we have studied, and it seems to greatly reduce initial problems. Equitable sharing of the computer time (as described in our account of the computer in the school) is another important ingredient in assuring success. Of course the computer will have to be placed in some corner of the classroom, and the volume of its speaker (if one exists) reduced to a minimum.

Initial excitement and questions of scheduling could thus be minimized. More important questions involve, though, the use of games and their potential benefits. Are there not many useless games which children will be attracted to? What about the danger that the weakest children, the ones needing other class activities most, will also be the ones most eager to spend time "playing" rather than "studying?" Before we move to a discussion of how games may be used as part of the more general curriculum we would like to describe a study involving one game. The "study" draws both on observations made of the use of that game in a classroom and at home, and on our analysis of how the game may be further used in other classroom activities.

ROADRACE - a case study of a "useless" game

ROADRACE is a race game available for the Apple II computers we have been using throughout our studies. It is commercially available, similar to other game programs for the Apple and other personal computers. The game is quite simple: The player becomes the driver of a car driven on a race course. Control of the car is achieved by means of two game paddles: one of them serves as the steering wheel (controlling left and right movement), and the other serving as the accelerator, controlling the speed of the car. At the outset of the game the car is positioned at the bottom of the computer screen, stationary until the driver turns the accelerator to start "moving." Actually the car remains stationary, with the winding racetrack changing, but the illusion of movement is quite strong. The course has some straight sections, but also many sharp turns. The game lasts for fixed amount of time (about two minutes), and a player's score is determined by the distance covered and the degree to which s/he managed to stay on the course. In addition to the car and the course the player can see at the bottom of the screen the points s/he has accumulated up to that time, the "speed" of the car, and time elapsed since the beginning of the current race. Before starting a new race players can choose to play it on a standard or a new course, and select the level of difficulty of the course (i.e., the number of sharp turns). Overall the game is not unlike many race games available in game arcades.

ROADRACE was very popular both among the fourth-graders we observed at school and among the younger children we have observed at home. Many adults play it for quite a while. Malone (1980) gives some insight as to why this game is so popular. Malone analyzed the attraction of computer games, and identified three aspects which make a game attractive: challenge, curiosity, and fantasy. ROADRACE is certainly high on challenge. It is very difficult to drive the car at maximum speed on the most difficult course without ever driving off the road. In other words, there is a very challenging maximum performance; at the same time this leaves ample room for improvement before reaching the top, providing the player with constant feedback as tc his/her current score and improvements in it. The game would probably rank low on curiosity value, since it does not include many "surprises." Yet even there the game is not completely predictable, since even for the standard courses, it is difficult to remember the sequence of turns. The fantasy element is there, of course. The player is a race car driver, going at a maximum speed of 164 mph (the game wouldn't be as much fun if the everything stayed the same except that the maximum speed was 55 mph, would it?).

Children love ROADRACE. Isn't it just another proof that computer activities will only be a distractive and destructive force for the normal process of education? We do not agree, and use ROADRACE to illustrate our point exactly because it looks like such a valueless activity. We feel that the game has intrinsic educational value. First, it is an excellent means for teaching hand-eye coordination. The child has to learn to control two paddles, each having a different function, in relation to the external constraints imposed by the course. In addition, the player is learning to integrate two separate streams of information (speed and position). As another benefit, the game provides continuous, effortless, and highly motivated practice in naming numbers which are in the thousands, by reading their scores. Most children know these numbers by the time they are in fourth grade, but some don't. And the range of numbers could be easily changed to fit various ability levels. It could be divided by ten (while still keeping only integers) to provide practice with hundreds, or it could be multiplied by ten to provide practice with larger numbers. Decimal fractions can be easily introduced into the score with the cover story, say, that the score is the number of miles they have traveled. The score could display, of course, both the miles and the kilometers travelled if one is interested in giving children a good intuitive feel for the conversion between these two units of length (speed could also be simultaneously displayed in mph and kph). It is even possible to start the game with a large number of negative points, with the object of the game being to end up having as few of them as possible. Here children will have the opportunity to observe how negative numbers become "smaller" (in absolute value) as they become larger (in algebraic value).

Another possible benefit of ROADRACE is that it may help increase the span of attention of its players. Teachers often complain that some of their students cannot concentrate on any undertaking for more than a few minutes at a time: "If I could only get him to sit down and do <u>anything</u> for ten minutes in a row." Well, one can get almost any child to play ROADRACE for fifteen or twenty minutes in a row. As a matter of fact, the problem may be to take the child away from the game.

So far we have described ROADRACE as if it were done in isolation by one child. But children communicate with each other, and compare their scores. Here another positive aspect of challenging computer activities manifests itself. Given the variety of available activities, almost each child stands a good chance of becoming one of the best in the class in one or more of those activities. Johnny is one such case. Johnny was one of the children in our classroom who never excelled in any scholastic activities. As a matter of fact he was in the lowest reading and math groups. He was well liked since he was always ready to help other children, but never highly thought of as a scholar. One time, when two other children from the class (both top students) were playing ROADRACE Johnny stood there and watched them. They were discussing their scores, which were somewhere above six thousand, when Johnny commented that his scores were over eight thousand. They turned to him in disbelief. "Well I have a method for doing it." This made them even more increduluous. Johnny then explained that he didn't look at the car (which is what most people do when playing the game). Instead he watched the top of the screen (which is where the course was "coming" from), and shifted the paddle accordingly. By concentrating on the top of the screen Johnny was able to get another split second to respond, two thousands extra points, and the esteem of his classmates. We believe that it is very important for children to have the sense that they are the best, or at least very good at some school related activity. It is probably a terrible feeling to go to school day after day knowing that you are going to be one of the worst in every activity engaged in. Computer activities, by the sheer fact of their diversity, increase the chances that each child will find one fitting his/her abilities. Practice on some infrequently used game may further increase the chances that any child will be the class expert on one or other activity. We feel that such a sense of power and expertise may be a turning point for many children in their view of themselves in relation to school and other children. Similar observations concerning the positive effects of mastery of computer activities on low achieving children are reported by Papert, Watt, diSessa, & Weir (1979).

It is important for children to learn to cooperate with other children. We have already discussed the often expressed concern that the introduction of computers may have an adverse effect on peer interaction; we also noted that, if anything, such interactions seem to increase in number and quality (they often involve constructive criticisms, explanations, or useful hints) as a result of having a computer in the classroom. ROADRACE provides a concrete example of how such interactions come about. The children played many rounds of this game in teams with one controlling the "steering wheel", the other the "accelerator". This not only gets twice as many children involved with a given limited number of computers, but also provides an intensive workshop in social interaction. If you do not cooperate while you are in charge of the acceleration (and it's really easy to be nasty under such conditions) your teammate will do the same to you two minutes later, when you switch paddles and roles. There were many initial cases of lack of communication or help between pairs of children playing ROADRACE together, but the advantages of cooperation were learned very quickly.

So far we have only discussed the potential benefits of playing ROADRACE. It is clear, though, that the limited availability of computing resources will necessarily restrict the amount of direct interaction between children and personal computers. Still, the motivation and interest generated by computer activities can be easily used as the focus point of many other scholastic <u>and</u> enjoyable activities. Let us continue for a while with the

ROADRACE example. We have noted before that performance on the game tends to improve with practice. We suggested to the children in the classroom that they keep track of their performance by writing down on a score sheet the trial number and their score on that trial. Now, it does not take much to interest the child in graphically representing their scores. A piece of graph paper, with the columns serving for trials and the rows for the score will do splendidly. In this way, the children learn about X and Y coordinate systems. There may not be enough rows to represent each individual score (remember, the scores are in the thousands), so they learn about scaling.

Avid players of ROADRACE will also soon run out of columns for trials. They can add another sheet of paper. But when the paper is a few feet long it becomes hard to use. They can take two adjacent trials and combine their scores. One nice rule might be to add the scores of those trials and divide them by two. This is a way for children to start learning about averages. In due time they will run out of space even with the use of pairs of trials. By then one of the children may suggest groupings of three or four. If the scores in the game are small numbers, then when calculating the mean of a number of trials the question of fractions emerges in a natural way. What's the mean value of 7 and 10? Thirds and quarters can follow suit, providing a basis for a more general discussion of fractions.

As trials and scores accumulate one can start discussing the shape of the emerging curve. Is it a straight line? Is it smooth? Why is it jagged at some places? What happens after we have combined a few trials together? What if we take the score of all children in the classroom and plot mean performance as a function of trials? Is the mean value a good measure? Maybe we should also indicate the degree of dispersion around it? Issues in statistics, measurement and measurement errors, and learning (the acquisition of skills) all can emerge very naturally as topics for class discussion. Were teams doing better than individuals? What about standard race courses versus novel courses? Has anybody already modified the program? How did that affect performance? What other factors could affect performance? Do children who play early in the morning do better than ones playing towards the end of the school day? Properly guided ROADRACE players can soon get into a host of questions involving testing of hypotheses, experimental design, random assignment, the control of variables, and elementary statistics. Children can volunteer ideas, class discussion can follow, and soon there can be a number of projects all centered around the game in question. Of course, some questions will require the introduction of new computer activities since by the time the hypotheses have been formulated the subject population had been contaminated in regard to the original activity.

ROADRACE is but one example, and a fairly typical one, of a computer activity and its potential benefits. Similar observations and analyses could be applied to many other, equally "useless" activities to come up with other ideas about how these activities can enhance education in its broadest sense.

Summary. In this section we have discussed in some detail a sequence of activities expected to improve a learner's knowledge of a particular knowledge domain, and how initial preoccupation with a motivating computer activity can be used as a springboard to a host of other educational interaction. We believe that the approaches outlined here can be fruitfully applied to many fields of knowledge. The sequence of playing within, then modifying, and finally creating a mini-world involving some phenomena is both highly motivating and a good way of gaining in-depth knowledge of subject matter.

Breadth of Educational Computer Activities

In this section we describe different kinds of activities which are available or can be easily implemented on present-day microcomputers. Many educators are unaware of the wide range of activities possible on the computer, and of the educational goals which may be achieved by using them. Variety in computer activities is necessary not only to achieve different goals, but also to accommodate individual preferences for such activities. In a recent survey of the popularity of computer games Malone (1980) found that even the most popular one was regarded as such by only 17% of the children. Furthermore, even when they have clear preferences for certain activities, people using computers like to engage in a variety of them. For an educational computer system to succeed it should offer learners a wide variety of activities. The purpose of this section is to illustrate some of the variety of educational computer activities.

Simulated worlds.

An important educational use for computers is to allow students to explore a domain of knowledge through interaction with a computer simulation of that domain. For example, there are air flight simulation programs for personal computers that present on the computer screen both the view out the airplane cockpit window and the instruments on the control panel. Within this simulated "world" a person can take off, fly around, land, even crash. Similarly, there are simulated worlds for learning about ecology, physics, psychology, chemistry. Student can safely conduct dangerous physics experiments; students can move around through complex organic molecules, viewing them from different angles; students can examine the policy implications of different government actions by experiencing their effects on simulated populations.

The exploration of simulated worlds can be valuable for learning, but often learners need additional organizing structure to maintain and focus motivation. A game format can provide this structure, and many computer simulations serve as the basis for computer games.

Many parents and teachers decry the current popularity of computer games on the grounds that children are spending so much time refining skills and acquiring detailed knowledge that can only be used in some fantasy world. However, it may be possible to capitalize on the immense popularity and power of these game situations for educational purposes, by building the games around accurate simulations of knowledge domains that the children will find useful outside the scope of the particular games.

Educational adventure games

Children who play existing adventure games learn in great detail the geography of the fantasy world (what cave rooms are connected to which, etc.), the various dangers and ways to combat them, the location and value of treasures and other objects found in the fantasy world. These game worlds are very flexible, though. In Dungeons and Dragons, for example, one of the players serves as the "Dungeonmaster", and is assigned with the task of creating and maintaining the particular world within which a game is played. Educational adventure games can be built and played within simulated worlds that reflect knowledge we want the players to acquire.

Geographical adventures. One of the advantages of historical novels or adventure narratives is the knowledge that the reader acquires as a side effect of reading the book for entertainment. We learn about India or Afganistan by reading Kipling. Similarly, the computer adventure games could be set in some part of the world, so that the geographical knowledge acquired while playing would carry over beyond the game. Children learn about caves, dragons, and arbitrary treasures in the current adventure games. A "Spy" adventure game could be set in Europe, with children learning about countries and cities, historical figures, and political events instead. Some children could act as GameMasters, by setting up the dangers and resources in the game world, and would have to do their "research" on the geography and history of Europe to create a realistic game.

Another possible set of games would involve an Election adventure, in which the players take the roles of candidates and their campaign staffs, and try to win an election. The game can be customized to the country, state, or district of the children, directing their attention to the geography and important political issues of their own area. Each player would have to select positions on issues and plan the campaign such that the candidate gets to speak before the appropriate groups to gain their votes. Given limited time before the election, the campaign staffs would have to pay close attention to the geography of the area, so that they can plan a campaign tour that minimizes travel time. In this mini-world children would acquire both geographical and social science knowledge, as well as important planning and problem solving skills.

So far, we have given some examples of how computer games can be used to teach geographical, historical, and social science knowledge and skills, as a by-product of their use for entertainment. What about more abstract knowledge and skills, such as those required in chemistry or biology? Could we imagine an "adventure" game involving the periodic table of elements or the evolutionary tree? Well, let us try.

Chemical Adventure. Many people are intrigued but puzzled by the periodic table of elements. They have seen the imposing chart hanging from the blackboard of a chemistry classroom, but it remains a strange, unknown world.

Recall that the table is arranged in such a way that similarities in properties are reflected by the proximity of the elements in it. The dimensions underlying the organization of the table are so well specified that once the table was proposed it was possible to describe the properties of "missing" (i.e., then unknown) elements.

Suppose that in a game world, we personify elements as people having characteristics analogous to their namesake elements. So we would have the muscle men Chromium, Manganese, and Iron, the attractive Chlorine, Fluorine, and Iodine, the casanovas Lithium, Sodium, and Potassium, the super rich Platinum, Gold, Silver, and Copper. A goal in this game might be to rescue Silver, who is being held hostage by the seductive Chlorine (the compound silver chloride, used on photographic paper). To carry out this mission, the player has to obtain a magic lantern that will distract the beautiful but dangerous Chlorine (bleached blond hair, long flowing green gown). Then the player could use a magic powder (free electrons) to sprinkle over Silver to reduce his attraction to Chlorine, so that he can be set free. (It could make a great film.)

Along the way the player would have to avoid the dangerous Arsenic and Plutonium, distracting Arsenic with Gallium, or using Lead as a shield from Plutonium's rays. The players could try to gather together a number of members of the Carbon family and the Hydrogen family (a light headed bunch) to form The Sugar Company, held together by bonds of friendship. This group might function for a while, only to break up when things got hot. The players ask for advice or a sign of approval from the aloof nobles Neon, Argon, or Xenon. They could use the "family resemblances" embedded in the periodic table to predict the behavior of a newly met elemental character, or to guide them in their search for a character needed to solve a problem posed within their chemical adventure.

This sketch of a chemical adventure points to the ways that a computer game program could draw upon the same aspects that make current adventures entertaining, yet teach an abstract knowledge domain. This game is a "quest" adventure - a coordinated electorial campaign or war game or detective story could just as well be designed within this world as alternate ways to teach the properties of chemical elements and the periodic law.

Evolutionary Mystery. Detective stories are a popular form of entertainment. Suppose we embed a "who-done-it" in a world where the suspects are the different members of the animal world. The player is the detective, and gathers clues about the criminal. The information might be provided at the scene of the crime, or gathered through interviews with the suspect animals. Clues may involve the size of the animal, its color, type of skin, footprint, food consumed, kind of terrain where the crime took place, time of the day when it occurred, etc. The player will have to narrow down the list of suspects given all that information, or even gather these kind of data him/herself. The suspects may be initially chosen to represent widely different animals, but as the player gains experience they can be picked from among increasingly similar groups. The player would quickly learn common and distinguishing characteristics of different animals. With appropriate directions players could also learn the hierarchical nature of the biological classification system and the different categories in it.

Multi-function Computer Activities. It is possible to design computer activities that are entertaining, yet at the same time teach the players skills and knowledge in several different domains. As an example, we created a computer program called HiSeas, in which players sail a ship across the ocean,trying to reach a destination port. At the top level, this game exercised spatial reasoning, as the players had to keep track of direction and distance to avoid getting lost. Along the way players encountered dangers (shark attacks, rogue waves, typhoons) and had to exercise problem solving skills to overcome them. Within the shark attack episode, for example, the player had to apply numerical estimation skills, determining where to throw a harpoon to hit the shark. Finally, the whole sequence of the player's "adventure" was stored by the computer. Players later reran their stories, and some practiced their writing skills by modifying their story.

Starting with this HiSeas game, it was easy to create different educational/entertainment worlds as variants on it. We simplified the language and the world (with the help of Peg Griffin) but retained the same task to develop a version for fourth graders called LoSeas. We completely changed the world (with the help of Warren Simmons) so that the task was to get from 80th St. to 112th St. in New York (avoiding street gangs, 5 alarm fires, and packs of wild dogs), in a version called MeanSteets. We also sketched out a variant of the game set in space, which introduces interesting new spatial reasoning problems. Obviously there are other worlds within which practice of the same skills could be embedded.

There are large individual differences in preferences for games and other computer related activities (see Malone, 1980; Papert et al., 1979; Watt, 1979). Our tour through the different worlds derived from the original HiSeas game has been designed to show how it is possible to develop a family of games cast in sufficiently different environments to appeal to a wide audience. While an inner city child may find it irrelevant to navigate a boat across the ocean, s/he may be intrigued when challenged to travel to updown Manhattan (and probably perform the task quite well). At the same time all these games provide practice with a common set of basic skills.

So far we have discussed the question of how the appeal of educationally relevant computer-based learning activities may be increased by presenting the same kind of activity within different worlds. To reach even wider audiences, and to provide variety to all, one might think of activities other than those represented by the action or adventure genres of games. Only some of the people enjoy war novels or science fiction books; many others enjoy other literary genres. We suspect that similar differences may exist in preferences for computer games, and efforts to design completely different games may pay handsome dividends.

Beyond Games: Creative Computer Learning Environments

Computer activities differ in how much freedom they give the user. The game programs described so far, while high on interactive value, define a rigid set of rules within which the users operate. There are other computer activities which can be viewed as providing the users with some tools, then allowing them to use these tools in whatever way they please. LOGO and Smalltalk, described before, were expressly designed with the "tool" philosophy in mind. In this section we describe a number of programs and activities which also serve as tools. These programs are narrower in scope than those provided by the programming languages of LOGO and Smalltalk, but on the other hand they provide the user with much more initial support. We label these activities "creative" since users can apply the tools to create a variety of unique stories, drawings, or music.

Writing. One of the most significant uses of computers in the near future will be for word processing. More and more of the people who make their living crafting words are replacing their typewriters with personal computers. The development of hardware and software to serve these people also makes it plausible to use personal computers to teach writing. In this section we will present four kinds of computer writing activities.

(i) Story Makers. Part of the reason novice writers have difficulty in organizing and expressing their ideas is that they have to deal simultaneously with problems at many levels. When writing with paper and

pencil, they have to draw each individual letter, they have to select and correctly spell words, they have to organize grammatical phrases and sentences, they have to maintain intersentential connectivity, and at the same time they have to follow an overall plan to ensure that the text they are creating will properly convey their ideas. Faced with such problems novice writers often find it easiest to "downslide" their effort, concentrating on the lower-rather than the higher-level aspects of writing (Collins & Gentner, 1979). There are some programs which have been developed to help novices exercise the higher level skills involved in writing, while providing support for handling the lower level requirements. The StoryMaker programs (Rubin & Gentner, 1979) are an example, presenting the user with parts of a story, and offer a number of possible continuations at choice points. The story reflects the user's choices, and the new choices presented to the user depend on previous ones. The program contains a tree of possible stories, and the resulting story is a path through that tree. This kind of program allows even young children to generate interesting stories. At each branch point, the children see the alternatives printed on the computer screen and type in the number of the choice they want. Rather than getting hung up in the lower level mechanics of printing, spelling, punctuation, syntax, or semantics, they engage in high level decisions about the flow of the narrative.

We have found that fourth grade children enjoy creating stories with these kinds of StoryMaker programs, and that the impact carries over even to their writing with pencil and paper. In our classroom study we also found that some of the children were eager and able to go beyond the use of the original branches provided. They created their own story trees, which they then entered into the computer. In doing that they combined a high level creative activity with elementary programming (the modification of an existing program).

(ii) Writing Adventure Worlds. The story trees of StoryMaker programs are an excellent way to introduce the practice of high level skills, yet any given story tree is soon exhausted by its users. The HiSeas program, mentioned earlier, was created to deal with this problem. Inspired by earlier "adventure" games, we created a moderately rich world in which the player is faced with problems to solve. The computer automatically saves in long term storage the text (and action) that the player experiences during the game as a "narrative" of that person's adventure on HiSeas. The action of the story, including graphics, sound effects, and animation, is part of the narrative since the story is saved as a program which recreates the text and action when run. Students can later edit the narrative if they want. Editing can be accomplished on a number of levels: words or phrases can be modified to improve the story, or whole units may be deleted or added. The higher level modifications are made easy since the hierarchical structure of the story is preserved as a structured Pascal program. In this way children participate in an intrinsically motivating writing activity (they rewrite the story of their own adventure), but can practice the different writing skills in the highly supportive environment provided by the computer editor.

(iii) A Writer's Assistant. In another attempt to provide a supportive environment for exercising writing skills, we have been developing a Writer's Assistant, a computer program designed to help people who compose and edit their own text on a computer (Quinsaat, Levin, & Gentner, 1980). As a base, we have started with an existing computer text editor available on Apple II computers. A good text editor program is by itself very helpful for creating and changing text. However, we have been augmenting this editor (the UCSD Pascal Editor) to help users at several different points in the writing process: At one end, the Writer's Assistant can help with spelling. If the writer is uncertain about the spelling of a word s/he can generate a hypothesis about how the word is spelled and ask the Writer's Assistant to verify it.

Another aid provided by the Writer's Assistant is the ability to display the text the writer has entered in a number of different presentation formats, which can help the writer evaluate the overall organization of the text. Finally, the Writer's Assistant can aid in generating possible combinations of words or ideas, to be evaluated and, if appropriate, used by the writer.

(iv) Electronic Pen Pals. Electronic mail will soon be a major communication medium, especially in the commercial world. We have been conducting research into ways that it can be used for education (Black, Levin, Mehan, & Quinn, 1980; Quinn, Mehan, Levin, & Black, 1980). At an informal level, students can send messages to other students, even when widely separated in space or time. For example, even when not sharing the same computer or communication net students could send information on disks to each other. These messages can go beyond the ordinary "I'm fine. How are you?" to include the exchange of graphics, music, animation, and programs. Again, just by having a flexible medium for the creation, modification, and transfer of varied forms of information we expect a large degree of interest and creative uses on the part of the students.

Graphic Worlds. Another domain in which computers can support learning is by creating static or dynamic visual images. Below we describe four kinds of graphic tools.

(i) Color Sketch. The simplest graphic programs serve as a sort of "etch-a-sketch", allowing the user to move a "brush" by controlling its horizontal and vertical position. In our computer club we found some children who quickly mastered such a program on Apple II computers, called Color Sketch, to create quite striking pictures. However, as with all uses of computers, one should ask how using a computer in this way compares to more standard means, such as paints or crayons. An analysis of Color Sketch is instructive in this respect. While the program limits the user to a 24x40 grid, it already offers a number of intriguing features which make it a favorite with children. For one thing, it is possible to erase or redraw any part of the screen at any time. It is also possible to change all occurrences of any color (including the color of the whole background) to any other color with a single command. Finally, pictures can be stored on the disk, then brought back for further work at a later time. The ability to store a drawing at any stage also comes in handy when users want to create and compare slightly different versions of the same drawing. Thus, while limited in many ways, even the simple Color Sketch program offers much flexibility in drawing.

(ii) PaintPot. Some existing programs for drawing go far beyond the capabilities of Color Sketch. The PaintPot program (Kay, 1977) is one of them. In PaintPot the user is provided with brushes and a palate of colors drawn on the screen. One striking property of this particular graphics environment is that users can draw brushes and paints for further use. For example, it is possible to draw a pattern such as a checkerboard, then select it as the current paint and continue painting with a checkerboard paint (previously the privilege of cartoon characters only). This is an example of the ways in which computer artistic worlds start to provide capabilities that go beyond the standard artistic media.

(iii) Turtle Graphics. The conventional way to define figures for computer graphics is to specify points and lines in terms of X and Y coordinates. This notion of Cartesian coordinates is difficult to master, especially for younger children. To make the task of drawing lines more concrete, the LOGO project (Papert, 1971) developed an alternate "world" for creating graphics, called Turtle Graphics. In this metaphorical graphics world, the user controls the motion of a creature called "the turtle". The turtle has a pen, which is either up or down. When the turtle moves with its pen in the down position a line is drawn. The turtle can be instructed to move forward a certain distance and to turn a certain number of degrees. The control of the turtle is not only a way of learning programming and geometry; it is at the same time a way of creating fascinating artistic designs.

The turtle geometry system renders the creation of certain kinds of figures much easier than other systems, and raises the issue of the relation between the properties of a given graphics world and the kind of drawings produced within it. To explore the impact of different graphics worlds on sketching, we created two simple sketching programs for Apple II computers. Both make use of the Turtle Graphics notion of an entity which the user orders about, and which can leave a line behind when moving. In Movetocar, the entity may be viewed as a "car" which is driven by turning two controls. One control determines the X (horizontal) position of the car; the other the Y (vertical) position. This graphic world feels very much like an "etch-a-sketch" world: drawing horizontal and vertical lines is trivial, but diagonals and curves are difficult to produce. The second graphics world, called Bouncecar, is more interesting. In this world, one control determines the acceleration of the "car"; the other turns the car left or right from its current heading. In this world curves are easy to generate, while horizontal and vertical lines are suddenly much harder to draw. The name Bouncecar comes from a feature we soon had to add to the original program. With that program, a user often drove the car off the edge of the screen and then never found it again. So we modified the program so that the car bounces off the edge of the screen. An unexpected side effect of this change was that "scallops" became very easy to draw, as the car bounces off the edge and then keeps turning back toward it, only to bounce again and again.

One general point made by these two simple sketching programs is that each graphics world has its strengths and weaknesses; each allows some things to be done with ease while others are difficult to achieve. It is important to allow learners to experiment with many different kinds of worlds, so that they can choose the environment best suited for a particular task, then shift worlds to accomplish some others.

(iv) Dynamic Graphic Worlds. The most powerful way in which computers can surpass more traditional visual artistic media is in the area of dynamic graphic productions. Cartoons and other dynamic graphics were produced in the past only by the laborious sketching of thousands of still frames. Computers allow motion to be specified directly by the user, with the computer producing the implied intermediate still frames. For example, Papert and the Logo group at MIT (Papert et al., 1979) have recently extended the Turtle Graphics world to include a "dynamic turtle". Lines created by that turtle while moving in some frame of reference continue to move after the turtle has passed on to draw new moving lines.

Music Worlds. Another domain in which computers can support learning is music. Traditionally, as in writing, learners have had to master laboriously lower level skills (playing an instrument, writing notes) before they could even experience, much less master, higher levels such as composing music. However recent advances in both the hardware and the software available for personal computers allow even musical novices to experiment with musical composition. The resulting musical piece can be played back immediately, but also stored for further "editing" and later replay. As in the graphic domain, there are many possible "music worlds" that can be provided by personal computers. Each of these worlds is particularly suited for certain musical tasks but not for others. Not only can these worlds provide support for musical education, but they will challenge and then help redefine the basic notions of musical composition, performance, and enjoyment, by turning music into an interactive experience for the vast majority that now are just passive audience.

There are now several "music editor" systems, which allow the user to enter notes (which appear graphically on a music staff drawn on the computer screen). These systems make it easy to enter new music, hear it performed by the computer, change it, and store it for later enjoyment or further modification. This general ability to easily display, modify, and save music, graphics, and text allows people to become involved with the higher, more creative levels of composition even while they are mastering the lower, more mechanical levels.

We have presented as broad a selection of types of computer activities for learning as we could generate, mainly to make the point that the use of computers for learning can go far beyond the traditional domains of mathematics and programming. In the next section, we will try to outline a set of design techniques that we have distilled for creating effective learning environments by using personal computers.

Designing Educational Computer Activities

A personal computer is a powerful tool available for building learning environments. Like other tools, the placement of the computer and the goals to be achieved are selected by the designer. Whatever the educational goals of the designer, our observations indicate that there exist a number of design techniques which increase the likelihood that a given computer activity will effectively achieve its goals. Broadly speaking these techniques can be classified as falling into two categories: some of them provide the users with the right amount of support, while others sustain the user's motivation. The division is not clear cut, though, and some of the suggested methods apply to both classes.

Methods for Achieving Supportive Environments

Dynamic Computer Support. Computer activities should be designed so that users have as much support as necessary (especially at the initial stages), but also have the ability to go beyond that support. One way to achieve dynamic support is to have levels of difficulty within the activity. For example, one can let a car go faster, have a more winding course, or introduce new obstacles. Level of difficulty can be manipulated not only by quantitatively changing the values along existing dimensions, but also by introducing new dimensions. In a car race activity, for example, one can have more expert drivers monitor a gas gauge as well as their position on the course. An adult expert can play an important role in providing dynamic support, both by suggesting new goals within an activity and by suggesting (or simply making available) new activities.

Social Support. Activities should be designed so as to enable and encourage peer interaction and help. Fostering interaction and help is a desirable goal in itself; it is a good way to prepare students to live within a society, to learn how to contribute to other members of it, and to know how to seek help when necessary. In addition, helping another student is one of the best ways for the helper to gain a better understanding of the activity in question. Finally, peer support is likely to ease the burden placed on the instructor. The simplest way to achieve peer support is to let it take place, since it is natural for most people to engage in such kind of behavior. It is also possible to encourage such behavior by designing activities which require cooperation between a number of users, suggesting projects which can be broken down into subparts, or encouraging group discussions of activities and projects.

Progression to Expertise. Educational computer activities should be designed so that they enable the students gain in expertise as they further engage in those activities. One simple way of achieving this goal is to have an adjustable difficulty level (as described in the section of dynamic computer support). Another way is to prepare and provide for activities likely to develop as a result of using the original one. The modification of an existing program is one such new activity. Writing new programs inspired by existing ones is another. Related activities need not be confined to the computer, as demonstrated by the ROADRACE example. The important point is for the instructor to be aware of the need for vertical variety and be on the lookout for opportunities for its application. Within simulated worlds and situation it is possible to demand increased expertise by adding additional realistic features to the simplified situation presented at the beginning. Finally, as students become proficient with some activity the designer may consider the possibility of presenting the same activity under a different representation. Multiple and parallel representations are one of the hallmarks of experts in almost any field. By providing students with different approaches to the same problem we can increase their own expertise.

Methods for Achieving Motivating Environments

Active/Interactive Activities. Activities should be designed to ensure an active role for the user. An active role is relatively easy to achieve when the program calls for much interaction between the user and the computer. It is preferable that the user have the feeling that his/her acts determine what the computer does rather than that the computer determine what s/he should do next. This technique applies not only within a given activity or program, but also across them: users should be allowed to freely switch between a number of different activities. Malone's (1980) three motivating factors -- fantasy, challenge, and curiosity -- can all be viewed as different ways for ensuring active participation on the user's part.

Breadth of Activities. Individuals greatly differ in interests, backgrounds, and capabilities. As a result they also differ in the kind of educational activities which attract them and from which they benefit. To accommodate these differences students have available a large number of varied activities. This variety is also necessary so that the individual user can shift between activities as s/he choses (the Active Principle). One special aspect of Breadth of Activity has to do with the preparation of isomorphic activities. These are activities which call upon the same mental faculties for their performance, but are disguised under different cover stories. The HiSeas, LoSeas, and MeanStreets games are an example of such isomorphs. Isomorphic activities are particularly useful when the educational designer feels that learners will benefit from practicing certain skills, and wants to increase the chances that each learner will find at least one relevant activity interesting. An important consideration in developing program isomorphs is to make the cover story meaningful to the intended audience (e.g., make the activity take place in an interesting environment). Isomorphic activities are also useful for helping users generalize (and transfer) the skills they have acquired in the original environment by applying it in the different situation of the isomorph.

An important kind of variety is provided by "practical" programs. These are programs which serve as tools for obtaining other goals. Text editing or "calculator" programs fall into this category. Programs used for creative activities (drawing, animation, music) are also included here. Students may be encouraged to develop their own tools, useful for their own purposes.

Powerful Environments. One way to sustain the motivation of novices is to have their actions produce easily discernable changes in the behavior of the program. Action games are a prime example of activities where this technique is often applied. The technique also applies for novice programmers, who should be directed, if possible, to write simple programs which have spectacular effects. A five-line program which fills the screen with interesting patterns is much preferable as an introductory exercise to a two-hundred-line program which prints a single number at the end. The LOGO language is a good example of an environment where the programmer can get a lot of "bang per buck." "Tool" or creative programs also often provide novices with the power to produce spectacular effects without having to toil to achieve them.

These design methods have focused on the relatively broad issues of the overall structure of computer learning activities and on the interactions of participants in those activities with computers and with other people in the learning environment. There are other levels in this design process that are also important. (For example, the Minnesota Educational Computing Consortium presents a number of presentation techniques such as screen layout for instructional software (MECC, 1980).) The global design methods described here are not meant as prescriptions to be slavishly followed, only as techniques found to be useful across a wide range of educational computer activities. As is the case in many other fields, experience and field testing will turn out to be the best guides for the design of fruitful educational activities.

THE CHALLENGE TO SCHOOLS

Let us now consider possible ways for schools to meet the challenge of of personal computers. The easiest approach would be to ignore the implications. However this "head in the sand policy" has its dangers, since the important learning might shift to other settings, leaving schools an anachronistic institution without any real function, a sad and transient existence at best. This is the threat of personal computers to schools.

The challenge has an positive side as well, providing exciting new opportunities for schools. One of the surprises of our three studies of personal computers in different environments was how well computers could be integrated into a school classroom. The elements of this successful use of computers in a classroom derive directly from the design methods described previously - the gradual transition from the use of pre-programmed activities to program modification to programming, the use of social resources, the provision of a broad variety of computer activities for learners to select among. If schools capitalize on the opportunities provided they can become new and envigorated learning centers, taking advantage of their ability to provide effective face-to-face support from peers and expert adults. Schools can also provide more cost-effective systems, especially when it comes down to the use of specialized and expensive peripherals. So schools are not necessarily made obsolete by the impending advances in computer technology. By adapting appropriately they can even gain from it, becoming environments where effective and rewarding educational processes take place.

How then can you, the reader, proceed in the face of this challenge? Should you immediately sign up for a FORTRAN programming course? Probably not. Instead we suggest that you use the same educational design methods described previously to proceed with your own plan for learning.

<u>Building your own dynamic support environment</u>. You can actively explore the usefulness of personal computers by trying one out. As they become more common, temporary access will become easier. Use social resources - find a friend who has access to and knowledge of personal computers, and learn from

this "local expert". Search out teachers who already use computers for education, and learn how they use them. When you get access to a personal computer, start your exploration with preprogrammed activities that achieve goals you find important. Start by using existing programs, and move gradually toward program modification (perhaps with the assistance of a friend who is a more experienced programmer), then move toward program design and programming.

Building your own motivating environment. Select activities that interest you personally and professionally. If you do not find computer games appealing, try out other computer activities, such as graphics, animation, music, text editing, or record keeping. Take an active role in searching out these activities, visiting your local computer stores, reading relevant computer periodicals, talking with friends and associates.

These suggestions are all meant as illustrations of how you can use for yourself the design methods described previously, as well another example of what each method involves. Education is currently in a state of transition. Rather than ignoring the threats implicit in the challenge to schools, we urge you to take an active role in meeting it. Then you will be able to help shape the course of future developments so that personal computers can be used to achieve a more effective, equitable, and humane educational process.

References

Abelson, H., diSessa, A., & Rudolph, L. Velocity space and the geometry of planetary orbits. Cambridge, MA: MIT Artificial Intelligence Memo No. 320, December 1974.

Atkinson, R. C., & Wilson, H. A. Computer-assisted instruction. In R. C.

Atkinson & H. A. Wilson (Eds.), Computer-assisted instruction: A book of readings. New York: Academic Press, 1969.

Bamberger, J. Developing a musical ear: A new experiment. Cambridge, MA: MIT Artificial Intelligence Memo No. 264, July 1972.

Bitzer, D., & Skaperdas, D. The economics of a large-scale computer-based education system: Plato IV. In W. H. Holtzman (Ed.), Computer- assisted instruction, testing and guidance. Nw York: Harper & Row, 1970.

Black, S., Levin, J. A., Mehan, H. B., & Quinn, C. N. Real and non-real time interaction: Unraveling multiple threads of discourse. La Jolla, CA: Laboratory of Comparative Human Cognition, 1980.

Brown, A. L., & DeLoache, J. S. Skills, plans, and self-regulation. In R. Siegler (Ed.), Children's thinking: What develops. Hillsdale, NJ: Erlbaum, 1978.

Brown, J. S., & Burton, R. R. Multiple representations of knowledge for tutorial reasoning. In D. G. Bobrow & A. Collins (Eds.), Representation and understanding: Studies in cognitive science. New York: Academic Press, 1975.

Brown, J. S., Burton, R. R., & Zdybel, F. A model-driven question answering system for mixed-initiative computer assisted instruction. IEEE Transactions on Systems, Man and Cybernetics, 1973, 3, 248-257.

Carbonell, J. R. AI in CAI: An artificial intelligence approach to computer-assisted instruction. IEEE Transactions on Man-Machine Systems, 1970, 11, 190-202.

Collins, A. Processes in acquiring knowledge. In R. C. Anderson, R. J. Spiro, & G. Montague (Eds.), Schooling and the acquisition of knowledge. Hillsdale, NJ: Erlbaum, 1976.

Collins, A., & Gentner, D. A framework for a cognitive theory of writing. In L. W. Gregg & E. Steinberg (Eds.), Cognitive processes in writing: An interdisciplinary approach. Hillsdale, NJ: Erlbaum, 1979.

Flanigan, J. Apple Computer looks tasty but its market needs ripening. Los Angeles Times, 19 October 1980.

Goldstein, I. The computer as coach: An athletic paradign for intellectual education. Cambridge, MA: MIT Artificial Intelligence Memo No. 339, February 1977.

Kohlberg, L., & Mayer, R. Development as the aim of education. Harvard Educational Review, 1972, 42, 449-496.

Laboratory of Comparative Human Cognition. Cross-cultural psychology's challenges to our ideas of children and development. American Psychologist, 1979, 34, 827-833.

Learning Research Group. Personal dynamic media. Palo Alto, CA: Xerox Palo Alto Research Center, SSL 76-1, 1976.

Levin, J. A. Interpersonalized media: What's news? Byte, 1980, 5, 214-228.

Levin, J. A., & Kareev, Y. Problem solving in everyday situations. The Quarterly Newsletter of the Laboratory of Comparative Human Cognition, 1980, 2, 45-51.

Malone, T. What makes things fun to learn? A study of intrinsically motivating computer games. Palo Alto, CA: Xerox Palo Alto Research Center, 1980.

Minnesota Educational Computing Consortium. A guide to developing instructional software for the Apple II microcomputer. St. Paul, MN: MECC, 1980.

Papert, S. Teaching children to be mathematicians vs. teaching about mathematics. Cambridge, MA: MIT Artificial Intelligence Memo No. 249, July 1971.

Papert, S., Watt, D., diSessa, A., & Weir, S. Final report of the Brookline LOGO Project. Part II: Project summary and data analysis. Cambridge, MA: Artificial Intelligence Laboratory, LOGO Memo No. 53, September 1979.

Quinn, C. N., Mehan, H. B., Levin, J. A., & Black, S. Real education in non-real time: Cross-media analysis of instructional interaction. La Jolla, CA: Laboratory of Comparative Human Cognition, 1980.

Quinsaat, M. G. But it's important data: Making the demands of a cognitive experiment meet with the educational imperatives of the classroom. _The Quarterly Newsletter of the Laboratory of Comparative Human Cognition_, 1980, _2_.

Quinsaat, M. G., Levin, J. A., & Gentner, D. A Writer's Assistant: The use of personal computers to teach writing. La Jolla, CA: University of California, San Diego, 1980.

Rubin, A. Making stories, making sense. _Language Arts_, 1980, 285-298.

Rubin, A., & Gentner, D. An educational technique to encourage practice with high-level aspects of texts. Cambridge, MA: Bolt Beranek and Newman, Inc., 1979.

Stevens, A. L., & Collins, A. The goal structure of a socratic tutor. In _Proceedings of the Association for Computing Machinery National Conference_. Seattle, WA: 1977.

Suppes, P. The uses of computers in education. _Scientific American_, 1966, _215_, 206-221.

Vygotsky, L. S. _Mind in society_. M. Cole, V. John-Steiner, S. Scribner, & E. Souberman (Eds.), Cambridge, MA: Harvard University Press, 1978.

Walling, V. C., Thomas, T. C., & Larson, M. A. Educational implications of in-home electronic technology. (Research memorandum 33). Menlo Park: SRI International, 1979.

Watt, D. Final report of the Brookline LOGO Project. Part III: Profiles of individual student's work. Cambridge, MA: Artificial Intelligence Laboratory, LOGO Memo No. 54, September 1979.

CHIP Technical Report List

1. David M. Green and William J. McGill. On the equivalence of detection probabilities and well known statistical quantities. October, 1969.
2. Donald A. Norman. Comments on the information structure of memory. October, 1969.
3. Norman H. Anderson. Functional measurement and psychophysical judgment. October, 1969.
4. James C. Shanteau. An additive decision-making model for sequential estimation and inference judgments. October, 1969.
5. Norman H. Anderson. Averaging model applied to the size-weight illusion. October, 1969.
6. Norman H. Anderson and James C. Shanteau. Information integration in risky decision making. November, 1969.
7. George Mandler, Richard H. Meltzer, Zena Pearlstone. The structure of recognition. Effects of list tags and of acoustic and semantic confusion. November, 1969.
8. Dominic W. Massaro. Perceptual processes and forgetting in memory tasks. January, 1970.
9. Daniel Graboi. Searching for targets: The effects of specific practice. February, 1970.
10. James H. Patterson and David M. Green. Discrimination of transient signals having identical energy spectra. February, 1970.
11. Donald A. Norman. Remembrance of things past. June, 1970.
12. Norman H. Anderson. Integration theory and attitude change. August, 1970.
13. A.D. Baddeley and J.R. Ecob. Reaction time and short-term memory: A trace strength alternative to the high-speed exhaustive scanning hypothesis. November, 1970.
14. A.D. Baddeley. Retrieval rules and semantic coding in short-term memory. December, 1970.
15. Roy D. Patterson. Residue pitch as a function of the number and relative phase of the component sinusoids. March, 1971.
16. George Mandler and Marilyn A. Borges. Effects of list differentiation, category membership and prior recall on recognition. May, 1971.
17. David E. Rumelhart, Peter H. Lindsay, and Donald A. Norman. A process model for long-term memory. May, 1971.
18. David E. Rumelhart and Adele A. Abrahamson. Toward a theory of analogical reasoning. July, 1971.
19. Martin F. Kaplan. How response dispositions integrate with stimulus information. August, 1971.
20. Martin F. Kaplan and Norman H. Anderson. Comparison of information integration and reinforcement models for interpersonal attraction. August, 1971.
21. David M. Green and R. Duncan Luce. Speed-accuracy tradeoff in auditory detection. September, 1971.
22. David E. Rumelhart. A multicomponent theory of confusion among briefly exposed alphabetic characters. November, 1971.
23. Norman H. Anderson and Arthur J. Farkas. New light on order effects in attitude change. March, 1972.
24. Norman H. Anderson. Information integration theory: A brief survey. April, 1972.
25. Donald A. Norman. Memory, knowledge, and the answering of questions. May, 1972.
26. David J. Weiss. Averaging: An empirical validity criterion for magnitude estimation.
    Norman H. Anderson. Cross-task validation of functional measurement. June, 1972.
27. David E. Rumelhart and Patricia Siple. The process of recognizing tachistoscopically presented words. August, 1972.
28. Ebbe B. Ebbesen and Richard Bowers. The effects of proportion of risky to conservative arguments in a group discussion on risky shift. September, 1972.
29. Ebbe B. Ebbesen and Michael Haney. Flirting with death: Variables affecting risk taking on our nation's highways. September, 1972.
30. Norman H. Anderson. Algebraic models in perception. November, 1972.
31. Norman H. Anderson. Cognitive algebra: Information integration applied to social attribution. December, 1972.
32. Jean M. Mandler and Nancy L. Stein. Recall recognition of pictures by children as a function of organization and of distractor similarity. January, 1973.
33. David E. Rumelhart and Donald A. Norman. Active semantic networks as a model of human memory.
    Marc Eisenstadt and Yaakov Kareev. Towards a model of human game playing. June, 1973.
34. George Mandler. Memory storage and retrieval: Some limits on the reach of attention and consciousness. July, 1973.
35. Kent L. Norman. A method of maximum likelihood estimation for stimulus integration theory. August, 1973.
36. Yaakov Kareev. A model of human game playing. August, 1973.
37. Donald A. Norman. Cognitive organization and learning. August, 1973.
38. The Center for Human Information Processing: A Five Year Report — 1968-73.

39. Larry D. Rosen and J. Edward Russo. Binary processing in multi-alternative choice. October, 1973.

40. Samuel Himmelfarb and Norman H. Anderson. Integration theory analysis of opinion attribution. December, 1973.

41. George Mandler. Consciousness: Respectable, useful, and probably necessary. March, 1974.

42. Norman H. Anderson. The problem of change-of-meaning. June, 1974.

43. Norman H. Anderson. Methods for studying information integration. June, 1974.

44. Norman H. Anderson. Basic experiments in person perception. June, 1974.

45. Norman H. Anderson. Algebraic models for information integration. June, 1974.

46. Ebbe B. Ebbesen and Vladimir J. Konečni. Cognitive algebra in legal decision making. September, 1974.

47. Norman H. Anderson. Equity judgments as information integration.
Arthur J. Farkas and Norman H. Anderson. Input summation and equity summation in multi-cue equity judgments. December, 1974.

48. George Mandler and Arthur Graesser II. Dimensional analysis and the locus of organization. January, 1975.

49. James L. McClelland. Preliminary letter identification in the perception of words and nonwords. April, 1975.

50. Donald A. Norman and Daniel G. Bobrow. On the role of active memory processes in perception and cognition. May, 1975.

51. J. Edward Russo. The value of unit price information. An information processing analysis of point-of-purchase decisions. June, 1975.

52. Elissa L. Newport. Motherese: The speech of mothers to young children. August, 1975.

53. Norman H. Anderson and Cheryl C. Graesser. An information integration analysis of attitude change in group discussion. September, 1975.

54. Lynn A. Cooper. Demonstration of a mental analog of an external rotation.
Lynn A. Cooper and Peter Podgorny. Mental transformations and visual comparison processes: Effects of complexity and similarity. October, 1975.

55. David E. Rumelhart and Andrew Ortony. The representation of knowledge in memory. January, 1976.

56. David E. Rumelhart. Toward an interactive model of reading. March, 1976.

57. Jean M. Mandler, Nancy S. Johnson, and Marsha DeForest. A structural analysis of stories and their recall: From "Once upon a time" to "Happily ever after". March, 1976.

58. David E. Rumelhart. Understanding and summarizing brief stories. April, 1976.

59. Lynn A. Cooper and Roger N. Shepard. Transformations on representations of objects in space. April, 1976.

60. George Mandler. Some attempts to study the rotation and reversal of integrated motor patterns. May, 1976.

61. Norman H. Anderson. Problems in using analysis of variance in balance theory. June, 1976.

62. Norman H. Anderson. Social perception and cognition. July, 1976.

63. David E. Rumelhart and Donald A. Norman. Accretion, tuning and restructuring: Three modes of learning. August, 1976.

64. George Mandler. Memory research reconsidered: A critical view of traditional methods and distinctions. September, 1976.

65. Norman H. Anderson and Michael D. Klitzner. Measurement of motivation.
Michael D. Klitzner and Norman H. Anderson. Motivation x expectancy x value: A functional measurement approach. November, 1976.

66. Vladimir J. Konečni. Some social, emotional, and cognitive determinants of aesthetic preference for melodies in complexity. December, 1976.

67. Hugh Mehan, Courtney B. Cazden, LaDonna Coles, Sue Fisher, Nick Maroules. The social organization of classroom lessons. December, 1976.

67a. Hugh Mehan, Courtney B. Cazden, LaDonna Coles, Sue Fisher, Nick Maroules. Appendices to the social organization of classroom lessons. December, 1976.

68. Norman H. Anderson. Integration theory applied to cognitive responses and attitudes. December, 1976.

69. Norman H. Anderson and Diane O. Cuneo. The height + width rule in children's judgments of quantity. June, 1977.
Norman H. Anderson and Clifford H. Butzin. Children's judgments of equity. June, 1977.

70. Donald R. Gentner and Donald A. Norman. The FLOW tutor: Schemas for tutoring. June, 1977.

71. George Mandler. Organization and repetition: An extension of organizational principles with special reference to rote learning. May, 1977.

72. Manuel Leon. Coordination of intent and consequence information in children's moral judgements. August, 1977.

73. Ted Supalla and Elissa L. Newport. How many seats in a chair? The derivation of nouns and verbs in American Sign Language. November, 1977.

74. Donald A. Norman and Daniel G. Bobrow. Descriptions: A basis for memory acquisition and retrieval. November, 1977.

75. Michael D. Williams. The process of retrieval from very long term memory. September, 1978.

76. Jean M. Mandler. Categorical and schematic organization in memory. October, 1978.

77. James L. McClelland. On time relations of mental processes: A framework for analyzing processes in cascade. October, 1978.

78. Jean M. Mandler and Marsha DeForest. Developmental invariance in story recall. November, 1978.
Jean M. Mandler, Sylvia Scribner, Michael Cole, and Marsha DeForest. Cross-cultural invariance in story recall. November, 1978.

79. David E. Rumelhart. Schemata: The building blocks of cognition. December, 1978.

80. Nancy S. Johnson and Jean M. Mandler. A tale of two structures: Underlying and surface forms in stories. January, 1979.

81. David E. Rumelhart. Analogical processes and procedural representations. February, 1979.

82. Ross A. Bott. A study of complex learning: Theory and methodologies. March, 1979.

83. Laboratory of Comparative Human Cognition. Toward a unified approach to problems of culture and cognition. May, 1979.

84. George Mandler and Lawrence W. Barsalou. Steady state memory: What does the one-shot experiment assess? May, 1979.

85. Norman H. Anderson. Introduction to cognitive algebra. June, 1979.

86. Edited by Michael Cole, Edwin Hutchins, James Levin and Naomi Miyake. Naturalistic problem solving and microcomputers. Report of a Conference. June, 1979.

87. Donald A. Norman. Twelve issues for cognitive science. October, 1979.

88. Donald A. Norman. Slips of the mind and an outline for a theory of action. November, 1979.

89. The Center for Human Information Processing: A Description and a Five-Year Report (1974-1979). November, 1979.

90. Michael Cole and Peg Griffin. Cultural amplifiers reconsidered. December, 1979.

91. James L. McClelland and David E. Rumelhart. An interactive activation model of the effect of context in perception. Part I. April, 1980.

92. James L. McClelland and J.K. O'Regan. The role of expectations in the use of peripheral visual information in reading. February, 1980.

93. Edwin Hutchins. Conceptual structures of Caroline Island navigation. May, 1980.

94. Friedrich Wilkening and Norman H. Anderson. Comparison of two rule assessment methodologies for studying cognitive development. June, 1980.

95. David E. Rumelhart and James L. McClelland. An interactive activation model of the effect of context in perception. Part II. August, 1980.

96. Jean M. Mandler. Structural invariants in development. September, 1980.

97. David E. Rumelhart and Donald A. Norman. Analogical processes in learning. October, 1980.

98. James A. Levin and Yaakov Kareev. Personal computers and education: The challenge to schools. November, 1980.

Note: Requests for CHIP reports should be addressed to the author. Reports are also available through the Library Loan Service of the University of California, San Diego, La Jolla, California 92093.