Penn Libraries
UNIVERSITY of PENNSYLVANIA

University of Pennsylvania
ScholarlyCommons

Departmental Papers (ASC)   Annenberg School for Communication

3-1-1983

Computer-Ease: A Twentieth-Century Literacy Emergent

Carolyn Marvin
University of Pennsylvania, cmarvin@asc.upenn.edu

Mark Winther

Follow this and additional works at: https://repository.upenn.edu/asc_papers

Part of the Communication Commons

Computer-Ease: A Twentieth-Century Literacy Emergent

Disciplines
Communication | Social and Behavioral Sciences

Computer-Ease: A Twentieth-Century Literacy Emergent
by Carolyn Marvin and Mark Winther

> ***The history of written literacy and a survey of user experiences with computing suggest that, without public commitment of social and economic resources, innovations in computer technology may serve to exacerbate the current inequities in the distribution of literacy skills.***

After fifty centuries of experience with writing and five centuries of experience with printing, the definition of literacy continues to trouble everyone with a serious interest in the topic. This is because every measure of literacy ever devised appeals to some criterion of success in interpreting messages, and success in interpreting messages will always be a socially constructed rather than an objective category. In the twentieth century in the West, literacy has come to denote a consensual level of competence in deciphering and manipulating written material. But literacy may be thought of in broader terms as decoding or manipulating whatever message systems particular cultures regard as important. It was once taken for granted, for example, that God Himself had inscribed the creation with messages that men must learn to understand, and that time was the unfolding of a divine meaning for mankind to read and profit from. The definition of literacy and its proper objects is at any time not only a standard of skillful performance but also a social sorting procedure for labeling certain tasks and domains of knowledge as worthy of social attention and effort.

Recent debates about literacy in Western culture have been concerned with whether computers are also proper objects of literacy and, if so, what the definition of computer literacy should be (see, e.g., 7). In part, these anxieties stem from our high valuation of written literacy, as so many things about computer literacy do, including the term itself. As a matter of public policy the United States has been committed to the achievement of universal written literacy within its borders for roughly a century. No serious public argument can be made today that teaching literacy is not a primary responsibility of the educational system. Our devotion to the ideal of universal written literacy is based both on a belief that democracy will not work unless individual citizens can become informed and on our response to historical changes in the circumstances of industrial societies. Those societies are now held together less by families, kin groups, and local neighborhoods (arenas of oral transmission, all) than by impersonal bonds of expertise and information that support geographical diffusion and functional specialization. For a long time the maintenance of social cohesion has required that citizens be able with facility to send messages to and receive messages from those they do not know and who are not present. Written language has become a necessary social cement and literacy an essential skill of modern citizenship.

Also figuring in these developments is a continuing twentieth century debate about the proper role of technological awareness and competence in the liberal education of modern citizens. A recent influential entry in this debate is the broad proposal by the Alfred P. Sloan Foundation to U.S. educators to re-think a new liberal arts emphasizing applied mathematics and technological literacy, including computing (7).

For all these reasons, every effort to define computing skills in terms of written literacy deserves our attention. If, indeed, the comparison between computer literacy and written literacy is a void one, and if computer literacy is likely to become as important a device for creating, maintaining, and reproducing collective life as written literacy now is, we may have to consider whether there is also a public obligation to train citizens in this newest technique of modern life.

> ***In the United States written literacy generally refers to the ability to read and write Standard English at some acceptable level--but what is the language of computer literacy?***

Is it FORTRAN, BASIC, COBOL, or LISP? Is it mathematics? Is it the abbreviated even pidgin English of the end-user system? Does computer literacy have distinct components of reading and writing? Is it the ability to design systems for others to use, or is it having the presence of mind to type "help" and being able to understand the response? Whatever the future of computer literacy may be, its present forms

include a great range of skills in manipulating hardware and software in different, often incompatible systems among diverse and sometimes overlapping user communities that transmit the elements of computer literacy to novices in a variety of ways.

Written literacy, by contrast, is usually regarded as a *uniform* set of skills. As it happens, this is a relatively recent historical development, representing the convergence over centuries of widely various skills for manipulating written codes for the separate purposes of church, state, and commerce. Clanchy (3) has examined the painfully slow process in England between the eleventh and thirteenth centuries by which writing became a routine administrative tool of government. Among the circumstances that fashioned and complicated this process were the existence side by side of sacred, bureaucratic, and vernacular literacies involving different languages and performing different functions.

Not only were there several distinct written literacies, but many familiar devices of literacy, such as alphabetizing, indexing, procedures for dating documents, and conventions for validating their authenticity were just being invented and developed. Making records, storing records, and retrieving records are not, after all, the logically inseparable, self-evident utilities of literacy we often take them to be. Each of these painfully worked out and perfected applications was a kind of *program* for creating, organizing, and manipulating written information. Moreover, the effective diffusion of these techniques required dedicated efforts over a long period by royal governments able to deploy the necessary social resources and to insist on replacing familiar oral customs with innovative written procedures.

Just as written literacy was at first several literacies, a collection of distinct codes using different languages and serving different ends, so computer literacies comprise a variety of languages for as many purposes. Just as written literacies gradually converged, so we can expect a similar long-range movement in computer literacy. Just as some computer literacies require mastering the conceptual logic of programs for manipulating electronically coded Information, so medieval literates invented and mastered structurally similar (if less elaborate) techniques for handling written information. And finally, just as official intervention was necessary to educate a substantial part of the population in the new cognitive habits of written literacy, there are good reasons to expect that a similar scale of effort will be needed to achieve the popular diffusion of computer literacy.

The fact that official efforts by church or state have so often been necessary to radically transform the demography of literacy makes literacy a political phenomenon as well as a social one.

Not only medieval authorities but industrial states have found it prudent to shape the distribution of literacy to their own ends. Governments in nineteenth-century Britain and North America fueled mass literacy movements with subsidies to public education that provided workers with a greater range of skills for operating the increasingly specialized machine of industrial society, and more efficiently acculturated new immigrants (5).

A century later, more or less, state-sponsored assistance to literacy training survives in the system of public education, but with mixed results, since it has been reckoned that perhaps one in five adults in the United States is functionally illiterate (1). Under the circumstances, it is difficult to imagine the diffusion of mass computer literacy, for which written literacy is virtually a prerequisite, in the absence of sustained official interest and assistance. Such assistance may be forthcoming, since industrial governments are now irreversibly committed to computerizing records and services, just as twelfth-century royal English governments required lesser officials and laymen to cope with new forms of record-keeping as a matter of practical survival.

Contemporary computer literacy is still socially organized very much as a craft literacy like that described by Eric Havelock for early alphabetic Greece, where expertise in the new code was managed by a restricted group of the population (6, pp. 185-190). For all the immediacy of computers in our lives, in other words, direct hands-on experience with them at more than a trivial level is still rare for most people. Computerized toys and automatic banking may be the opening wedge in mass computer literacy, but their power is to the real power of computing as the ability to sign "X" in lieu of one's own signature is to the

real power of the written word. Computer games and automated banking are far more likely to play the same legitimating and familiarizing role that calendars, chapbooks, and almanacs played in the early days of printing. The continuing development of computer technology will also present short-term obstacles to the smooth growth of computer literacy by rendering some existing literacies obsolete, remaking others, and giving rise to others whose existence cannot yet be imagined.

On the other hand, there are sturdy and familiar continuities between the past and the present. New technologies characteristically learn forms from their predecessors and organize around imaginative habits associated with them. In the same way that new forms of writing first imitated oral traditions and asserted their own novel features only gradually, computer literacies bear the visible stamp of their origins in written discourse and their continuing dependency on it. Basic end-using terms like "print," " type," "read," "text," and "edit" come directly from print literacy (a species of written literacy), though in the new medium the old terms no longer mean exactly what they signify in reference to printed texts.

While it is conceivable that computers of the future will become bearers of vast quantities of information that is not alphabetically based, the potential impact of mass interactive computer graphics may not be conceivable, just as the impact of television was unimaginable to pre-electronic generations, and just as mass literacy was unimaginable to the eleventh century. At the present moment, however, the user interface is primarily alphanumeric, and print is the foundation upon which computer literacy must be built. The development of computer literacy is much less likely to de-emphasize written literacy than to make it more important than ever. The extent to which written literacy is already economically and socially stratified will strongly influence the distribution of computer literacy as well.

On the other hand, now that computer literacy is no longer the exclusive domain of cryptographers, ballistics experts, and other modern priesthoods, the diffusion of simple computer survival skills as a problem- solving response to imposed necessities resembles the diffusion of early written literacy. Our popular notions of computer literacy are certainly emphatically instrumental. While we speak of *readers* in discussing written literacy, in computing we speak of *users*. But while airline reservations clerks, secretaries, and others are learning end-user fundamentals on the job, the acquisition of powerful programming skills remains largely an elite cultivation.

In the context of an apparent decline of popular interest in books and reading, there are those who see computerization as a devaluation of imaginative thinking in favor of the narrowly instrumental, explicit, and rationalized. This is an inappropriate attempt to equate beginning computer literacy with advanced written literacy. Contemporary levels of computer literacy are more properly compared to earlier stages of written literacy. When written literacy first passed out of the Church's control, for example, its initial expansion was fueled less by imaginative impulses than by prosaically practical ones. Generally distributed skills in dealing with royal writs demanding money or information long preceded a generally distributed appreciation of the masterpieces of written literature, which are no more the characteristic form of the written word than the invoice, the laundry list, or the calendar. Computing, likewise, is more a practical than an art form at the present time, but as amenable to aesthetic treatment and development as any written format.

While the early history of written literacy offers a useful comparative framework for the analysis of computer literacy, it is only by knowing what contemporary practitioners of computer literacy consider their craft to be that we can begin to discuss the most likely development of computer literacy. If computer literacy is a shifting set of skills socially defined in response *to* changing circumstances, its future direction will emerge from the specific practices and perceptions of those who now work most closely with computers as designers, programmers, and operators. Wittingly and unwittingly, those in the midst of the current transformation are its engineers.

In order to investigate what having facility in computing means to experienced users, we asked a group of these users to reflect on their computing experience.

An electronic mail questionnaire was addressed to systems on the Advanced Research Projects Agency Network (ARPANET), one of the largest and best-known computer networks. In five open-ended

questions we asked volunteer respondents *to* discuss computer literacy in relation to their more general notions of what a computer is and what it is for, how computing compares to familiar written forms of discourse, and the social context in which computing takes place. These were our questions:

1. How do you conceptualize the major types of skills and levels of knowledge that organize computer users into major classes?
2. If you were to use a metaphor (you may use more than one) to describe the kind of thing a computer is, what would it be?
3. If you were to use a metaphor (one or more) to describe the relationship of power between you and the computer, what would it be?
4. Do you think distinctly different habits of imagination are employed in being comfortable with a book, and in being comfortable with a computer? If so, what are they?
5. How would you define computer literacy?

We received sixty replies from users at a variety of institutions, including among others the University of Texas at Austin, Columbia University, the Massachusetts Institute of Technology, Carnegie-Mellon University, Rutgers-The State University, California Institute of Technology, Harvard University, Stanford University, Lawrence Livermore National Laboratory, Bell Labs, Wright-Patterson Air Force Base, Digital Equipment Corporation, and Bolt, Baranek and Newman. Certain themes appeared over and over in our respondents' replies, but in summarizing this evidence, we have drawn generously from full responses to illustrate how contemporary practitioners understand computer literacy. At the same time, those responses we have quoted in detail have also been selected for the representativeness of their content.

Any notion of what computer training should consist of depends first of all on what computers are perceived to be. Our users chose to describe computers either (a) as tools, malleable extensions of their own wills (34 respondents selected descriptions in this category), or (b) as distinctly independent intelligences-even, in some circumstances, as opposing personalities (19 responses belong in this category). Subscribers to an "ethic of realism," as one respondent labeled the moral mode he assigns to those who refuse to mystify or anthropomorphize the machine, emphasize the tool-like computer. For these users the computer has only two states. Either "it is deterministic, or it is broken and inoperable" (programmer, Carnegie-Mellon). Put another way, the computer is "like a new kind of clay. Science and art are just now discovering what shapes it can be molded into and what the shapes are good for" (systems staff member, MIT Artificial Intelligence Lab).

When respondents describe the *kind* of tool a computer is, they often invoke tools that men rather than women use. Among 18 users who offered examples, 11 specified tools traditionally associated with men's work, 3 specified tools traditionally associated with women's work, and 7 selected gender-neutral tools. For example, "It's like a typewriter, a table saw, or a basin wrench. Actually, it's more like a toolbox or workbench when appropriately organized" (professor, Carnegie-Mellon). It was called "a logical extension of the idea of power tools, like a band saw" (programmer, Rutgers), and "a giant pocket knife with many knives and tools. It is an invaluable tool but sometimes hard to work with. Did you ever try to use all the parts of a pocket knife at once?" (systems analyst, Monroe Corp.).

Users who personify the computer often see it as an independent personality. For one user it was a "companion," since

> *I don't start from scratch on a computer: someone has already created something out there with which I have to work and which will at least occasionally frustrate me* (systems programmer, Digital Equipment Corp.).

> *Certainly each different system I "talk" to is a separate identity, and to some extent each of the major subsystems is a separate identity. I think of the editor and the compiler as different personalities, they were written by different people after all and the method of interaction is very*

different. But getting a program to compile is much like negotiating with someone for something. You give it a program and it gives it back if it doesn't like some part of it (computer scientist, Lawrence Livermore National Laboratory).

Another user explained:
When a program does something which I have taught it to do, I am pleased in much the same way, perhaps, as I would be if I had taught a person to do such things (research staff member, MIT Artificial Intelligence Lab).

Another described the computer as a "friendly helper," but added a qualification:
That depends on the computer. I would never call a TRS-80 a friend of mine (student, Rutgers).

Real people are the valid source of anthropomorphic impressions of encounters with computers-- real people who design programs, operating systems, and hardware. For users at most levels, to negotiate any of these is to take account of someone else's intellectual creation made manifest. This is rarely a transparent encounter.

The sense of the computer as a resistant other, variably helpful and irritating, also surfaced in response to our question about the power relation between a user and a computer.

Some users deny that the negotiation of power between a user and a computer is a notion with any meaning, but others concede that computer systems represent persons who are in relationships of power to them as users. For example:

People clearly control the computer. Unfortunately, if I am [directing] it to connect me to another computer and another person (more powerful than me) is [directing] it not to connect me (by disconnecting the wire, for instance) then the computer says, "Sorry, I can't connect you." It never says "because David won't let me," so the impression is made that the computer won't let me (employee, Digital Equipment Corp.).

Others described the process of transaction:

I pull the strings, but it has a pair of scissors (graduate student, Carnegie-Mellon).

I can usually get what I want, but sometimes I have to run around the block first (graduate student, Cal Tech).

Sometimes it's the classic pain of dealing with an "over-literal" child, and cursing your flawed efforts to communicate with it (graduate student, MIT Artificial Intelligence Lab).

[Some] computer systems constantly refuse to trust their users. They are programmed to act as if they always know what is best, and what is right, and will not consider changing their minds. Just like outside bureaucrats. So I don't use those systems if I can help it (staff member, MIT Artificial Intelligence Lab).

There is wide general agreement on levels of skill and areas of expertise defining broad classes of computer users.

Hard distinctions oversimplify the real state of affairs, as users constantly reminded us. But users were consistent in the distinctions they were willing to make. The two broadest hierarchical categories of skill are being able to build systems for others to use, and using systems one does not necessarily

understand. In more detailed schemata, this division characterizes the relationship of any one level to those above and below it. It is possible to specify five concrete levels of skill that most of our respondents would recognize:

1. First, there are non-users--people who have daily contact with machines without realizing that the candy machine, the microwave, and the grocery store cash register all contain microcomputers (graduate student, Rutgers).

2. Next are data-entry, casual end-users ("tourists," in their contemptuous designation), or applications users. These are non-programmers who follow fixed procedures designed by others to accomplish specific tasks. They include merchants with point-of-sale terminals, lab technicians with analyses to run, accountants who generate reports, travel agents in need of a constantly updated reservations system, civil engineers, or secretaries using sophisticated word processing (undergraduate, MIT). It is essentially between this level and the next that the debate about where to draw the criterion line for fundamental computer literacy takes place.

3. Programmers run the gamut of sophistication from "anyone who has ever written a computer program that worked" (graduate student, Carnegie-Mellon), to someone with "an extremely good model of the domain, including odd boundary cases and details" (professor, Carnegie-Mellon), that is, with advanced knowledge of data structures and algorithms. Programmers specify the procedures by which computers can be used to solve problems assigned by others. Programmers are often referred to as the middle class or clerks of computer using. They are software operatives: white-collar workers whose power is sometimes more apparent than real. They work for others, but do not make major decisions because they do not design operating systems, direct computer research, or allocate computer budgets and resources.

4. Hackers appear by name in nearly everyone's list. Hackers are a highly valued resource, knowledgeable and chaotically creative. "Nothing is sacred, no conventions have to be maintained" (undergraduate, MIT). The hacker "skillfully alters or extends the domain" of computer problem-solving performance (professor, Carnegie-Mellon). Hackers break bottlenecks and start logjams moving. The lore of hacker obsessiveness sometimes depicts them "as having below-normal social skills and concern for personal hygiene--the 'greasy nurd' syndrome" (graduate student, Carnegie-Mellon). The same inquisitiveness that enables hackers to blaze new paths through integrated circuitry can make them undisciplined programmers, but the very best of them are included in an elite called:

5. Wizards. The wizard is at the top of the literacy pyramid and has "achieved such intense competence that he is capable of amazing feats: wizards make patches, in octal, to running monitors, debug hardware by passing their hands over circuit boards," and so forth (graduate student, Carnegie-Mellon). Wizards are the most talented and respected members of several high-level expert specialties, including designers of hardware and software systems, analysts, and computer scientists.

Within this five-level hierarchy, the largest number of our respondents identify themselves as being somewhere between hacker and wizard. One user offered another summary of levels of skill above nonuser: "can design them, fix them, program them, can't program them" (high school programming teacher).

Whether there should be a hard or soft criterion definition for computer literacy--whether, that is, the ability to do simple programming in some language should constitute minimum literacy, or whether understanding "enough about computers to approach the subject rationally" (computer scientist, Lawrence Livermore National Lab), a criterion of social awareness, is sufficient--is a subject on which users have mixed opinions. Soft literacy is often defined as the state of being neither in awe nor in fear of computers. As one respondent graphically put it in specifically contemporary context:

> *The minimum I would expect in computer literacy in a reasonably well educated person today...is [knowing] that computers are complex devices which when properly instructed can do almost anything: that the instructing currently requires very painstaking and meticulous work and may not always be correct: that any one who says, "We can't do anything about that, the computer*

(followed by any asinine excuse)," should be told politely that the problems of the computer are something to be resolved by the staff on the company's time, and not on the victim's, and you will please straighten this out immediately without any further excuses (computer scientist, Carnegie-Mellon).

The same respondent went on to argue, however, that "anyone should be able to write a program," even if writing superior programs is not within everyone's capability. This is a widespread view, not least because the special logic system that a computer is cannot really be understood, many respondents feel, without some first-hand experience in manipulating it.

In general, however, respondents are fairly equally divided in choosing standards for fundamental literacy. Seventeen of our respondents believe programming ability is the bottom line for computer literacy. Fourteen respondents are willing to settle for some level of endusing skill, and 14 more believe social awareness is sufficient. A few users reject the term computer literacy entirely because it seems to focus on narrowly intellectual rather than broadly practical skills. "The analogy [should be] to tools not books," one writes (staff member, Bell Labs). These users define computer facility in functional (and frequently masculine) terms "like telephone literacy, or soldering iron literacy, or chain saw literacy." "Computers are the new automobile," suggests a respondent. "If you can't drive, you're going to miss a lot" (graduate student, Carnegie-Mellon). Consumers who use automobiles, telephones, television, and other technological furnishings of modern life without understanding how they work will not find computers any more problematic, argued another:

> *Damned few people are electrically literate, yet electricity pervades our society. The electricians have developed some simple user interfaces (plugs and switches and lightbulbs) and it suffices to know only a few trivial rules and to have someone you can call if something goes wrong* (computer scientist, Carnegie-Mellon).

While the notion of computer literacy as a practical survival skill hearkens back to the introduction of written literacy, crucial differences remain between skills in manipulating physical environments and skills in manipulating new symbol systems. Computers are very different from, say, plastic, another versatile product of modern technology. Even when computers control physical operations, their irreducible essential feature is their symbolic power.

For many respondents, the heart** of **the distinction between reading and computing is the sense** of **a contrast between the object** of **computer literacy as a text created and modified by users in real time and the fixed printed text.

While printed text seems unalterable and independent of the reader, the computer "text'' is a potential realized only through user participation and constrained only by a program that sets parameters for interaction. In two words, respondents perceive computer literacy as active, print literacy as passive. Users occasionally compare books to "television, in that neither makes demands on the user" (programmer, Carnegie-Mellon). "With a book one can sit back and let the author's ideas fill your head. With a computer you are sitting in front of a blank blackboard, and must say: 'Now what?'" (respondent, Rutgers).

In spite of the fact that contemporary literary theory has disabused us forever of the simplistically fixed printed text and the passive reader-interpreter, many of our respondents greatly underestimate the interactive encounter between a good reader and a challenging text.[1] Success in computing is associated with a level of concentration and sense of action in the world that is assumed to be unnecessary in conventional reading practice. Consider these comments from our respondents:

A book does not force you to react, it presents itself to you and allows the freedom to react. A computer demands that you react to it so it can react to you (senior technical writer, Carnegie-Mellon).

I cannot relax in the presence of a computer as I can relax in the presence of a book. To be comfortable with a computer takes great effort and patience, while a book can be ignored, read at your own pace, or skimmed (professor, Temple University).

A computer is an interacting component in a complex system of creativity and reinforcement (graduate student, MIT Artificial Intelligence Lab).

The generally held sentiment that books do not offer the same degree of creative involvement that computing does is based on an arguably erroneous comparison of users' experiences as readers of books to their experiences as writers of programs and systems.

Some respondents do discriminate reading and writing functions in both print and computing. They argue that there are similarities between reading books and executing programs written by others, and between writing books and creating programs. "For simple tasks such as; using the computer as a mere provider of data, there is no difference between reading a book and 'reading' a computer," explains one (systems programmer, Stanford Research Institute). A few users see no significant differences at all in print and computer literacy. "Both involve 'picturing' things in one's mind," one writes (programmer, Rutgers).

One of the defining features of the stored-program computer is in fact the capacity to simulate any operation that can be explicitly articulated, or made completely visible. But from its hardware to its logical procedures, none of which is easily physically observed, a computer may be comparatively invisible to its users. Computing mastery depends, therefore, on some adequate mental model of relevant computing operations.

A number of accounts of teaching software use suggest that visual mental models of information processing spatially represented in three-dimensional arrangements of letters, numbers, and items are common among novice users (9). The use of spatial imagery to describe interaction with computers is also common among our respondents.[2] Besides demonstrating in operation an imaginative modality that is apparently quite useful with computers, the spatial character of many of these descriptions also suggests the intense intellectual and emotional absorption of their authors in computing, which demands that users "enter its world" (graduate student, Carnegie-Mellon). According to one such description:

A suitably rich computer system is a universe to explore, conquer, change and live in (part of the time at least).... In some of these universes I am a stranger and easy prey for the hostile local spirits; in others I am a serious explorer; others I know intimately and in these I am a formidable wizard--if a mountain is in my way, I move it. Some universes are unfit for civilized habitation and I stay out of, them unless someone pays me a lot to go in and zap something (computer programmer, Carnegie-Mellon).

In this account the computer constructs the user's operating world. In an alternative configuration the computer itself operates in a world of shifting spatial contours:

It would probably be like a car with a small but extremely powerful engine where the outside of the car could change as the terrain required. Sleek and fast for the long flat stretches and high and rugged for climbing over trees and rocks like a jeep. With all that power I could drive the car anywhere, even underwater… If it has power steering l can even get the job done without having to exert much effort (systems analyst, Monroe Corp.).

Visual spatialization characterizes descriptions of hardware--"a row of dominos, set up to do all kinds of tricks that seem intelligent (spelling out words, etc.) but which obviously have no innate intelligence" (graduate student, Carnegie-Mellon)--definitions of computer literacy--"the ability to be able to find one's way around a computer with minimal effort" (programmer, Computer Performance Co.)--and discussions of programming--"I imagine what I am going to build with the computer, what I am going to do" (hardware designer, Raytheon Data Systems).

We come finally to a discussion of the politics of computer literacy, that is, to a more specific consideration of the institutional and social settings in which computing takes place.

Because of the growing concern of educators and others with the expansion of computing into so many aspects of our lives, there are now several widely current,E broadly similar definitions of computer literacy in circulation. One of the best known comes from the Duke University Personal Computing Project, a university-wide effort aimed at acquainting Duke students with computing skills. The Duke Project defines computer literacy as "an awareness of computing capabilities within a discipline or profession, and an ability to recognize and articulate problems that can be solved with the aid of computing technology. The definition does not necessarily imply an ability to program or operate computers" (4). This definition strives admirably to make the computer age a common resource, but it evades the real issue. No one accepts awareness of what print literacy can do as a substitute for the ability *to* read or write. In a similar way, much of the social alarm about our most popular contemporary mass medium, television, concerns the transmission of vicarious experience to passive viewers. It would be unfortunate if our social model of the computer were a kind of television programmed by others. Warner Communications' computer-interactive QUBE cable television system in Columbus, Ohio, in which computer-connected viewers become members of an extended game-show audience, suggests that this familiar impulse will be a powerful one.

If we recognize that social awareness of computing does not by itself provide a personal command of this symbol system sufficiently powerful to make it a tool that ordinary citizens can use to their benefit, we must also recognize that whatever social definition of computer literacy we choose will be enforced only at the level of economic commitment that our society is willing to make. If that level is low, as it now is, we will have only two options: to train a powerful but tiny elite of highly skilled computer literates, or to diffuse widely a low-level computer literacy without true social power. Even if we were willing to commit social resources on the scale required to educate all citizens in the rudiments of computer literacy, the very size of this task would preclude most training above elementary end-using and programming skills for a long time.

In the meantime, broad computer literacy training is already institutionalized in comparatively elite sectors: computer science departments in colleges and universities, other schools with special financial resources, certain sectors of private industry, and government research and administration. Those not formally situated within this web of power may slip through the cracks of the system to acquire very powerful computational skills, but these haphazard occurrences are the most unreliable and certainly the least democratic option for advancing the diffusion of computer literacy.[3] It is still the case that the faculty child will be more likely than the ghetto child to find a way into the computer center of the university where both their parents work.

While some argue that computer technology itself will effect the conditions necessary for social change with plentiful supplies of cheap and powerful microprocessors the existence of cheap books has not been enough to keep our own population fully literate without an extensive accompanying commitment to public education.

It is also important to recognize that a microprocessor is not a true communications medium until it is attached to networks of other users and computers by landline, satellite, or optical fiber, none of which is inexpensive. Even stand-alone microprocessors suitably equipped with the latest and best in software packaging now represent an investment of between $1,000 and $2,500 apiece, an amount well beyond the resources of many families.

Even if that price comes down in the future, inequalities of great consequence are being created in the meantime. How much prices will come down depends on the continuing capacity of new computer technologies, particularly advances in miniaturization, to increase the productivity that provides the major economic reward of technological innovation. The limits of miniaturization are unclear, but that there are such limits is certain. Despite gains that have been made to date, no evidence suggests we will soon be awash in unused computer capacity--nor, given the history of most developments in communications, are we *ever* likely to be. Innovations in communications technology have always been advertised as far surpassing the capacities of previously existing technologies for transmitting and storing information. But demand for more capacity has always caught up, and rapidly, with each new technology. New communications capacity generates its own rising demand, or to say it another way, its own scarcities. Powerful computer systems will continue to be consumers of costly resources and to require cadres of expensive experts for logistical support. Even the most public spirited network, a network not now on the horizon, would be unthinkable without its own technocratic elite to maintain and therefore, inevitably, to govern it. Some computer literates will always be more equal than others by virtue of their knowledge and ability to decode and recode the system and their value to those whose hands are on the switch.

The possibilities for institutionalizing new forms of political and economic inequity in the emerging social organization of computing, and for exacerbating old ones, cannot be wished away by recourse to arguments about electronic abundance. A large part of the social organization of computing has already been fixed and will continue to be shaped by the diffusion of computer literacy in all its varieties. To surrender by default and without citizen consultation the personal right to manipulate these powerful new symbolic systems that have permanently entered our public, private, economic, political, and recreational worlds by inattention to the need for effective and equitable access to the intellectual and material resources of computing is democratically unacceptable. Diligent and thoughtful planning for the significant extension of computer literacy will be necessary to guarantee for the greatest number of citizens the considerable opportunities that computing offers. The alternative will be an institutionalized form of elite domination sure to take its toll on democratic life and values.

REFERENCES
1. *Adult Functional Competency: A Report to the Office of Education Dissemination Review Panel.* Austin, Texas: Division of Extension, University of Texas, 1975.
2. Bott, Ross A. "A Study of Complex Learning: Theory and Methodologies." Center Human Information Processing Report #82. Department of Psychology University California, San Diego, March 1969.
3. Clanchy, Michael. *From Memory to Written Record: England, 1066-1307,* Cambridge, Mass.: Harvard University Press, 1979.
4. Gallie, Thomas M., Charles H. Lochmuller, Sharon S. Poss, Mel Ray, and John Rousseau: "The Duke Personal Computing Project: Its Strategy for Computer Literacy. *Perspectives in Computing* 1, October 1980, pp. 4-8

5. Graff, Harvey J. *The Literacy Myth: Literacy and Social Structure in the Nineteenth-Century City,* New York: Academic Press, 1979.

6. Havelock, Eric. *The Literate Revolution in Greece and Its Cultural Consequences* Princeton, N.J.: Princeton University Press, 1982.

7. Koerner, James D. (Ed.) *The New Liberal Arts: An Exchange of Views.* Occasional paper. New York: Alfred P. Sloan Foundation, 1981.

8. Marvin, Carolyn, "Fables for the Information Age: The Fisherman's Wishes." *Illinois Issues* 8, September 1982, pp. 17-24.

9. Negroponte, Nicholas *et al. Spatial Data Management.* Cambridge, Mass.: MIT Press, 1979.

10. Yates, Frances. *The Art of Memory.* London: Routledge & Kegan Paul, 1966.

Note

[1] The widely accepted view of a passive relation to a fixed printed text has an interesting echo in research about how naive users learn an electronic bibliographic retrieval system, One of the most difficult steps is to convince the novice that he or she, not the computer, is in charge of initiating and maintaining interaction (see 2). The tendency of naive users to seek direction from the computer may reflect their accustomed superficially passive roles as readers of printed text or viewers of television.

[2] Spatial models of information are not unique to computing. In the traditional memory system of classical Greece, one of the oldest information retrieval systems we know of, items for recall were mentally visualized in specific architectural spaces. The spatialization of information is also a structural feature of every typewriter keyboard and of the very concept of moveable type (see 10).

[3] For a more extended discussion of the tension between democracy and computer literacy, see (8).