June 1982
Volume VII, No. 3

Boston Critic, Inc., 10B Mt. Auburn Street, Cambridge, MA 02138

$1.00
62105

Boston Review

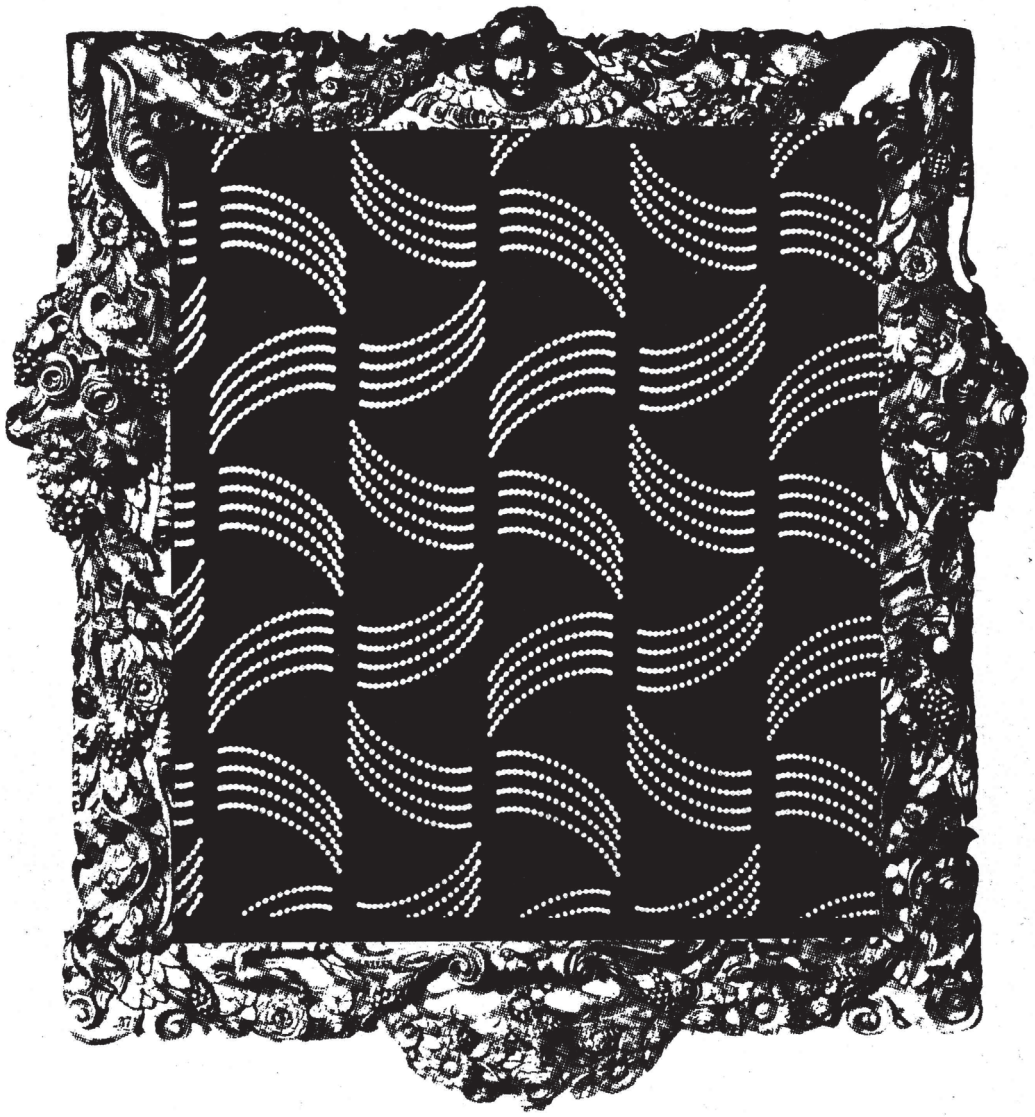
COMPUTERS

in Art, Music, Education, Psychology, Philosophy

This special issue was made possible by a grant from the Massachusetts Foundation for Humanities and Public Policy.

Computer Art or Artifice?

David Rollow



Two Space (1979), 8 min. B&W 16mm optical sound by Larry Cuba.

In an old industrial loft building on Fort Point, on the Boston waterfront, a new company has recently finished installing a computer system called Logo on an Apple microcomputer. Not long ago, I spent some time at Logo Systems learning how to draw pictures on a video display screen by typing instructions into the computer. Logo on an Apple puts into the hands of the average high school student a computer with which it is possible to draw pictures. More important, with this system it is possible to create programs that produce images. The images are not of very high resolution—the Apple's video display has less detail than a television screen—but Logo is very flexible. You can start doing interesting things with it in less than half an hour. A skeptic when I sat down, I went home impressed. All I had done was make a bright triangular spot at center screen (known as a "turtle") do cartwheels, move back and forth, and describe curves, trailing phosphorescent clouds of glory everywhere it went.

It is one thing to be impressed by a diverting afternoon's doodling with Logo; it is quite another to accept the idea that computer graphics systems (Logo and its cousins) can become a new medium of artistic expression. But, accept it or not, the idea is upon us. There are now people who call themselves computer artists and what they do is being called computer art.

Computer art. The simple conjunction of the two words makes me think of Lautreamont's famous definition of surrealism—"the gratuitous encounter of a sewing machine and a bicycle." While there is nothing gratuitous about most computer systems (at least on their own terms), the connection of computers and art has something persistently surreal about it. But why? Artists have always rushed to exploit technology. Why not the computer? Maybe it is just the newness of the computer that makes its use in art seem odd. Or maybe there is a deeper reason for uneasiness. After all, computers seem so unalterably scientific. Does the artistic use of computers threaten an encroachment of science onto art's privileged ground? Or is it the nature of the computer itself that threatens? Although computers may be tools, in principle no different than the painter's brush or the sculptor's chisel, they seem capable of becoming much more than tools. It should be reassuring that in my recent tour around the rarified world of Boston's computer artists, I saw nothing that could be called art produced by an unaided computer. There is no computer art produced without the intervention of human agency, and it would be a misconception to suppose that anything of the sort exists. Still, the specter of computer-generated art haunts the enterprise, as though the use of computers in art somehow risks the ultimate usurpation, where the computer has replaced the artist it is supposed to serve and art has lost its place as a last refuge for the uniquely human.

It is impossible to know whether these are just dreams without knowing more about the computer and its artistic capacity. Perhaps the easiest way to think of this capacity is as a means for translating numbers into pictures. A computer graphics system ordinarily consists of a computer hooked up to some type of video display tube. An image is created when the phosphor-coated screen is disturbed by a beam of electrons fired at it from an electron gun located at the back of the tube—just as in a television set. The orientation of the electron beam can be controlled by two numbers: X, specifying the horizontal position of the beam, and Y, specifying its vertical position. Thus, any point of light on the screen can be described by values for X and Y and a complete picture can be described by a list of values for X and Y, which, in effect, direct the electron gun to paint a picture on the screen by moving from one point to another. The computer can be used either to store pictures in its memory as a set of X and Y values, or to calculate a picture by executing a program that generates X and Y values. Computer graphics lan-

"Computer art. The simple conjunction of the two words makes me think of Lautreamont's famous definition of surrealism—'the gratuitous encounter of a sewing machine and a bicycle.' "

David Rollow writes fiction and works with Computer Solutions in Newton, Massachusetts.

guages like Logo relieve some of the tedium of drawing pictures in terms of X and Y by allowing the user to build powerful procedures for creating an image piece by piece instead of point by point. Once such a procedure is created it can be saved, copied, and modified at will. Some of the most powerful effects are achievable by reapplying simple procedures over and over again to transform a picture previously created and stored in computer memory, as, for example, when the picture of an object is rotated by applying a procedure for matrix multiplication to the stored X and Y values specifying a picture. Although Logo-like systems protect the user from many of the mathematical details, it is important to remember that whatever computer artists do, they must have some way of specifying the numbers that govern computer images. Often, but not always, this will involve writing a computer program.

The kind of image the computer is best suited to producing, at present, is very regular. Programs that make use of the same procedures repeatedly are the norm: the computer is good at tasks that are repetitive and hence tedious to execute. At Logo I asked a programmer friend of mine who draws whether he thought of using a computer to draw realistic pictures. "Probably not," he said. "It would be too cumbersome. I don't see the point of doing it." (That should stand as a warning to any artists who are reading this piece: If you are looking for a computer pen or paintbrush, you may not like what you find, at least not if you like to draw.)

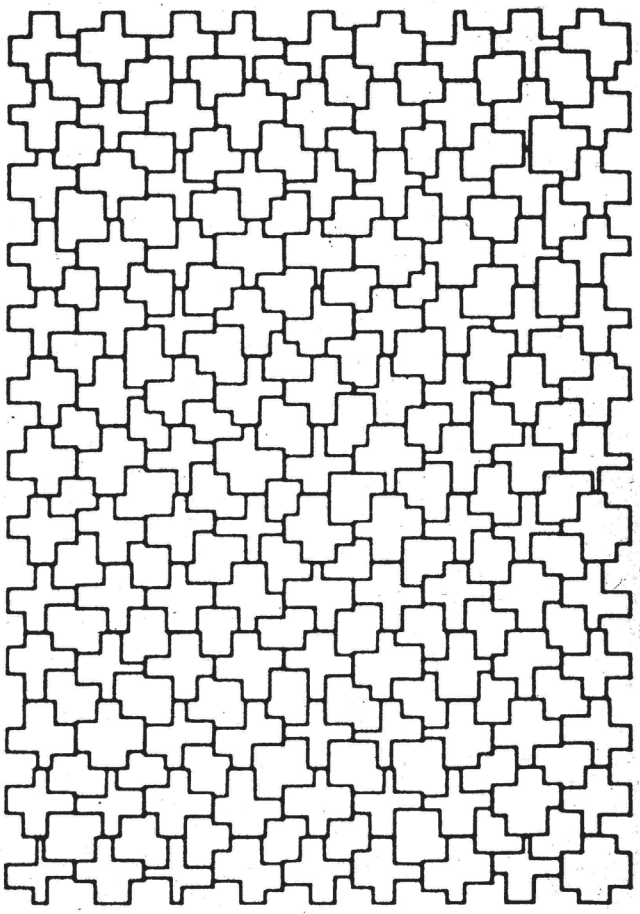
Broadly speaking there are two kinds of computer-produced image: static and moving. What's intriguing is that on close examination the distinction falls apart: The image on a graphics display screen must be "refreshed" between thirty and sixty times a second, to prevent flickering. So much work is involved in putting a static image on a display screen, it's tempting to say, that you might as well go ahead and make it move. To put the same thing in another light, the computer is so well adapted to doing things over and over that the tedious work of making mobile images—animation—is significantly reduced. Perhaps that is why much of what is currently available from the would-be computer artist consists of computer-animated films. A fair estimate of the state of the art in computer animation could be made from a Center Screen show that filled three recent evenings this spring at Harvard University. Several of the films shown were of almost no aesthetic value—there was even a brief clip from a flight simulation showing a jumbo jet taxiing down a runway—but there were good reasons for looking at nearly all of them.

A dichotomy that has always been present in visual art seems to be present in computer animation as well: Some of the work was of interest solely because of the difficulty of making it, while other work emphasized aesthetic concerns. On one side there is the everlasting confusion of art and technique, attractive to many computer scientists because they are interested in well-defined skills. On the other side there is the notion of art as an experience the work offers the audience. The art-as-skill side of this dichotomy is represented by a film called *Adam,* by T. J. O'Donnell and Arthur Olson. It opens with a circle within which a schematic figure is outlined, a sort of mechanical-drawing rendition of da Vinci's man-the-measure drawing. The two-dimensional figure steps out of the circle and becomes three-dimensional in a perspectively accurate three-dimensional world, where it goes for a walk. Limbs and body look like aircraft fuselages and flashlight batteries. The figure walks in an oddly bodiless way, since no gravity resists his muscles, and his world is friction-free. The image, we are told, is an animation of Edward Muybridge's sequential photographs of a man walking, but much has been lost in the translation. Well, who am I to complain about the absence of any dynamism in the figure? The accomplishment is astonishing! That's just the point: The "beauty" of this film, if it is beautiful, is all technical.

Another film impressive for its technical accomplishment on several levels was Jim Blinn's *Voyager,* simulating the visit of Voyager 1 to Jupiter and its moons. This work was done at the Jet Propulsion Lab, and it takes the transmissions from the satellite and puts them together in a single continuous film. I should be grateful, but I felt you had to have been there. The film has no overall rhythm, and shows no concern for its visual impact on the audience beyond the wow of the picture of the planet and its moons.

Two other films were an experience of an entirely different order: *Two Space* and *3/78* by the computer artist Larry Cuba. *Two Space* consists of austere black-and-white images, the screen subdivided by a cellular grid through which dots of light in strings like beads dance in various symmetrical patterns, translations, mirrorings, rotations. The tessellation dissolves into swarms of dots that interlace with retinal afterimages, some in color—Cuba says that the extreme black and white contrast is what produces the illusion of color in the eye—and figure-ground reversals. The rhythm of the pattern is strongly related to the Javanese gamelan music on the soundtrack. The second film, *3/78,* uses the same basic device of white dots on black, but with greater sophistication and control than displayed by *Two Space.* There is no sense of watching tiling patterns carried out for the sake of going through all the changes. The figures are more geometric. The transformations are more exciting. A single line metamorphoses into a melon shape, one square becomes a series that tumbles gracefully across the screen.

Both films have clear affinities with certain styles of abstract film and painting, especially the "pattern" painting of recent years. Cuba's strongest declared affinity is with the work of two filmmakers, Jordan Belfen and Oscar Fischinger. Cuba would prefer his films to be seen in the context of abstract animation, rather than computer animation. Asked what painters he admires, Cuba gives a rapid response that seems suggestive. "Al Held. Sol Lewitt. And the Constructivists." His reference to the Russian painters is significant, because his films fall squarely into the constructivist branch of modernism (as distinct from the expressionist branch). The static constructions, paintings, and



Chopped Square Tessellation (1980), 133 x 92mm by William Kolomyjec.

> *" 'When people talk about technology and art, they think of the latest, least familiar technology. But all art is based on technology. A pencil is technology.' "*

drawings of Sol Lewitt, for example, demonstrate precisely the same concerns as Cuba's films.

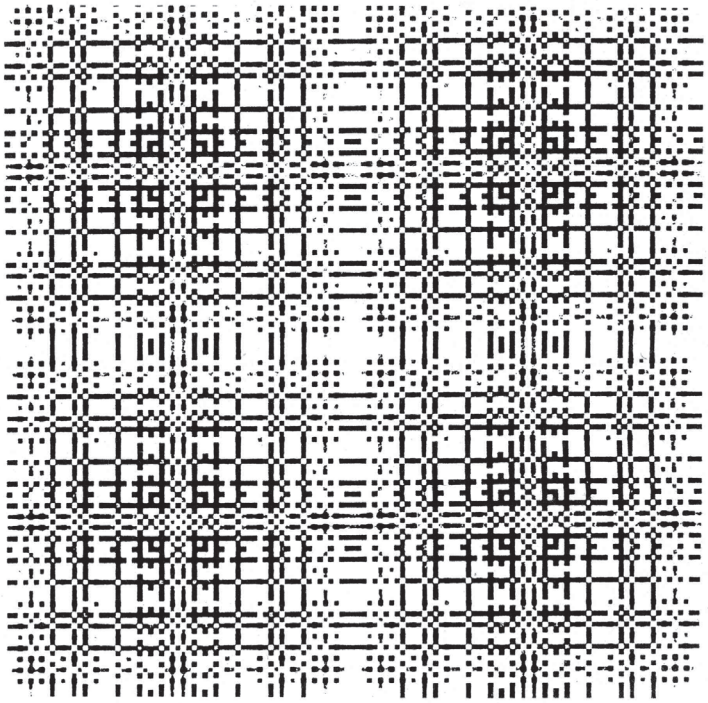
After the last screening, I asked Cuba what he thought it meant to be a computer artist. He laughed. "I'm an artist because I say I am." Cuba is a slightly built, intense man in his thirties who laughs easily. "It's the artist who says it's art. The museums and the critics who act as the arbiters tend to forget that. And most people depend on them to say what art is." But people associate computers with science, I suggested, and it seems odd, somehow frightening, that computers might have something to do with art. "That can only come from misconceptions about computers," he said. "Maybe you're not supposed to get art from a machine, like a camera, for instance. Does that mean that Ansel Adams isn't an artist? When people talk about technology and art, they think of the latest, least familiar technology. But all art is based on technology. A pencil is technology."

One difficulty for many people is that in both photography and computer animation we can't point to the unique object that is the work itself. Even after the sardonic researches of Marcel Duchamp and his kin, most of us expect the art work to be a unique, handmade object in which the artist has, through work over time, invested his spirit. But what is invested in computers is thought to be mind, not spirit, and the object is nowhere to be found. Today's artists have separated themselves so far from virtuoso effects that they work obsessively at eliminating all signs of personal handiwork, at the overthrow of the object status of the art work, at dispensing with the requirement that an art work must be executed in order to be realized. It need only be planned—or, in the case of work by magicians like Robert Irwin, the artist need only have a perceptual experience of his own and try to direct others toward it.

◆

Many people doing work in computer graphics today are not artists at all. I asked Cuba if he thought there was any difference between scientists and artists. "Computer scientists produce tools, artists produce works," he said. Without in any way denying the value of this tool-building, which will eventually make a great storehouse of graphic images available to anyone who wants to use them, Cuba is at pains to disassociate himself from this engineering orientation. He is not using the computer to mimic or preserve the traces of handiwork. Computer graphics systems do exist—so-called "paint-brush programs"—designed to reproduce gesture. His work, he says, is technically primitive by comparison. He is interested in putting certain expressive qualities into his work—above all, the serenely elegant movements of patterns inspired by mathematics. "If the basis of your work is mathematics, gesture isn't important." He points out that using language involves being bound up in a particular world view, while forms derived from mathematics are spiritually universal, the way music can be. Cuba's work resembles choreography. The experience, as always in film, is perception over time, but Cuba's films are free of narrative content. It is motion in itself that Cuba thinks is the most important thing the computer adds to art.

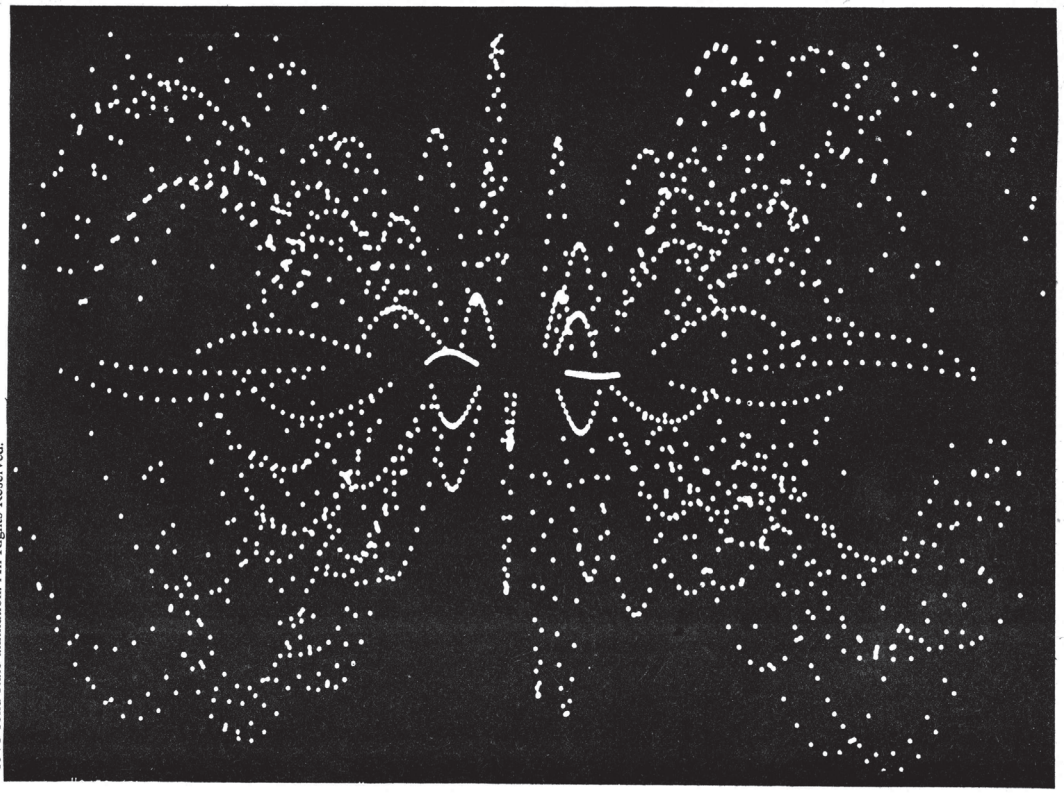
One of the interesting things about computers is that, like art, they challenge and sometimes confuse old, very familiar human boundaries. "No other human artifact responds to an outside stimulus," said my friend at Logo. "No other human artifact can interact with humans." Computers print messages on the video screen, for instance, and ask you questions, say unexpected things, call you by name, do different things according to how you respond. This *is* an interaction. It is the imitation of a dialogue. The sense you have is of dealing with a power—a genie or a splendid valet. "No general ever had such obedient troops, no prince such loyal subjects," a computer artist told me. What happens in such statements as these, however, is a kind of unwitting metaphorical transfer: What are ordinarily regarded as the attributes of person, and thus part of a total human experience, are *transferred* to computers, or rather to computer programs. The program is said to react, respond, interact, act—and before we are aware of the shift in meaning, we find our artist engaged in earnest conversation with the forms of his imagination, with a machine, as if he were talking to a stone. Computers, or computer programs, don't *do,* they imitate doing. They don't act, they imitate action. In many cases the imitation is so effective that we cease to see it as figurative.



$Z = X^4 + Y^4$ *1980* by Herbert Franke and Ernst Triendl.

Artists have always confronted us with this problem: It was the boast of the painter Zeuxis that his paintings were so convincing that birds attempted to eat the fruit on the trees he painted. So too the claim of Orpheus, the original poet, whose song silenced birds and brought stones to tears. Behind both these myths is a central myth about art, that it rivals or even surpasses nature. Orpheus shamed the birds, Zeuxis made fools of them. The

3/78 (Objects and Transformations) (1978), 6 min. B&W 16mm optical sound by Larry Cuba.



birds in these stories are the audience—ourselves—deceived by an imitation.

◆

The ancestry of the digital computer shows a close, if little appreciated, relation to art that ought to make us question still further its seeming opposition to aesthetic values. The computer's mythic father was Charles Babbage, a nineteenth-century inventor who daydreamed of automating the logarithm tables and spent much of his adult life designing a machine he called the Analytical Engine—a machine that was never built, but which laid down in its design the fundamental principles of today's digital computers. Babbage, as Hugh Kenner points out in his entertaining book, *The Counterfeitors,* was a twentieth-century man of sense in the post-romantic nineteenth century. But he was also fascinated by the mechanical ducks and flute-playing automata of the century before, the mechanical dolls that are so ominously prominent in the stories of E.T.A. Hoffmann. Babbage as a child saw in the home of a man named Merlin (believe it or not) a mechanical ballerina made of silver. "This lady attitudinized in a most fascinating manner," Babbage wrote. "Her eyes were full of imagination, and irresistible." In old age, he bought the ruined silver lady and restored her to working condition.

From the standpoint of the historian of the arts, this anecdote places Babbage clearly in the line of artificers that began with Hephaistos and Daedalus and continued through clockmakers to the architects and engineers of present-day America—a somewhat suspect, arcane, semi-magical kind of artist, if indeed an artist of any kind. It may be said that the computer descends from automata like the silver lady. Kenner reminds us that the analytical engine "was to have taken instructions from punched cards borrowed from the Jacquard loom, which Jacquard in turn had developed while restoring a loom devised (after principles taken from a clockwork flautist) by Vaucanson." Vaucanson, the builder of the original mechanical duck!

Babbage's collaborator and apologist was "the only daughter of the house and heart" of Lord Byron, the romantically named Lady Ada Augusta Lovelace, after whom a computer language has been named in recent years. Of the resemblance of Babbage's great engine to the Jacquard loom, Ada wrote, "We may aptly say that the analytical engine weaves algebraic patterns just as the Jacquard loom weaves flowers and leaves." Now, after the passage of a century, computers are getting good at reproducing the kind of visual patterns those looms were weaving when Babbage appropriated their punched cards for his algebraic loom. One form of computer-aided "art" we can all look forward to is the design of new plaids. Some very *odd* plaids, probably.

◆

The question whether a computer could, in principle, make art, looks a lot like the question whether a computer could, in principle, think.

The computer's great power comes from three principles: repetition, contraction, and generality, and the greatest of these is generality. The power of repetition is the computer's ability to do the same thing over and over indefinitely, and also to change what it does, little by little, so that when the cycle of repetition ends, what you started with emerges completely transformed. The principle of contraction permits you to give any program for carrying out a particular task a name. Then, whenever you need to perform that task you can simply start it up by invoking its name (a little like "open-sesame" but without the magic). Repetition and contraction can be combined: you can repeatedly run a given program to accomplish a big job bit by bit.

Finally, the power of generality is that the computer manipulates symbols with which it can imitate any kind of object or activity that can be formulated as a set of rules. Generality is the greatest of these three principles because our intelligence is general, and this apparent (if misleading) similarity gives people the idea that computers might some day be autonomously intelligent, the way humans are—that is, as we say, "able to think for themselves." As others writing in this issue point out, computers possess—*or imitate the possession of*—characteristics very like some of ours. They are goal-oriented, purposive, they comprise many interconnected systems, they are capable of interaction, and they manipulate symbols. But, you say, it's all just a simulation, right? Well, isn't that exactly what we do? Don't we simulate the world to ourselves constantly? Worse, don't we simulate ourselves to ourselves? This line of thought demonstrates one thing that is unnerving—and for many people chilling—about computers: They confront us with the difficulty of distinguishing the real from the pretend. (Ah, but so does art! so does art!)

About thirty years ago the English mathematician Alan Turing proposed that we translate the question of computer intelligence into a game, now usually called the Turing Test. In a simplified version, the game runs this way: Two players are linked by teletype in a long-distance conversation. It is one player's task to decide, by means of asking any question or playing any conversational trick he likes, whether the other player is a human being or a computer. Turing proposed that if computers could be designed to consistently fool human beings in this game, then there would be no justification for continuing to deny that computers can think.

The real beauty of Turing's parable—for it is a parable—lies not in its providing criteria for recognizing machine intelligence but in its demonstrating the difficulty of recognizing human intelligence. He spoke wryly of "the polite convention that humans think." We will do well to keep in mind that to ask whether a machine thinks is not to ask whether it is human; still less is it to ask whether humans are machines. As with thought, so too with art. In principle, a computer can be programmed to write programs—even complex programs such as those that produce graphic displays. If—and it is a big if—in the future, computers got so good at writing graphics programs that we began to find it difficult to distinguish graphic displays produced by computer-generated programs from those displays produced by hand-written programs, then would we have to grant that computers had become artists? As Woody Allen said, "If you act like an artist, they'll treat you like one." ■

"Using language involves being bound up in a particular world view, while forms derived from mathematics are spiritually universal, the way music can be."

"Computers confront us with the difficulty of distinguishing the real from the pretend. (Ah, but so does art! so does art!)"