There is similar evidence that the ability to analyze a face's emotional expression is also handled by a separate module. Kurucz and Feldmar (1979) describe a patient who was unable to interpret facial expression, despite being able to identify the face. In a different study (Kurucz, Feldmar, and Werner 1979), the same authors discovered a patient who could not identify a face, but was able to recognize the emotional expression on it. Again, this suggests that different modules handle facial recognition and interpretation of facial expression.

Finally, "facial speech analysis" (especially the ability to lip-read) appears to be carried out independently from other forms of facial processing. Campbell, Landis, and Regard (1986) came across two patients who, taken together, demonstrate this. The first could neither recognize faces *nor* read emotional expressions, but performed normally on a task that required the ability to lip-read. The second had intact recognition and emotional-analysis skills, but could no longer judge what was being said on the basis of lip-reading.

With these kinds of dissociations in mind, the most influential models of the cognitive operations involved in face processing have posited separate modules for each of the dissociable functions. The famous Bruce and Young (1986) model features different modules for the analysis of expression, lip-reading, recognition, and so on. This fragmentation of what were thought to be unitary mental processes is a pervasive feature of contemporary cognitive-scientific research. In this respect, cognitive science and folk psychology part company—or so it seems to us. It's our impression that folk psychologists regard a person's performance at any and all cognitive tasks as due to the operation of a unitary mind that does all its work in accordance with a single set of rules. Thus folk psychologists are inclined to believe that we go through the same types of mental processes when we recognize the face of a friend as when we recognize our hat. Moreover, the process of object/face recognition is thought to be a special case of a more general cognitive strategy for drawing conclusions from sensory evidence—a strategy that's employed in picking out grammatical sentences as well as in picking out our hat. The data on language learning and face recognition (inter alia) do not support this view.

In closing, it's worth noting that Freud anticipated the modularity theory. Id, ego, and superego are clearly intended to be modular components of the mind: they have different contents, and the contents are manipulated by different rules (recall the differences between the primary-process thinking of the id and the secondary-process thinking of the ego). By our reckoning, that makes four central cognitive-scientific themes that were foreshadowed in psychoanalysis: (1) the modularity of mind, (2) the pervasiveness of human irrationality, (3) unconscious mental processes, and (4) functionalism.

7.7. Artificial Intelligence

There are several quite different enterprises that go by the name of "artificial intelligence," or AI. Some of them are intimately connected with the goals and methods of cognitive science. All of them involve attempts to write programs that enable computers to perform cognitive tasks, such as alphabetizing lists (which turns out to be easy to program) or writing summaries of longer texts (which turns out to be very difficult). To write a program for a task is to give a series of absolutely explicit instructions—an *algorithm*—for how to do the task. As an example of a set of instructions that illustrate what it is *not* to be an algorithm, here are the directions given in a children's magazine for how to put on an opera: (1) write the opera, (2) get your friends to play all the parts, and (3) charge admission.

Naturally, any form of instruction must presuppose that the instructee is able to perform certain tasks without being told how. These are the *primitive operations* of the system. It's widely believed that any set of instructions that can be followed by a computer of any design can be analyzed into the following primitive operations: (1) recognizing the difference between two symbols (e.g., 0 and 1); (2) following the instruction either to leave the symbol as it is or to change it to the other symbol; and (3) following the instruction to move on to the next symbol to the right or to the left in a series, or to halt. A device that's capable of performing these primitive operations and nothing more is called a *Turing machine,* after the computer science pioneer Alan Turing.

Here's a simple Turing machine program: if the current symbol is 0, leave it alone and move to the symbol on the left; if the current symbol is 1, change it to 0 and move to the symbol on the right. Suppose that the machine starts by reading the underlined symbol of the following series:

. . . 111<u>0</u> . . .

Since the current symbol is 0, the program instructs the machine to leave it alone and move to the left:

. . . 11<u>1</u>0 . . .

Since the current symbol is now 1, the program instructs the machine to change it to 0 and move to the right:

. . . 110<u>0</u> . . .

The next few steps are as follows:

. . . 11<u>0</u>0 . . .

. . . 1<u>1</u>00 . . .

. . . 10<u>0</u>0 . . .

. . . 1<u>0</u>00 . . .

. . . <u>1</u>000 . . .

. . . 0<u>0</u>00 . . .

This particular Turing machine is programmed to be an eraser of 1s.

As noted above, it's widely accepted that a Turing machine can do anything that any computer can do (although it may take much longer to do it than a contemporary PC). Nobody has ever been able to construct a formal proof of this thesis. However, it's believed to be true by the overwhelming majority of discussants of AI. Granting that the thesis is true, it's easy to understand why Turing machines loom large in the theoretical literature of AI: if you want to show that computers can't be programmed to perform a certain kind of task, you need only show that Turing machines can't do that task.

There are various reasons why one might want to write programs that enable computers to perform cognitive tasks. One reason is simply to get the job done so that people don't have to do it. This is

applied AI. The goal here is a practical one. It's to help people avoid tedium (as when a program is used to alphabetize long lists of names), or to perform essential tasks that human minds are not very good at (such as working out airplane traffic patterns at busy airports). This enterprise has no bearing on the problems of psychology. There are, however, at least two types of AI that are intimately connected to the work of cognitive science. John Searle (1980) has dubbed them *weak* and *strong* AI. We'll devote a separate section to each.

7.8. Weak AI

When we're doing applied AI, all we care about is that the program that we're writing does the cognitive job at hand. If the job is to alphabetize lists of names, "Berger" should come before "Berkowitz," and so on. When we're doing weak AI, however, we want the steps of the program to recapitulate the cognitive steps that a human agent goes through when she's engaged in the task. Weak-AI programmers regard such a program as a *theory* of how human agents perform the task. The difference between the goals of applied AI and weak AI is highlighted by their treatment of cognitive errors. Suppose that we human beings have a tendency to make a characteristic mistake when we try to solve a particular cognitive problem. If we're doing applied AI, our concern is only with getting the correct solution to the problem; thus we'll try to write a program that *avoids* committing the error that we humans are prone to. But if we're doing weak AI, our concern is to lay out the problem-solving procedure that's actually used by humans. In this case, the program fails to achieve the aim of the programmer unless it makes the computer commit the same errors as we do.

A caveat: it can never be claimed of any program that every program line corresponds to a cognitive step taken by human minds. For example, most computers are constructed in such a way that when the task is to do an arithmetic problem like finding the sum of 7 + 3, the program calls for converting these decimal numbers into binary numbers, adding the two binary numbers, and then reconverting the binary sum to its decimal equivalent. Nobody wants to claim that people perform these binary-decimal conversions when they add. The point is that weak-AI researchers have to specify what aspects of the program are to be treated as theoretically significant.

So, weak AI is a style of psychological theorizing. What are the advantages, if any, of theorizing by writing programs? A common answer is that it enables us to formulate and derive consequences from theories that are too complex to be wielded by the natural human mind. We can write programs that specify how hundreds, or even thousands, of factors interact to influence performance. Even if we could formulate such a theory using only paper and pencil (which is doubtful), the task of calculating a prediction from the theory would surely be beyond our cognitive capacities. If the theory is written in the form of a program, however, deriving predictions is the easiest thing in the world: just run the program on a computer and see what it does. The output of the computer is the theory's prediction of what the human output will be. The bottom line is that weak AI extends the range of psychological theories that are in play.

This putative advantage has been challenged. As is always the case in writing programs, computer simulations of human cognition go through numerous rounds of trial-and-error revisions to get the bugs out of the system. When the program is finally in the form that the programmers want it to have, it's rarely the case that the programmers have a clear vision of how it is that this particular program yields the desired results. In fact, considering both the extreme length and the cobbled-together genesis of all but the most trivial programs, it's fair to say that no human mind *could* comprehend the deductive relation between the program and its consequences. The AI researcher knows only that the program does the job. This knowledge is enough for the practical purpose of finding out what happens next. But if you don't understand why *this* program yields *this* performance, it isn't clear that you can claim to have a theoretical understanding of why this performance took place. At the very least, getting an accurate simulation of a phenomenon doesn't satisfy the conventional goal of scientific work.

A more secure benefit of weak AI is that casting one's theory in the form of a program provides us with a convenient and foolproof method of checking whether the proposed theory really does explain the performance for which it was designed. You need only run the program and see whether the expected performance takes place. If it does, then you can be sure that your theoretical explanation is complete. (An explanation of a phenomenon is complete if it says enough for us to be able to deduce the phenomenon. Of course, the completeness of a theoretical explanation doesn't ensure its truth.) The rigors of

AI have already borne fruit in revealing hidden gaps in theories whose completeness seemed intuitively self-evident. When the theory was put in program form, the machine did not run. The most notable example is undoubtedly the *frame problem*, which, according to Dennett, is a "new, deep epistemological problem—accessible in principle but unnoticed by generations of philosophers—brought to light by the novel methods of AI, and still far from being solved" (1998, 183).

The frame problem arose in the course of trying to devise a program for updating one's stock of beliefs (misleadingly called a "knowledge base") upon receipt of new information. Suppose, for example, that we receive and accept the information that an Antarctic penguin has been found who speaks fluent English. As a result, there are many propositions that we might previously have endorsed, but that we must now repudiate. These might include the propositions that only human beings can master a natural language, that no beings native to the Antarctic speak English, that only featherless bipeds possess the ability to give a passable after-dinner speech, and so on. On the other hand, a great many of our beliefs will remain totally unaffected by the new discovery. These include the belief that penguins are native to the Antarctic, that Paris is the capital of France, and that 2 + 2 is not equal to 5. What is the procedure followed in making such a revision?

Prior to AI, it had been tacitly assumed that something like the following account is more or less adequate: the new item of information P is checked for consistency against our old beliefs Q1, Q2, . . . Qn. When a Qi is found that is inconsistent with P, it is changed to its negation not-Qi. When AI researchers actually tried to implement this idea, they ran into an immediate problem: any knowledge base comparable to a human being's is so large that the requisite exhaustive check is simply impractical. According to the account we are considering, the discovery that a penguin speaks English is followed by a process of ascertaining that the new information is *not* inconsistent with our arithmetical beliefs, or with our beliefs about the genealogy of the royal houses of Europe, or with the recipes of all the foods that we know how to prepare, and so on.

Evidently, we can't assume that the consistency check of the knowledge base proceeds randomly, or in alphabetical order, or in any other order wherein the items that need to be negated are distributed randomly. We need to develop an algorithm whereby the great mass of knowledge that is clearly irrelevant to the new item is bypassed

altogether. But how do you specify a priori what class of beliefs may potentially be affected by the news that a penguin speaks English? Consider the suggestion that we should look at the items in our knowledge base that make reference to penguins, or to English, or to any other nonlogical term that appears in the new item. On the one hand, this recommendation will cause us to overlook indefinitely many necessary changes—for the fact that a penguin speaks English has indefinitely many consequences in which neither "penguin" nor "English" appears. For example, it's incompatible with the proposition that no bird speaks a Germanic language. On the other hand, even the apparently narrow scope of a search through items relating to penguins still leaves us with too many irrelevancies to wade through—for we would have to ascertain that the new item has no effect on our beliefs that penguins are not mammals, that penguins have no credit cards, that no penguin has ever been elected to the U.S. Senate, and so on.

Moreover, suppose that we *could* tell whether a given item in the knowledge base is sufficiently relevant to the new information that it deserves to be checked for consistency. How, exactly, is this capability going to be deployed? To be sure, we now have access to the information that the existence of English-speaking penguins is irrelevant to our belief that Paris is the capital of France but it isn't at all clear how this access helps. On the face of it, it seems that we still have to consider each and every item in the knowledge base in turn. The only difference is that previously we assessed each item in turn for consistency with the new item. Now we assess each item for *relevance* to the new item. Only the items that are found to be relevant are sent along for an evaluation of their consistency with the new item. But because this stage has to be preceded by an exhaustive differentiation of the relevant items from the irrelevant items, there is no theoretical gain.

So we need even more than an algorithm for relevance. We need to come up with a procedure wherein the irrelevant items don't have to be attended to at all—or at least where the number of irrelevant items that have to be attended to is greatly thinned out. But how can you *avoid* the irrelevancies without first having to identify them *as* irrelevancies? Nobody has any idea. It had been thought that the general idea of a search through a memory store would do the explanatory job—that it was just a matter of ironing out the details. But when AI researchers started to work on the details, they found that they didn't know how to proceed. The failure of weak AI in this regard is undoubtedly its most important accomplishment, for it has alerted psychologists to the existence of a major theoretical problem that everyone had previously overlooked.

7.9. Strong AI

The weak-AI style of theorizing is currently flourishing not only in psychology, but in all the sciences. Just as AI researchers try to write programs that simulate cognitive performances, so do meteorologists try to program simulations of weather systems. To the extent that the simulation is accurate, the computer output tells us what the weather in the real world will be like. The advantages of computer simulation over traditional, "manual" prediction are the same in meteorology as in weak AI: (1) it brings complex, multifactor theories into the scope of the manageable, and (2) it reveals implicit, taken-for-granted theoretical assumptions. Weak AI is psychology's adaptation of the new computer technology to scientific ends. As such, it's part of a broad movement that has profoundly affected all the sciences. But there's nothing in the other sciences that corresponds to *strong* AI.

Strong AI isn't merely a style. It's a *hypothesis* that's either true or false. It makes no sense to ask whether *weak* AI is true or false—that would be akin to asking whether psychology is true or false—but *strong* AI is something that one either believes or disbelieves. It's the thesis that a properly programmed computer doesn't just simulate a mind—it *is* a mind. Stated bluntly, it's the thesis that computers can literally think. If strong AI is correct, computer simulations of human cognition are very different from the meteorologists' simulations of weather systems. When the latter program a computer to simulate a hurricane, no one supposes that there really is a hurricane somewhere in the computer. But (if strong AI is correct), a computer simulation of mental processes can itself be a mental process.

The case for strong AI depends essentially on functionalism. According to functionalists, mental states are defined in terms of input-output relations. It follows that a system that perfectly simulates the input-output relations of a human mind will itself be a mind. To get from functionalism to strong AI, you need only [!] show that a perfect simulation of human cognition is possible. This was already clear to Alan Turing as far back as 1950. In a seminal article published at that time,

Turing proposed what has come to be called the *Turing Test* (Turing 1950). Here's how the test works. You, the tester, can freely exchange messages with both a human being and a computer, but you don't know which is which. Your task is to ask questions that will reveal to you which communicant is the computer. Thus if you think that humor can't be programmed, you might send the communicants a joke, ask them to explain what's funny about it, and identify the one who gets it as the human being. Of course, strong AI is committed to the view that humor *can* be programmed. In fact, if strong AI is correct, any and all human cognitive capacities can be programmed. That's what it is for a computer to pass the Turing Test—for it to be impossible for the tester to tell the difference between the computer and the human.

To reiterate: if strong AI is to be true, it has to be possible to program *all* aspects of human cognition. There's a lot about human cognition that AI researchers haven't yet been able to program. The frame problem is a good example. In fact, no computer can currently come even close to passing the Turing Test. But of course, failure to date doesn't establish impossibility. The strategy of strong AI researchers is to divide and conquer: find a program than can simulate one aspect of cognition, then find another program that simulates another aspect of cognition, and so on. The reasoning is that if you keep finding ways to simulate more and more aspects of cognition, this constitutes evidence that a computer will eventually be able to pass the Turing test. To be sure, this inductive inference isn't a sure thing, but it's a reasonable guess.

Some critics of strong AI have presented general arguments to the effect that no computer will ever be able to pass the Turing Test (e.g., Dreyfus 1992). We don't think that any of these arguments is compelling. The most effective critique of strong AI begins by granting, for the sake of the argument, that the Turing Test can be passed, and proceeds to show that the computer that's passed the test still doesn't have mental states. This is John Searle's (1980) famous *Chinese room argument*. Searle asks us to imagine that a person who doesn't speak Chinese is sitting inside a room. This person is given a box of Chinese characters, together with a rulebook. The room also has a mail slot so that the person can send and receive Chinese symbols to and from the outside. The rulebook tells him what symbols to output when he receives certain symbols as input. The person in the room just looks at the shape of the symbols he receives and looks this shape up in the book, and then determines what symbol to produce as output.

To complete the thought experiment, Searle also asks us to imagine that there are native Chinese speakers standing outside the room. These people ask the person in the room questions by entering Chinese characters through the slot. The person in the room takes these symbols, looks them up in the book, and then determines what symbols to produce in response to each question. Thus a Chinese speaker may send the Chinese characters for "how much is seven plus three?" into the room, whereupon the inhabitant of the room looks up those characters in the rule book; the rule book tells him to respond with certain other Chinese characters, which the Chinese speakers outside the room understand to mean that seven plus three is equal to ten. But the person inside the room has no understanding of either the question or the answer—he's just following the book of rules.

Finally Searle asks us to suppose that the rulebook is so sophisticated that for any question the person in the room receives, he is able to produce reasonable answers in Chinese. In fact, the answers are so good that the native Chinese speakers would not be able to distinguish between the responses they get from the person in the room and those that they would get from a native Chinese speaker. The question is: does the person in the room understand Chinese? Most of us would say that he doesn't. The person is just manipulating symbols without understanding what they mean. *But he is doing everything that a computer that passes the Turing test can possibly do!* That is, native Chinese speakers could not distinguish the responses that the person inside the room gives from those that a native speaker would give. Nonetheless, the person in the room does not understand Chinese.

Let us now state the problem more generally. The point of the thought experiment is that computers are *always* in the position of the person in the Chinese room. They are always carrying out formal manipulations of symbols using rulebooks (a.k.a. programs). As the thought experiment makes clear, however, carrying out such manipulations is *insufficient* for understanding. So computers never understand anything. And, if they don't understand anything, they don't have minds. Having a mind involves grasping the *meaning* of symbols. This grasping of the meaning of symbols doesn't happen merely by manipulating them in accordance with formal rules. So the mind can't just be a computer program. Hence strong AI seems to be a false theory about the nature of mental states.

Here's another way to describe the conclusion of the Chinese

room argument. The argument for strong AI requires the premises (1) that a computer can pass the Turing Test, and (2) that functionalism is true. Premise 1 asserts that a computer can exhibit all the same input-output relations as a human being; premise 2 asserts that two systems with identical input-output relations are in identical mental states. The Chinese room argument grants premise 1 for the sake of the argument, and purports to show that the second premise—functionalism—is false. Searle's argument is very similar to the qualia arguments against functionalism that we discussed in section 6.5. One of the arguments in that section was that a zombie devoid of qualia could still be functionally identical to a normal human being. Searle's argument is that a system that lacks any understanding of Chinese can still be functionally identical to a native Chinese speaker.

It should come as no surprise that Searle's Chinese room thought experiment has generated a great controversy in the cognitive scientific literature. Examining the huge variety of responses would require a book-length treatment in itself—indeed, there are several (see, for example, Preston and Bishop 2002). The most popular line of response among Searle's critics is the so-called *systems reply.* Proponents of the systems reply note that what does the translating in Searle's argument isn't just the person in the room. The rulebook and the box of characters are just as essential as the person: without any one of these components, there would be no translation. In other words, what does the translating—the equivalent of the computer—is a *system,* of which the person in the room is only a part. Let's grant that the person in the room doesn't understand Chinese. This is not yet to say that the system as a whole doesn't understand Chinese. To suppose that it does is to commit the *fallacy of composition,* which is to attribute properties (in this case, inabilities) to a whole simply because those properties (or inabilities) are present in the parts. In sum, Searle is accused of advancing the following argument:

(6) The man in the room does not understand Chinese

therefore:

(7) The system of which the man is a part does not understand Chinese

This is obviously a non sequitur. If it weren't, you could also conclude that you don't understand English because your kidney doesn't understand English.

Searle says that he's "somewhat embarrassed" to take the systems reply seriously, because, he thinks, it's so wildly implausible. According to Searle, the systems reply amounts to agreeing that the man in the room can't understand Chinese, but saying that somehow the mere addition of the rule book, the boxes of symbols, the pen and paper, the input and output slots, and the bricks and mortar of the room can give rise genuine understanding. He makes the obvious point here that it's difficult to see how the addition of all this extra paraphernalia could give rise to understanding where previously there was none. The only response that's available to advocates of the systems reply is to bite the bullet and aver that this could happen.

Searle also has a crisper and more effective counterargument. Suppose that the man in the room *memorizes* the symbols and the rulebook. Suppose also, that by some impressive (but not impossible) feat of mental agility, he is able to carry out all of the calculations in his head, without recourse to paper and pencil. The man need not, therefore, be constrained to stay in the room, but would be free to walk around with the whole system, as it were, inside his head. In this case, the Chinese-speaking system would consist of nothing more than the man himself. But despite his continued ability to conduct passable "conversations," we would surely want to say that he *still* doesn't understand Chinese. After all, he's still responding to Chinese questions by looking up the answers and parroting what he finds. The only difference is that, the contents of the rulebook having been memorized, he can find the answers in his own memory store. The point is that the systems reply simply doesn't apply to this case. As Searle says: "If [the man in the room] doesn't understand, then there is no way the system could understand because the system is just a part of him" (Searle 1980, 419-20).

The dispute over the Chinese room is by no means settled. It wouldn't be too much of a stretch to regard Searle's argument as dividing cognitive scientists into two broad categories: those who see it as a powerful demonstration of the failure of strong AI and its functionalist underpinnings, and those who regard it as a minor and surmountable confusion. Indeed, the two authors of this volume stand on opposite sides of this divide.