LOGO - THE EDUCATIONAL LANGUAGE OF THE FUTURE

*Narayan Ugar

The relevance of computers in the classroom

Schools are in the grip of a computer mania. Educational computing, like the force, is with us. Microcomputers are proliferating in our schools, and unless lot of people are wrong, there is no doubt they are here to stay and they should. No one disputes any more the need for making our children computer literate and able to cope with the rapid technological changes in our world. Ordinarily conservative and slow to change, schools are embracing new technology and new educational methods more rapidly than they can learn to use them. The influx of microcomputers in education is growing from a trickle to a torrent, engulfing teachers and administrators in a flood of confused expectations and unfulfilled promises. Changes are taking place so rapidly that it's not at all clear as to who is in charge. Are the computer manufacturers and software publishers pushing computers into the schools? Or do school administrators, school boards, or individual teachers have the ultimate responsibility for making decisions about purchasing and curriculum? The frightening answer may well be that no one is in charge! There's no master plan, no one at the helm - just innovation for its own sake. The 25000 rupees question is whether these computers will make any difference to the education of our children. When my grand daughter passes out from higher secondary certificate examination in the year 2000, will she have received a better education with the help of computers than I did without them?

I don't have the answer to that question and nobody else does either. Nevertheless, there are questions that need to be asked about computers in the schools. Defining them may help to outline the deliberations that may eventually determine the course of educational computing.

All of us feel we can't have a school of excellence these days without computers. But the real question is what are we going to use the computer for? It's important to recognize that books can be salacious literature of the most dubious sort for children; on the other hand they can be powerful learning tools. It is the same way about computers. There is a big difference between saying computers have the CAPACITY to enhance children's learning and that they automatically WILL enhance children's learning. It's a hard question to figure out how children truly can learn more as a result of using microcomputers. Although we might be skeptical of the

* Systems Consultant Nalini Foundation for Symbolic Logic Saavitri, Nirmal Baag, Pune.

hoopla, we may be cautiously optimistic about the long-term potential of the microcomputer to enhance learning. The questions now revolve around HOW to do it, how to choose the best software programs from among the myriad out there, ones that will truly teach our children and - just as important - that will NOT discourage any of them from learning to use computers.

Pressures from many sources are combining to push schools rapidly and irrevocably into the information age. Hardly a day goes by without a story in the media about how technology is changing the society, or how there are not enough people for technical jobs, despite record unemployment. A THOUGHT FOR TODAY on a schoolboard said: "If you think education is expensive, try ignorance." One of the sad truths today is that it's easier for a school to get approval to purchase a bunch of computers than to use that money to retrain good teachers.

Additional pressure comes from the marketplace itself. Hardware and software vendors and educational publishers are all moving aggressively to capture a lucrative piece of action. Rather than slowing down and carefully examining their long and short term goals, schools are scrambling for the latest technology, the most elaborate systems, and the lowest hardware prices without fully knowing what they are going to do with computers when they get them. Although it is almost universally acknowledged that educational software is still in its infancy and that most available software is inadequate and unimaginative, schools have little choice but to purchase and use the software that's available.

Educational publishers, who traditionally take at least five years to develop and test a new line of textbooks, are rushing inro the educational software market, hoping to replace the standard type of workbook with electronic means of helping Abhijit and Anuradha practise their spelling and arithmetic lessons.

Courses on computer literacy and computer programming are proliferating. Again, the course directors and publishers are competing to be first. No one quite knows what it is, but everyone is sure that it's good for us. In the final analysis, it is our schools that must bear the brunt of educating a technologically sophisticated population of workers for high-tech jobs. But is the current wave of computer literacy courses really what the next generation needs ? Do elementary school children need to practice KEYBOARDING or learn to write simple programs in BASIC in order to grow up computer-literate? Do these courses give our children the conceptual basis for solving complex problems and handling large systems of information ? The question is largely unexamined.

There is often a talk about lack of motivation in schools. It is pointed out that when computers are used, children ARE motivated, they like to come to school, they like to play with computer, they stay after school hours. Joseph Weizenbaum, Professor of Computer Science at the Massachussets Institute of Technology, creator of the psychiatrist mimicking computer program ELIZA, and an outstanding critic of what he considers inappropriate use of computers, says that if the intention is to increase motivation amongst students, there are lots of other things that can be done. For example, suppose we showed pornographic movies everyday at lunchtime or gave out small samples of heroin once a week or once a day. Of course we don't do these things because it isn't just motivation as such that we want, we want a RELEVANT motivation, motivation to read, for example.

Another source of pressure comes directly from parents. In ever increasing numbers they're sending their children to computer courses and buying computers for their homes. This will tend to widen the cultural gap between computer-clever and computer-silly children.

Software for people of all Ages

For many people, word processing is the primary, if not the only, reason for buying a computer. But learning to use some of the powerful word processing programs can be difficult. Hefty manuals must be studied and referred to time and again; numerous commands have to be committed to memory. With some programs, the documentation is so ponderous that special paperbacks are being written and marketed to supplement the manuals of the more popular products.

Creativity becomes secondary when someone is first mastering the intricacies of a software program. If adults experience frustration, think of how young children must feel when they first try to use a software. Feature-laden programs tend to complicate rather than simplify writing. This sophistication versus ease-of-use trade-off also holds true for database- management software, accounting packages, spreadsheets and other types of programs. A software with its straightforward design gains a wide following. PROGRAMS THAT ARE EASY TO USE, SELL.

Logo

Like BASIC, LOGO is a computer language. One explanation for its increasing popularity in the classroom is that LOGO is more than just another programming language - it also represents a philosophy of education. LOGO is better than BASIC in three ways : by using LOGO, you can more easily create graphics, build LIST OF LISTS, and design your OWN computer languages (by adding your own words to LOGO). In the world of programming languages, LOGO may someday emerge as a universal first programming language. LOGO's suitability for structured programming and modular problem solving may also lead to its use in introductory computer science course, easing the transition to more complex languages like PASCAL and LISP.

Papert, The explorer of Logo power

Although Logo is a fairly new language for microcomputers, its development anduse actually go back to 1968 or earlie. Logo is a first cousin of LISP (an acronym for LIST Processing), an artificial intelligence language.Children fall in love with written language because they see it as something their fathers do. For students who do not have this kind of relationship, something else is needed. The problematic personality isn't the one that can't eventually adapt to the written language but the one for whom our culture has not provided any mode of access' they don't have any way of making contact with it. The computer is a catalyst, a bridge; it's a matchmaker to get them together, and once togther they don't need a matchmaker any more.

Originally developed at Bolt, Beraneck and Newman,a Cambrige, Massachusetts, high technology consulting firm, Logo had been used in research laboratories around the world for a number of years before a microcomputer version became available. Many individuals influenced Logo's development, but Seymour Papert is most prominently recognized as its primary conceptualizer, developer, researcher, and public advocate. For many years, Papert was director of MIT's Logo group, and his research efforts during that time made Logo into a practical and effective educational vehicle. Also, Papert's influential MINDSTORMS (Basic Books, 1980) paved the way for Logo's growing acceptance among computer users all over the world. Papert proposed that the computer could become a learning vehicle by allowing interaction with objects environment. in Logo, a small movable triangle/figure on the computer screen, called the turtle becomes the vehicle between child and object. Papert was explicit when he stated that the turtle is not a cure for the problems we find in education, but is a " ... model for other objects, yet to be invented. My interest is in the process of invention of OBJECTS TO THINK WITH, objects in which there is an intersection of cultural THINK WITH, objects in which there is an intersection of cultural presence, embedded knowledge, and the possibility for personal identification."

Papert's colleague, Swiss psychologist Jean Piaget, also emphasized that children learn best not from listening to lectures, but by acting upon objects in their environment. Logo frees the students to operate the computer in self-directed ways. They establish their own problems, tinker with different solutions and build on what has gone before. Logo combines abstract reasoning and actual experience, avoiding many of the conceptual difficulties of learning to program. Best of all, particularly with young students, Logo is an interactive language-student and computer work together in what could almost be termed a personal relationship. The language is learned with the help of the computer.

Papert and his colleagues at MIT have thus given us Logo to explore alternative methods of helping students learn. Papert stated that the turtle was only a model for the future development of objects which allow children to interact more with their environment, thus facilitating the development of their intellectual abilities. If this philosophy is to move into the classroom, it will be through the elementary school teacher. Papert's philosophy of education combines the theories of Piaget with programming approaches developed for research in artificial intelligence. Programming technicques like recursion and list processing were derived from the artificial intelligence language LISP and have been integral to Logo since its conception.

Part of the Logo's strength comes from the fact that it uses the computer as a control device. Logo users can control an external robot called a turtle, create animated cartoons, invest video games and interactive quizzes, or compose poetry or music. By making the action of the computer concrete, Logo builds a bridge between abstract reasoning and actual experience, thereby sidestepping many of the conceptual difficulties people have in learning to program a computer.

The Turtle

Piaget's research stresses the necessity of children being involved in physical manipulation of objects to build their intellectual structures. Spatial relationships become meaningful to children because of their interacation with the environment. Children develop concepts of distance because of their own action and involvement in movement. Self-involvement by children in spatial activities facilitates understanding movement of the turtle in Logo.

The best-known Logo programming environment, TURTLE GEOMETRY, gives you a simulated turtle that is used as a sort of pen to draw on the computer's video display screen. Turtle geometry begins with a small, triangular or turtle-shaped, cybernetic figure on a video screen. This figure is called a TURTLE. During the development of Logo, the original turtle was a small robot that was electronically programmed to move about the room. The robot became the figure on the video screen. Using Logo, students enter commands for the turtle to move on the screen, drawing lines or not.

The turtle is controlled by simple commands like FORWARD, BACK, RIGHT, and LEFT. These commands can be used in combination to make the turtle create virtually any geometric shape. to program in Logo you teach the computer new commands called PROCEDURES. Once the procedure is defined, it becomes a new command in the language and can be used as a SUBPROCEDURE to create more complex procedures. Logo's ability to create complex geometric shapes from simple procedures makes turtle geometry a powerful tool for creating graphics with the the computer and for learning geometry in a way that was never before possible.

Following the ideas of Piaget and Papert, the programming activities of children need to be preceded by real spatial movement activities. Students must identify strongly with the turtle as it turns, rotates and darts about the screen, Hence, the teaching strategies have to address the child's ability to use self as a point of reference in spatial relatioinships. The younger the child, the greater need there will be for PLAYING TURTLE. As more complicated procedures are designed for the turtle, a turn in the wrong direction can produce very different directional movements than were planned. When the BUG (an error, in computer jargon) becomes difficult to find, teacher should recommend that students PLAY TURTLE. Errors can be discovered more easily when students walk out the commands.

Although turtle geometry seems simple at first glance, beginners of any age may find creating drawings quite challenging. To draw a sketch of a person, a learner encounters many of the key ideas in geometry, algebra, and general problem-solving as well as important concepts of computer programming. Far from being limited to introductory activities, turtle graphics can incorporate the complex mathematical concepts of trigonometry, calculus, and topology. A good college text on the subject is Tutle Geometry, by Harold Abelson and Andrea disessa (MIT Press, 1981).

Zooming with Sprites

Another introductory Logo activity is programming with SPRITES - small, user-defined blocks that can be made to move rapidly across the screen. Sprites are like the turtle in that you can command them to move forward, back, right, or left. You can also give sprites certain speed and make them zoom across the screen. They can take on different shapes and colours, and several sprites can appear on the screen at once. Just as the Logo turtle is the vehicle for creating geometric line-drawings, sprites enable you to create animated scenes and video games.

Sprites respond to built-in Logo commands such as SETSPEED, SETCOLOR, and SETSHAPE. One common introductory activity with sprites is to create abstracat patterns bursting with colour and motion. Another is to animate one sprite by making it to go through a succession of shapes. To simulate a flying bird, for example, you create two shapes representing wings in different positions and then write a procedure that makes a moving sprite change repeatedly from one shape to another, giving the appearance of flapping.

Creating animated scenes with sprites gives full rein to a child's imagination and forges a strong link between that child's fantacy life and the power of a computer to make things happen. Because their implementation depends on capabilities built into a computer's hardware, sprites are not available for all versions of Logo.

Paddles and Joysticks

Descriptions of Logo that stop with turtles and sprites are missing a lot of action. Logo also includes commands that accept input from the keyboard or from game paddles or joysticks. These interactive programms range from simple quizzes to complicated action games that use sprites and turtles as players as in tic- tac-toe, racing the turtle around a maze, or a game like guessing a number.

Composing Music

Composing music is another Logo application that was developed at MIT and is now becoming available for microcomputer versions of Logo. Musical phrases are built from procedures just like turtle drawings or sprite animations. You can define each part of a tune as a separate procedure and experiment with different arrangements, rhythms, and harmonic combinations. This procedural approach to music could have as large an impact on the way music is taught and learned as turtle geometry is having on the study of mathematics.

List Processing

The list processing mode of Logo allows students to explroe the manipulation of words and sentences. Students are able to create lists of information that have meaning for their age group. They can make lists of the books they have read; lists of their friends; lists of their math scores. Through simple recursive commands, they can add to or delete informatoin from the lists. Information can be passed back and forth among lists, and arithmetic operations can be performed. By creating short procedures, students can manipulate information that is important to them. While working with list processing, they are developing skills to categorize and structure sentences and to gain insight into logical branching.

In a Logo environment, students are free to explore and interact with the computer using the turtles, sprites and list processing capabilities. While interacting with the computer, their discoveries strengthten their developing knowledge bases. By integrating the knowledge students already have with new ideas from other curriculum areas, the classroom can become a Logo environment that facilitates a child's intellectual growth. Thus, Logo is not an isolated subject called LOGO or LEARNING ABOUT COMPUTERS. Logo is a language for learning subject matter, developing skill areas, solving problems, building intellectual structures, exploring language and communication - and is just plain fun.

Artists, software developers, psychologists and educationists, thus, speak glowingly of the potential for computers and creative expression. At the same time, we do not identify one example of using a computer in an art class as part of the regular curriculum. There may be many reasons but the suspicion that many art teachers have simply not yet joined the computer age appears to be true. The greatest trap in any educational software is to try to remould an old curriculum in this new medium. If we look at video games, for example, we can see wonderfully complex things going on between children and computers. What we have to do now is ask what we can do with computers that we can't do in art education with other media.

All said and done, art classess may be the last places where we will see computer creativity. The problem may have more to do with attitudes toward art than toward computers. In school, art always has to be in service of something that looks like WORK. There is a systematic avoidance of pleasure, of fun, of fantacy. Those are the things that make art wonderful, and there doesn't seem to be any place for them in schools.

The limitations of Logo

Despite its powers, Logo has some disadvantages as a programming language. First of all, Logo is an extremely complicated system that fills a large part of your computer's memory. DR Logo from Digital Research, when implemented on the IBM PC computer, should be able to access upto 1 megabytes of RAM. Other limitations stem from the graphics capabilities of different computerse rather than from Logo itself. On some computers, the lack of screen resolution diminishes the effectiveness of turtle graphics by displaying thick, jagged lines where you would prefer to see thin, straight ones. It can also be difficult to tell exactly where the turtle is pointing - another problem caused by limited resolution. As with speed and memory, graphics limitations should become less significant as microcomputers become more sophisticated.

***