answer that question for himself.