TWO

CAN COMPUTERS THINK?

In the previous chapter, I provided at least the outlines of a solution to the so-called 'mind-body problem'. Though we do not know in detail how the brain functions, we do know enough to have an idea of the general relationships between brain processes and mental processes. Mental processes are caused by the behaviour of elements of the brain. At the same time, they are realised in the structure that is made up of those elements. I think this answer is consistent with the standard biological approaches to biological phenomena. Indeed, it is a kind of commonsense answer to the question, given what we know about how the world works. However, it is very much a minority point of view. The prevailing view in philosophy, psychology, and artificial intelligence is one which emphasises the analogies between the functioning of the human brain and the functioning of digital computers. According to the most extreme version of this view, the brain is just a digital computer and the mind is just a computer program. One could summarise this view – I call it 'strong artificial intelligence', or 'strong AI' – by saying that the mind is to the brain, as the program is to the computer hardware.

This view has the consequence that there is nothing essentially biological about the human mind. The brain just happens to be one of an indefinitely large number of different kinds of hardware computers that could sustain the programs which make up human intelligence. On this view, any physical system whatever that had the right program with the right inputs and outputs would have a mind in exactly the same sense that you and I have minds. So, for example, if you made a computer out of' old beer cans powered by windmills; if it had the right program, it would have to have a mind. And the point is not that for all we know it might have thoughts and feelings, but rather that it must have thoughts and feelings, because that is all there is to having thoughts and feelings : implementing the right program.

Most people who hold this view think we have not yet designed programs which are minds. But there is pretty much general agreement among them that it's only a matter of time until computer scientists and workers in artificial intelligence design the appropriate hardware and programs which will be the equivalent of human brains and minds. These will be artificial brains and minds which are in every way the equivalent of human brains and minds.

Many people outside of the field of artificial intelligence are quite amazed to discover that anybody could believe such a view as this. So, before criticising it, let me give you a few examples of the things that people in this field have actually said. Herbert Simon of Carnegie-Mellon University says that we already have machines that can literally think. There is no question of waiting for some future machine, because existing digital computers already have thoughts in exactly the same sense that you and I do. Well, fancy that ! Philosophers have been worried for centuries about whether or not a machine could think, and now we discover that they already have such machines at Carnegie-Mellon. Simon's colleague Alan Newell claims that we have now discovered (and notice that Newell says 'discovered' and not 'hypothesised' or `considered the possibility', but we have *discovered)* that intelligence is just a matter of physical symbol manipulation ; it has no essential connection with any specific kind of biological or physical wetware or hardware. Rather, any system whatever that is capable of manipulating physical symbols in the right way is capable of intelligence in the same literal sense as human intelligence of human beings. Both Simon and Newell, to their credit, emphasise that there is nothing metaphorical about these claims ; they mean them quite literally. Freeman

Dyson is quoted as having said that computers have an advantage over the rest of us when it comes to evolution. Since consciousness is just a matter of formal processes, in computers these formal processes can go on in substances that are much better able to survive in a universe that is cooling off than beings like ourselves made of our wet and messy materials. Marvin Minsky of MIT says that the next generation of computers will be so intelligent that we will 'be lucky if they are willing to keep us around the house as household pets'. My all-time favourite in the literature of exaggerated claims on behalf of the digital computer is from John McCarthy, the inventor of the term 'artificial intelligence'. McCarthy says even 'machines as simple as thermostats can be said to have beliefs'. And indeed, according to him, almost any machine capable of problem-solving can be said to have beliefs. I admire McCarthy's courage. I once asked him: 'What beliefs does your thermostat have?' And he said: 'My thermostat has three beliefs – it's too hot in here, it's too cold in here, and it's just right in here.' As a philosopher, I like all these claims for a simple reason. Unlike most philosophical theses, they are reasonably clear, and they admit of a simple and decisive refutation. It is this refutation that I am going to undertake in this chapter.

The nature of the refutation has nothing whatever to do with any particular stage of computer technology. It is important to emphasise this point because the temptation is always to think that the solution to our problems must wait on some as yet uncreated technological wonder. But in fact, the nature of the refutation is completely independent of any state of technology. It has to do with the very definition of a digital computer, with what a digital computer is.

It is essential to our conception of a digital computer that its operations can be specified purely formally; that is, we specify the steps in the operation of the computer in terms of abstract symbols – sequences of zeroes and ones printed on a tape, for example. A typical computer 'rule' will determine that when a machine is in a certain state and it has a certain symbol on its tape, then it will perform a certain operation such as erasing the symbol or printing another symbol and then enter another state such as moving the tape one square to the left. But the symbols have no meaning; they have no semantic content; they are not about anything. They have to be specified purely in terms of their formal or syntactical structure. The zeroes and ones, for example, are just numerals; they don't even stand for numbers. Indeed, it is this feature of digital computers that makes them so powerful. One and the same type of hardware, if it is appropriately designed, can be used to run an indefinite range of different programs. And one and the same program can be run on an indefinite range of different types of' hardwares.

But this feature of programs, that they are defined purely formally or syntactically, is fatal to the view that mental processes and program processes are identical. And the reason can be stated quite simply. There is more to having a mind than having formal or syntactical processes. Our internal mental states, by definition, have certain sorts of contents. If I am thinking about Kansas City or wishing that I had a cold beer to drink or wondering if there will be a fall in interest rates, in each case my mental state has a certain mental content in addition to whatever formal features it might have. That is, even if my thoughts occur to me in strings of symbols, there must be more to the thought than the abstract strings, because strings by themselves can't have any meaning. If my thoughts are to be *about* anything, then the strings must have a *meaning* which makes the thoughts about those things. In a word, the mind has more than a syntax, it has a semantics. The reason that no computer program can ever be a mind is simply that a computer program is only syntactical, and minds are more than syntactical. Minds are semantical, in the sense that they have more than a formal structure, they have a content.

To illustrate this point I have designed a certain thoughtexperiment. Imagine that a bunch of computer programmers have written a program that will enable a computer to simulate the understanding of Chinese. So, for example, if the computer is given a question in Chinese, it will match the question against its memory, or data base, and produce appropriate answers to the questions in Chinese. Suppose for the sake of argument that the computer's answers are as good as those of a native Chinese speaker. Now then, does the computer, on the basis of this, understand Chinese, does it literally understand Chinese, in the way that Chinese speakers understand Chinese? Well, imagine that you are locked in a room, and in this room are several baskets full of Chinese symbols. Imagine that you (like me) do not understand a word of Chinese, but that you are given a rule book in English for manipulating these Chinese symbols. The rules specify the manipulations of the symbols purely formally, in terms of their syntax, not their semantics. So the rule might say : 'Take a squiggle-squiggle sign out of basket number one and put it next to a squoggle-squoggle sign from basket number two.' Now suppose that some other Chinese symbols are passed into the room, and that you are given further rules for passing back Chinese symbols out of the room. Suppose that unknown to you the symbols passed into the room are called 'questions' by the people outside the room, and the symbols you pass back out of the room are called 'answers to the questions'. Suppose, furthermore, that the programmers are so good at designing the programs and that you are so good at manipulating the symbols, that very soon your answers are indistinguishable from those of a native Chinese speaker. There you are locked in your room shuffling your Chinese symbols and passing out Chinese symbols in response to incoming Chinese symbols. On the basis of the situation as I have described it, there is no way you could learn any Chinese simply by manipulating these formal symbols.

Now the point of the story is simply this: by virtue of implementing a formal computer program from the point of view of an outside observer, you behave exactly as if you understood Chinese, but all the same you don't understand a word of Chinese. But if going through the appropriate computer program for understanding Chinese is not enough to give you an understanding of Chinese, then it is not enough to give *any other digital computer* an understanding of Chinese. And again, the reason for this can be stated quite simply. If you don't understand Chinese, then no other computer could understand Chinese because no digital computer, just by virtue of running a program, has anything that you don't have. All that the computer has, as you have, is a formal program for manipulating uninterpreted Chinese symbols. To repeat, a computer has a syntax, but no semantics. The whole point of the parable of the Chinese room is to remind us of a fact that we knew all along. Understanding a language, or indeed, having mental states at all, involves more than just having a bunch of formal symbols. It involves having an interpretation, or a meaning attached to those symbols. And a digital computer, as defined, cannot have more than just formal symbols because the operation of the computer, as I said earlier, is defined in terms of its ability to implement programs. And these programs are purely formally specifiable – that is, they have no semantic content.

We can see the force of this argument if we contrast what it is like to be asked and to answer questions in English, and to be asked and to answer questions in some language where we have no knowledge of any of the meanings of the words. I magine that in the Chinese room you are also given questions in English about such things as your age or your life history, and that you answer these questions. What is the difference between the Chinese case and the English case? Well again, if like me you understand no Chinese and you do understand English, then the difference is obvious. You understand the questions in English because they are expressed in symbols whose meanings are known to you. Similarly, when you give the answers in English you are producing symbols which are meaningful to you. But in the case of the Chinese, you have none of that. In the case of the Chinese, you simply manipulate formal symbols according to a computer program, and you attach no meaning to any of the elements.

Various replies have been suggested to this argument by workers in artificial intelligence and in psychology, as well as philosophy. They all have something in common; they are all inadequate. And there is an obvious reason why they have to be inadequate, since the argument rests on a very simple logical truth, namely, syntax alone is not sufficient for semantics, and digital computers insofar as they are computers have, by definition, a syntax. alone.

I want to make this clear by considering a couple of the arguments that are often presented against me.

Some people attempt to answer the Chinese room example by saying that the whole system understands Chinese. The idea here is that though I, the person in the room manipulating the symbols do not understand Chinese, I am just the central processing unit of the computer system. They argue that it is the whole system, including the room, the baskets full of symbols and the ledgers containing the programs and perhaps other items as well, taken as a totality, that understands Chinese. But this is subject to exactly the same objection I made before. There is no way that the system can get from the syntax to the semantics. I, as the central processing unit have no way of figuring out what any of these symbols means; but then neither does the whole system.

Another common response is to imagine that we put the Chinese understanding program inside a robot. If the robot moved around and interacted causally with the world, wouldn't that be enough to guarantee that it understood Chinese? Once again the inexorability of the semantics-syntax distinction overcomes this manoeuvre. As long as we suppose that the robot has only a computer for a brain then, even though it might behave exactly as if it understood Chinese, it would still have no way of getting from the syntax to the semantics of Chinese. You can see this if you imagine that I am the computer. Inside a room in the robot's skull I shuffle symbols without knowing that some of them come in to me from television cameras attached to the robot's head and others go out to move the robot's arms and legs. As long as all I have is a formal computer program, I have no way of attaching any meaning to any of the symbols. And the fact that the robot is engaged in causal interactions with the outside world won't help me to attach any meaning to the symbols unless I have some way of finding out about that fact. Suppose the robot picks up a hamburger and this triggers the symbol for hamburger to come into the room. As long as all I have is the symbol with no knowledge of its causes or how it got there, I have no way of knowing what it means. The causal interactions between the robot and the rest of the world are irrelevant unless those causal interactions are represented in some mind or other. But there is no way they can be if all that the so-called mind consists of is a set of purely formal, syntactical operations.

It is i mportant to see exactly what is claimed and what is not claimed by my argument. Suppose we ask the question that I mentioned at the beginning: 'Could a machine think?' Well, in one sense, of course, we are all machines. We can construe the stuff inside our heads as a meat machine. And of course, we can all think. So, in one sense of 'machine', namely that sense in which a machine is just a physical system which is capable of performing certain kinds of operations, in that sense, we are all machines, and we can think. So, trivially, there are machines that can think. But that wasn't the question that bothered us. So let's try a different formulation of it. Could an artefact think? Could a man-made machine think? Well, once again, it depends on the kind of artefact. Suppose we designed a machine that was molecule-for-molecule indistinguishable from a human being. Well then, if you can duplicate the causes, you can presumably duplicate the effects. So once again, the answer to that question is, in principle at least, trivially yes. If you could build a machine that had the same structure as a human being, then presumably that machine would be able to think. Indeed, it would be a surrogate human being. Well, let's try again.

The question isn't: 'Can a machine think?' or: 'Can an artefact think?' The question is: 'Can a digital computer think?' But once again we have to be very careful in how we interpret the question. From a mathematical point of view, anything whatever can be described *as if* it were a digital computer. And that's because it can be described as instantiating or implementing a computer program. In an utterly trivial sense, the pen that is on the desk in front of me can be described as a digital computer. It just happens to have a very boring computer program. The program says: 'Stay there.' Now since in this sense, anything whatever is a digital computer, because anything whatever can be described as implementing a computer program, then once again, our question gets a trivial answer. Of course our brains are digital computers, since they implement any number of computer programs. And of course our brains can think. So once again, there is a trivial answer to the question. But that wasn't really the question we were trying to ask. The question we wanted to ask is this: 'Can a digital computer, as defined, think?' That is to say: 'Is instantiating or implementing the right computer program with the right inputs and outputs, sufficient for, or constitutive of, thinking?' And to this question, unlike its predecessors, the answer is clearly 'no'. And it is 'no' for the reason that we have spelled out, namely, the computer program is defined purely syntactically. But thinking is more than just a matter of manipulating meaningless symbols, it involves meaningful semantic contents. These semantic contents are what we mean by 'meaning'.

It is important to emphasise again that we are not talking about a particular stage of computer technology. The argument has nothing to do with the forthcoming, amazing advances in computer science. It has nothing to do with the distinction between serial and parallel processes, or with the size of programs, or the speed of computer operations, or with computers that can interact causally with their environment, or even with the invention of robots. Technological progress is always grossly exaggerated, but even subtracting the exaggeration, the development of computers has been quite remarkable, and we can reasonably expect that even more remarkable progress will be made in the future. No doubt we will be much better able to simulate human behaviour on computers than we can at present, and certainly much better than we have been able to in the past. The point I am making is that if we are talking about having mental states, having a mind, all of these simulations are simply irrelevant. It doesn't matter how good the technology is, or how rapid the calculations made by the computer are. If it really is a computer, its operations have to be defined syntactically, whereas consciousness, thoughts, feelings, emotions, and all the rest of it involve more than a syntax. Those features, by definition, the computer is unable to *duplicate* however powerful may be its ability to *simulate.* The key distinction here is between duplication and simulation. And no simulation by itself ever constitutes duplication.

What I have done so far is give a basis to the sense that those citations I began this talk with are really as preposterous as they seem. There is a puzzling question in this discussion though, and that is: 'Why would anybody ever have thought that computers could think or have feelings and emotions and all the rest of it?' After all, we can do computer simulations of any process whatever that can be given a formal description. So, we can do a computer simulation of the flow of money in the British economy, or the pattern of power distribution in the Labour party. We can do computer simulation of rain storms in the home counties, or warehouse fires in East London. Now, in each of these cases, nobody supposes that the computer simulation is actually the real thing; no one supposes that a computer simulation of a storm will leave us all wet, or a computer simulation of a fire is likely to burn the house down. Why on earth would anyone in his right mind suppose a computer simulation of mental processes actually had mental processes? I don't really know the answer to that, since the idea seems to me, to put it frankly, quite crazy from the start. But I can make a couple of speculations.

First of all, where the mind is concerned, a lot of people are still tempted to some sort of behaviourism. They think if a system behaves as if it understood Chinese, then it really must understand Chinese. But we have already refuted this form of behaviourism with the Chinese room argument. Another assumption made by many people is that the mind is not a part of the biological world, it is not a part of the world of nature. The strong artificial intelligence view relies on that in its conception that the mind is purely formal; that somehow or other, it cannot be treated as a concrete product of biological processes like any other biological product. There is in these discussions, in short, a kind of residual dualism. **Al** partisans believe that the mind is more than a part of the natural biological world; they believe that the mind is purely formally specifiable. The paradox of this is that the **AI** literature is filled with fulminations against some view called `dualism', but in fact, the whole thesis of strong **AI** rests on a kind of dualism. It rests on a rejection of the idea that the mind is just a natural biological phenomenon in the world like any other.

I want to conclude this chapter by putting together the thesis of the last chapter and the thesis of this one. Both of these theses can be stated very simply. And indeed, I am going to state them with perhaps excessive crudeness. But if we put them together I think we get a quite powerful conception of the relations of minds, brains and computers. And the argument has a very simple logical structure, so you can see whether it is valid or invalid. The first premise is:

1. Brains cause minds.

Now, of course, that is really too crude. What we mean by that is that mental processes that we consider to constitute a mind are caused, entirely caused, by processes going on inside the brain. But let's be crude, let's just abbreviate that as three words – brains cause minds. And that is just a fact about how the world works. Now let's write proposition number two:

2. Syntax is not sufficient for semantics.

That proposition is a conceptual truth. It just articulates our distinction between the notion of what is purely formal and what has content. Now, to these two propositions – that brains cause minds and that syntax is not sufficient for semantics – let's add a third and a fourth:

3. Computer programs are entirely defined by their formal, or syntactical, structure.

That proposition, I take it, is true by definition; it is part of what we mean by the notion of a computer program.

4. Minds have mental contents; specifically, they have semantic contents.

And that, I take it, is just an obvious fact about how our minds work. My thoughts, and beliefs, and desires are about something, or they refer to something, or they concern states of affairs in the world; and they do that because their content directs them at these states of affairs in the world. Now, from these four premises, we can draw our first conclusion; and it follows obviously from premises 2, 3 and 4:

CONCLUSION I. *No computer program by itself is sufficient to give a system a mind. Programs, in short, are not minds, and they are not by themselves sufficient for having minds.*

Now, that is a very powerful conclusion, because it means that the project of trying to create minds solely by designing programs is doomed from the start. And it is important to re-emphasise that this has nothing to do with any particular state of technology or any particular state of the complexity of the program. This is a purely formal, or logical, result from a set of axioms which are agreed to by all (or nearly all) of the disputants concerned. That is, even most of the hardcore enthusiasts for artificial intelligence agree that in fact, as a matter of biology, brain processes cause mental states, and they agree that programs are defined purely formally. But if you put these conclusions together with certain other things that we know, then it follows immediately that the project of strong AI is incapable of fulfilment.

However, once we have got these axioms, let's see what else we can derive. Here is a second conclusion:

CONCLUSION 2. *The way that brain functions cause minds cannot be solely in virtue of running a computer program.*

And this second conclusion follows from conjoining the first premise together with our first conclusion. That is, from the fact that brains cause minds and that programs are not enough to do the job, it follows that the way that brains cause minds can't be solely by running a computer program. Now that also I think is an important result, because it has the consequence that the brain is not, or at least is not just, a digital computer. We saw earlier that anything can trivially be described as if it were a digital computer, and brains are no exception. But the importance of this conclusion is that the computational properties of the brain are simply not enough to explain its functioning to produce mental states. And indeed, that ought to seem a commonsense scientific conclusion to us anyway because all it does is remind us of the fact that brains are biological engines; their biology matters. It is not, as several people in artificial intelligence have claimed, just an irrelevant fact about the mind that it happens to be realised in human brains.

Now, from our first premise, we can also derive a third conclusion:

CONCLUSION 3. *Anything else that caused minds would have to have causal powers at least equivalent to those of the brain.*

And this third conclusion is a trivial consequence of our first premise. It is a bit like saying that if my petrol engine drives my car at seventy-five miles an hour, then any diesel engine that was capable of doing that would have to have a power output at least equivalent to that of my petrol engine. Of course, some other system might cause mental processes using entirely different chemical or biochemical features from those the brain in fact uses. It might turn out that there are beings on other planets, or in other solar systems, that have mental states and use an entirely different biochemistry from ours. Suppose that Martians arrived on earth and we concluded that they had mental states. But suppose that when their heads were opened up, it was discovered that all they had inside was green slime. Well still, the green slime, if it functioned to produce consciousness and all the rest of their mental life, would have to have causal powers equal to those of the human brain. But now, from our first conclusion, that programs are not enough, and our third conclusion, that any other system would have to have causal powers equal to the brain, conclusion four follows immediately:

CONCLUSION 4. *For any artefact that we might build which had mental states equivalent to human mental states, the implementation of a computer program would not by itself be sufficient. Rather the artefact would have to have powers equivalent to the powers of the human brain.*

The upshot of this discussion I believe is to remind us of something that we have known all along: namely, mental states are biological phenomena. Consciousness, intentionality, subjectivity and mental causation are all a part of our biological life history, along with growth, reproduction, the secretion of bile, and digestion.