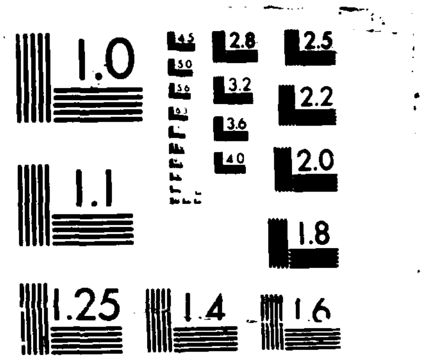
AD-A180 231   CREATIVITY IN EDUCATION: A STANDARD FOR COMPUTER-BASED   1/1
TEACHING(U) YALE UNIV NEW HAVEN CT DEPT OF COMPUTER
SCIENCE R C SCHANK ET AL FEB 87 YALE/CSD/RR-518
UNCLASSIFIED   N00014-82-K-0149   F/G 5/6   NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963 A

Creativity in Education:
A Standard for Computer-Based Teaching

Roger C. Schank and Robert Farrell

YALEU/CSD/RR #518

February 1987

YALE UNIVERSITY
DEPARTMENT OF COMPUTER SCIENCE

Creativity in Education:

A Standard for Computer-Based Teaching

Roger Schank and Robert Farrell

Yale Artificial Intelligence Project
Yale Dept. of Computer Science
Box 2158 Yale Station
New Haven, CT 06520

This work was supported in part by the Advanced Research Projects Agency of the Department of Defense and monitored under the Office of Naval Research under contract N00014-82-K-0149 and the Air Force contract AFOSR 85-0343.

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
#518

2. GOVT ACCESSION NO.
AD-A180 231

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)
Creativity in Education: A Standard for Computer-Based Teaching

5. TYPE OF REPORT & PERIOD COVERED
Research Report

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)
Roger C. Schank and Robert Farrell

8. CONTRACT OR GRANT NUMBER(s)
N00014-82-K-0149

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Yale University
Computer Science Department
10 Hillhouse Avenue - New Haven, CT 06520

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS
Advanced Research Projects Agency
1400 Wilson Boulevard
Arlington, VA 22209

12. REPORT DATE
February 1987

13. NUMBER OF PAGES
26

14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)
Office of Naval Research
Information Systems Program
Arlington, VA 22217

15. SECURITY CLASS. (of this report)
Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

learning
creativity
education
Artificial Intelligence

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

See next page

The full potential of computers in education can only be realized if we look beyond our current educational philosophy and methods of teaching. We believe children should learn in school as they do when not in school: experientially, by trying and failing. Experiential learning is motvated by the student instead of the teacher and provides a fertile ground for creativity. We show how a new type of software, intelligent simulation programs, can support experimental learning by creating a changing learning environment that is interesting, challenging, and rewarding. Based on years of research in building programs that understand and learn experientially, we outline a set of specific cognitive mechanisms for creative understanding. We use these mechanisms of prescribe maxims for the construction of future experiential learning environments that will enhance creativity.

OFFICIAL DISTRIBUTION LIST

Defense Documentation Center — 12 copies
Cameron Station
Alexandria, Virginia 22314

Office of Naval Research — 2 copies
Information Systems Program
Code 437
Arlington, Virginia 22217

Dr. Judith Daly — 3 copies
Advanced Research Projects Agency
Cybernetics Technology Office
1400 Wilson Boulevard
Arlington, Virginia 22209

Office of Naval Research — 1 copy
Branch Office - Boston
495 Summer Street
Boston, Massachusetts 02210

Office of Naval Research — 1 copy
Branch Office - Chicago
536 South Clark Street
Chicago, Illinois 60615

Office of Naval Research — 1 copy
Branch Office - Pasadena
1030 East Green Street
Pasadena, California 91106

Mr. Steven Wong — 1 copy
New York Area Office
715 Broadway - 5th Floor
New York, New York 10003

Naval Research Laboratory — 6 copies
Technical Information Division
Code 2627
Washington, D.C. 20375

Dr. A.L. Slafkosky — 1 copy
Commandant of the Marine Corps
Code RD-1
Washington, D.C. 20380

Office of Naval Research — 1 copy
Code 455
Arlington, Virginia 22217

Office of Naval Research — 1 copy
Code 458
Arlington, Virginia 22217

Naval Electronics Laboratory Center 1 copy
Advanced Software Technology Division
Code 5200
San Diego, California 92152

Mr. E.H. Gleissner 1 copy
Naval Ship Research and Development
Computation and Mathematics Department
Bethesda, Maryland 20084

Captain Grace M. Hopper, USNR 1 copy
Naval Data Automation Command, Code OOH
Washington Navy Yard
Washington, D.C. 20374

Dr. Robert Engelmore 2 copies
Advanced Research Project Agency
Information Processing Techniques
1400 Wilson Boulevard
Arlington, Virginia 22209

Professor Omar Wing 1 copy
Columbia University in the City of New York
Department of Electrical Engineering and
Computer Science
New York, New York 10027

Office of Naval Research 1 copy
Assistant Chief for Technology
Code 200
Arlington, Virginia 22217

Computer Systems Management, Inc. 5 copies
1300 Wilson Boulevard, Suite 102
Arlington, Virginia 22209

Ms. Robin Dillard 1 copy
Naval Ocean Systems Center
C2 Information Processing Branch (Code 8242)
271 Catalina Boulevard
San Diego, California 92152

Dr. William Woods 1 copy
BBN
50 Moulton Street
Cambridge, MA 02138

Professor Van Dam 1 copy
Dept. of Computer Science
Brown University
Providence, RI 02912

Professor Eugene Charniak 1 copy
Dept. of Computer Science
Brown University
Providence, RI 02912

Professor Robert Wilensky 1 copy
Univ. of California
Elec. Engr. and Computer Science
Berkeley, CA 94707

Professor Allen Newell 1 copy
Dept. of Computer Science
Carnegie-Mellon University
Schenley Park
Pittsburgh, PA 15213

Professor David Waltz 1 copy
Univ. of Ill at Urbana-Champaign
Coordinated Science Lab
Urbana, IL 61801

Professor Patrick Winston 1 copy
MIT
545 Technology Square
Cambridge, MA 02139

Professor Marvin Minsky 1 copy
MIT
545 Technology Square
Cambridge, MA 02139

Professor Negroponte 1 copy
MIT
545 Technology Square
Cambridge, MA 02139

Professor Jerome Feldman 1 copy
Univ. of Rochester
Dept. of Computer Science
Rochester, NY 14627

Dr. Nils Nilsson 1 copy
Stanford Research Institute
Menlo Park, CA 94025

Dr. Alan Meyrowitz 1 copy
Office of Naval Research
Code 437
800 N. Quincy Street
Arlington, VA 22217

LCOL Robert Simpson 1 copy
IPTO-DARPA
1400 Wilson Blvd
Arlington, VA 22209

Dr. Edward Shortliffe 1 copy
Stanford University
MYCIN Project TC-117
Stanford Univ. Medical Center
Stanford, CA 94305

Dr. Douglas Lenat 1 copy
Stanford University
Computer Science Department
Stanford, CA 94305

Dr. M.C. Harrison 1 copy
Courant Institute Mathematical Science
New York University
New York, NY 10012

Dr. Morgan 1 copy
University of Pennsylvania
Dept. of Computer Science & Info. Sci.
Philadelphia, PA 19104

Mr. Fred M. Griffee 1 copy
Technical Advisor C3 Division
Marine Corps Development
and Education Command
Quantico, VA 22134

Dr. Vince Sigilitto 1 copy
Program Manager
AFOSR/NM
Bolling Airforce Base
Building 410
Washington, DC 20332

Abstract

The full potential of computers in education can only be realized if we look beyond our current educational philosophy and methods of teaching. We believe children should learn in school as they do when not in school: experientially, by trying and failing. Experiential learning is motivated by the student instead of the teacher and provides a fertile ground for creativity. We show how a new type of software, intelligent simulation programs, can support experimental learning by creating a changing learning environment that is interesting, challenging, and rewarding. Based on years of research in building programs that understand and learn experientially, we outline a set of specific cognitive mechanisms for creative understanding. We use these mechanisms of prescribe maxims for the construction of future experiential learning environments that will enhance creativity.

Table of Contents

The knowledge we acquire is biased by the kinds of questions we ask. It is futile to search for answers unless one is asking the right kinds of questions. Often one cannot even understand the answers that others have discovered unless one has posed the question oneself. This idea has profound ramifications for education at every level. Children naturally pose their own questions all the time, but most educational systems are oriented toward handing down answers, as if being educated means knowing all the answers. Children educated in this system eventually stop asking questions altogether.

We propose that learning in schools should be experiential, learning by doing, and this experience should be supplied by an entirely new type of software: intelligent simulation programs. We attempt to replace the view of the computer as an electronic textbook for handing down answers with the view of the computer as a powerful medium for exploring ideas. We show how this exploration can lead to creativity and we use a cognitive theory of the mechanisms that drive this creativity to derive a new standard for computer-based teaching.

Computers in Education

Everyone thinks computers should be used in schools but no one knows exactly how. Many experts have opinions about how computers should be used, but few really seem to comprehend what the fuss is all about. Everyone believes that children must learn about computers, or must learn to use computers, or that computers will cause a revolution in education. Why is the state of awareness about computers and their potential in education poorly understood?

The problem is that when something new is invented, people's opinions are formed by what they perceive to be the immediate benefits and drawbacks of that invention. They evaluate the invention on the basis of how that invention appears at the time. It is very difficult for even the most englightened observers to base their evaluations on what may happen or what could happen in the far distant future. It is even more difficult for them to make their evaluations if they haven't been able to see the right form of the invention to evaluate. Who could imagine the effects of the gasoline engine when it hadn't yet been installed in its first automobile, yet alone in its first airplane? The status of computers in education is in a similar state: everyone is thinking about how the computer might be used, but few understand the computer's actual potential.

Computers in Education: The Problem

But computers are already in the schools, you say! Everyday, in many schools that consider themselves quite advanced and enlightened, children troop down to the computer room for their time on the computer. Why do they go there? The reasons are complex:

Reason 1: They go because it is there

In today's competetive world, schools are rushing to beat each other to the punch. They are interested in proving to parents, school boards, legislatures, alumni, and the like, that they are right on top of the new trends, that they have bought the latest computer and that the students are using it. In that case, of course, students had better use the thing.

Reason 2: They go to become computer literate

The teachers, parents, and principals of the current crop of school children consider themselves to be computer illiterate. When we were tinkering with the monstrously slow and awkward computers of the fifties and sixties, our peers, the vast majority of parents of current school children, were carefully avoiding such machines and even the people who played with them. Now they suddenly believe that they made a horrible error. Of course, for the most part these people are no more interested in computers than they ever were, but now they at least believe that their children should know all about them. And why should their children know about them? Because its the Computer Age. Because not knowing about computers will make one illiterate in the 80s and 90s. But most of all, because parents are frightened that somehow the world is passing them by.

Reason 3: They go to play computer games

And what do the little tykes do when they get down to the computer room? What is all the fuss about? When they rush home to tell mommy "I used a computer today!" and mommy is so thrilled, what does it actually mean that they did? For the most part it means that they ran a few game programs or learned to use a word processor.

The reason for the multitude of educational games is very simple. Games are fun and school subjects need to be learned, so why not just mix them together to make learning school subjects fun? Unfortunately most computer games are either poorly disguised workbooks or involve game components that detract from the subject matter. Few software manufacturers give thought to the student's role as a learner. Even fewer software manufacturers think of computer games as a way to promote active discovery of new ideas on the part of the student.

Learning to run programs can be very important; word processors are the typewriters of the future. But how excited would Johnny's mother be if he came home and said "Mommy, today I learned a modern typing technique!" No, Johnny says he has learned "word processing". Mommy is overjoyed that Johnny has been launched into the computer age while Johnny has actually learned how to use a simple application program.

Reason 4: They Go to Learn to Program

Any good computer class teaches children to program. But should children learn to program? Well sure, why not? Its a useful mental exercise, sort of like learning mathematics or Latin. No. Perhaps that demeans it. Learning to program means learning to produce step by step procedures that emulate processes. Kids can come to understand how something works by writing a program to do it because writing a program forces them to take the time to do a step by step analysis.

Learning to program really consists of two parts: learning to solve problems by thinking of structured solutions and translating those solutions into a particular computer language. The first part consists of learning to formulate problems into specifications, developing algorithms to carry out those specifications, and choosing appropriate representations. This is where the most thinking is involved because these activities apply not only to programming but to many other intellectual activities. But what do schools teach in computer class? Schools teach children the available constructs of specific languages and how to translate algorithms into those constructs.

The problem is not just with the schools. Look at virtually any beginning programming language textbook. The book will introduce each construct in the programming language followed by a group of problems oriented around that construct. The majority of computer textbooks are written by computer language designers and thus do not teach how to think about problems and test ideas on a computer. These books teach the specific features that the designer was so proud of when he invented the computer language.

Children are taught programming languages not programming. There is a big difference. Learning to program involves thinking: generating potential solutions, finding problems with those solutions, testing ideas, and being wrong. Learning a programming language does not necessarily involve any of this: it often involves memorizing lots of command names and what they do and learning to produce a given input/output behavior. It's not really the teachers' fault that kids don't learn to program either. If the teachers really understood programming they'd go get a job at three times the salary. Teaching is nice and rewarding, but for most people its not *that* rewarding. So, for the most part, programming is taught all wrong in the schools.

The state of educational software is no better. Software companies are making electronic workbooks and drill and practice software and educators are buying these unimaginative applications of computer technology. Why?

1. *They are available* - It is easy to make a workbook into a computer quiz. No new lesson planning needs to be done. Companies can manufacture this software easily and inexpensively.

2. *They are easy to teach kids to use* - To use an electronic textbook product, you just have to teach the kid how to press "RETURN" and how to press a number between 1 and 10 and you are in business. What could be simpler?

3. *They are familiar* - If something looks like a workbook or a textbook, it is more familiar to educators. Some computer companies want their products to approach traditional teaching media as closely as possible so that teachers will accept the products more easily. This kind of attitude stifles innovation.

We must not form analogies of computers to books, desktops, typewriters, or even teachers. Instead we must take a serious look at education, decide where the problems are and see if computers can help. Computers are not just storehouses of information or devices for increasing the speed or drill and practice exercises. Computers are a means of exploring worlds that are not otherwise accessible to children. Computers are active and responsive; children can generate their own ideas and test them on the computer. When we stop thinking of how to insert the computers into the curriculum and start thinking about how to change the curriculum altogether, we open up a whole new array of possibilities.

Computers in Education: The Solution

What are and will be the advantages of programs as educational tools? Computers are a new medium for displaying information, they allow schools to access information easier and more efficiently , and they are tireless teachers that can repeat lessons again and again. But these are short-sighted views of the power of the computer in education.

We need to look beyond fancier graphics, larger data bases, and better drill and practice software. We need to examine how children learn in school and out of school. As long as we think of learning and computers separately we will not utilize the computer's potential for teaching.

Experiential Learning: Learning by Doing

There are many different kinds of learning found in the schools: children learn the multiplication tables by rote memorization, they learn addition and subtraction by inducing a general procedure from examples. Children learn how to write the alphabet by imitation, they learn facts about history and science by observation, they learn how to play piano by repeated practice, and they learn formulas and procedures by direct instruction.

What is largely missing from the schools, yet is found universally when children are left to themselves, is learning from *experience.:* learning by trying and failing. Consider learning how to solve a puzzle , like Rubic's cube, by trial and error -- or learning to play your older brothers' harmonica with no help from others.

Experiential Learning: Advantages

Anyone who has been around children knows that they learn experientially all the time. As Lepper (1985) points out, intrinsic motivation makes learning more productive. In addition, experiential learning can be very rewarding: consider how proud a child is of their first art project or how excited they are to tell their friends about a new method of jumping rope. If the school system made more subject areas accessible through experiential learning, children would enjoy schools more, would be more motivated, and would have a great freedom to be creative.

Experiential Learning: Problems

Given these advantages of experiential learning, why hasn't it replaced traditional instruction? There are many reasons:

1. time - Experiential learning seems very slow and unproductive to a teacher. It requires letting the student try things that will obviously fail. But the results of experiential learning are often remembered better and the failures are often useful learningexperiences, but the problem remains that most experiential learning is time consuming because the learner never quite finds themselves in an identical situation so that they can see what went wrong and avoid it or see what went right and repeat it.

2. applicability - The knowledge in many domains is either not accessible for direct manipulation, as in mathematics or economics, or it is too dangerous for unsupervised experimentation, as in chemistry and biology.

3. motivation - Children become disinterested and at critical times must be spurred on to learn more by a friend or a parent.

4. pace - A teacher can't keep up with 30 students each on different materials, with different problems, learning at different rates.

5. confidence - Some students have bad initial experiences and are turned off quickly. Without the help of a teacher, they can lose their self-confidence and become isolated from furthur experience.

These problems are not intrinstic to experiential learning, but are instead problems of using experiential learning in a classroom with one teacher and many students. In fact, all of these problems go away when the medium of experience is a computer program that is providing a specialized environment to a single student. With software that can provide one-on-one instruction and allow exploration, experiential learning should be embraced by the schools as the primary method of education using the computer.

Solving the Problems: Intelligent Simulation Software

If computers are to become vehicles for exploration of ideas, we need to establish what type of programs will encourage experiential learning and what is required of these programs to make an impact on education at all levels.

Experiential learning requires that a task be performed by the student, that it is open-ended and at least partially under the student's control, and that the student can try out new ideas and get feedback on them directly. *Simulation software* meets these requirements well: the student performs a task in the simulated environment, they control what experiments they want to perform, and they get feedback directly from the simulation.

SIMULATION SOFTWARE

Simulation programs have seen their debut for airplanes and war games. But what about a program that simulates a cooking world and lets student chefs compose new recipes and try them out? Or a program that simulates the process of constructing a bridge so that student civil engineers can see the stability of their designs by trying and failing. Simulations can provide unlimited experiences and their open-ended nature permits creativity.

Simulation programs can also make abstract worlds accessible to the student through examples and analogy. Programs can teach children probability theory by having them bet on outcomes of games [Goldstein 82]. They can introduce children to the federal legal system or the business world by simulated interactions, letting them have practice at being a federal judge or a creative entrepeneur. Programs can introduce children to a famous person of the past and encourage them to ask questions to learn about history from that person's perspective. Simulation programs open up an exciting world to the student where they can construct their own ideas and try them out without criticism and fear of failure.

STUDENT MODELS

Providing interesting simulation software is not enough to make a major impact on education, however. Educational software should include a model of what the student knows and doesn't know so that it can tailor problems to the interests and background of the particular student. Math programs should start out by encouraging students to pose sample problems involving new math concepts and then help them solve these problems. Later, if one of these problems appears again, it should permit the student to skip over the problem if it knows the student has already mastered the concepts involved. Good educational software should permit a large number of possible interactions so that students at different levels can proceed through different material, not just the same material at a different pace. Student models should be used to hand-tailor simulation environments so that they are always interesting and challenging.

INTELLIGENT SIMULATION SOFTWARE

We propose that *intelligent simulation environments* become a new standard for computer-based teaching. An intelligent simulation environment is a computer program that combines a simulation program with a student model - it can manipulate the environment the student is working in to make it more challenging and more appropriate to the individual student. If we go back to the problems with experiential learning, we see that they can all be solved by providing the student with intelligent simulation software:

1. time - Intelligent simulation software should make attempting
problems easier and completing them faster, resulting in much more efficient experiential learning.

2. applicability - Intelligent simulation software will make abstract domains accessible and dangerous domains harmless.

3. motivation - Computers will become friends to students, encouraging students to try new things , constantly confronting them with new experiences so they are not bored.

4. pace - Students work at their own pace and are monitored by the program, so there is no need for all students to work at a similar pace.

5. confidence - Simulations with a student model will be able to organize experiences so as to build the student's confidence. More difficult problems with more chance of failure can be postponed until the student has acquired a positive attitude and is more familiar with how to experiment and have fun with the program.

To derive more specific requirements for intelligent simulation software, we need to examine the process of experiential learning in more detail.

Experiential Learning and Creativity: Lessons from AI Research

After nearly a decade of research, we now believe we understand the process of learning by experience enough to apply it to educational objectives. This understanding has come from both the construction of computer programs that learn and from the examination of data obtainable from people.

The Nature of Everyday Understanding

We consider processing mechanisms to be bundles of expectations --- expectations about what someone will do next, about how a given situation is structured, about the likelihood of various events occurring, about what word or sound is coming next in a sentence. In short, people have bundles of expectations about everything. When these expectations are satisfied, when something happens just the way you thought it would, processing proceeds normally, sentences are parsed, you get what you asked for, your predictions about what will happen next are satisfied. But, the news in that situation is not all good. When things happen the way you expected, no learning takes place.

Expectation Failure

Learning takes place when failures occur. When an expectation fails, when something doesn't go the way we expected, even if we expected to not achieve a goal or unexpectedly succeeded at the goal, we must attempt to explain what happened. Learning is very tied up with the concept of explanation [Schank 86]. When events in the world can be explained by prior events, that is when old standard explanations suffice, learning does not occur. When one is learning, one is learning, among other things, new explanations. One is learning why something happened so as to be able to make more accurate predictions in the future.

Reminding

Another phenomenon occurs in this paradigm of expectation failure and explanation, namely, the phenomenon of reminding [Schank 82]. At first we observed that people would report that one situation reminded them of another, quite different situation, and we weren't quite sure why. Later we noticed that the two situations were almost always linked by having an identical expectation failure and an identical explanation of the expectation failure. So, it seemed clear that explanations of expectation failure were likely to be some of the indices that human memory uses for retrieval of old information.

But, why was the reminding occurring? Our first answer to this question was that the mind was bringing up other relevant data to be considered when an expectation failure occurred. The idea was that if you were trying to understand something peculiar, it might be easier if one had additional data and then made a generalization that held for both.

Our second idea was that reminding was a kind of verification for already hypothesized explanations. That is, once an explanation had been proposed by the mind for an expectation failure, other memories were brought to mind by the use of the explanation as an index so as to see if that explanation related to anything else that we knew about.

Explanation Tweaking

It seems clear that reminding does occur in both of these ways, but now we are beginning to see that reminding is part of the more general process of experiential learning. A great deal of human reasoning ability is bound up in our ability to find prior reasoning experiences and relate them to the current situation. A great deal of everyday planning involves solving problems by relying upon a repetoire of previously solved problems cleverly indexed so they come to mind at just the right time. Of course, these problems are not identical to the one being considered at any given moment, so after one is reminded of a somewhat relevant one, the issue is to attempt to tweak it into relevance. In other words, various tricks for getting it to look like the problem actually at hand are employed, and when the identity is made, a candidate plan or explanation has been found.

Often these candidate explanations are quite silly. One cannot be sure one has found the right initial problem, through reminding, from which to start tweaking. So, often the issue is how to find another prior problem for tweaking.

This having been said then, it seems clear that there are a great many important general phenomena which come about from getting reminded of specific prior experiences. One must know how to look at a problem in such a way as to be able reformulate it many times in the search for new indices that will yield more old problems to tweak. One must have a storehouse of what we call explanation patterns (XPs) , that is, good old standards that can be adapted to new situations [Schank 86]. Finally, one must be able to change these patterns in such a way as to adapt them into being relevant to situations in which they may not obviously apply.

Creative Understanding

Why is understanding the nature of reasoning from experience so important? Because we believe that reasoning from experience is the starting point for creative thinking. And it seems clear to me that we want to teach creativity to children. Before we do, however, we must understand the nature of creative thinking.

What Is Creativity?

Most people believe that creativity is something rare and impressive. They believe you must have a gift for it. With this outlook, understanding creativity seems impossible, let alone teaching creativity.

But creativity is really just coming up with your own idea. Having your creativity be recognized by others as true creativity depends upon the acceptability of your idea and the area of inquiry in which that idea occurred. We award the notion of creative to scientists and artists. To sports handicappers and farmers we are somewhat less expansive in our use of the term. The premise is it is all the same process.

In general, laymen tend to think of creativity as something mystical, beyond the reaches of the explainable. But, what this really means is that there seems no obvious algorithm to describe creative processes. Our premise here is that such an algorithm must exist, in principle. The reason that such an algorithm may be assumed to exist is, first and foremost, because people are capable of being creative. They come up with new ways of looking at things, with new ideas, by some method. It can't all be simply magic after all.

One reason to be hopeful about finding this algorithm emerges when one considers that most creativity is a process of evolution from prior discoveries. People rarely invent things completely out of the blue. Most creative acts are, in essence, modifications of old ideas that have preceded the new idea. So, while it may seem like a tall order to expect to find the rules for the creation of new ideas, dividing the task into two sub-processes can make the problem much more tractable. First, find the old ideas and second, modify those old ideas to the new situation.

Reusing Old Ideas: Explanation Patterns

We said that much of our ability to understand comes from our ability to produce explanations when our expectations are not met. Our premise, with respect to the explanation process, is that all explanations are essentially modifications of old explanations. Furthur, the claim is that people are stockpiled with old standards, explanations that they have heard so many times that they use them without thinking about them. Thus, the claim is that creativity, with respect to the problem of explanation, can be reduced to two sub-processes. The first, which is inherently a search process, is finding a candidate explanation pattern, that is an XP which might be modifiable in some creative way to help with a current problem. The second, which is inherently an alteration process,is modifying and adapting that XP. A creative thinker must be able to do both well.

Explanation Patterns: Search

What would it mean for search, or alteration, to be considered to be a creative process? In a sense, any creativity that would exist in those processes would be with respect to the novel techniques used in that process. And, of course, novelty being relative, the issue becomes one of the lack of usualness of the techniques being employed. This seems like an odd measure of creativity. Or is it? Perhaps creativity means no more than the application of a technique or rule where one would not expect to apply it. If this is true, we can see why creativity is not being fostered by the current school system. Children are taught that there are rules to follow and to use these rules only in cases where they are applicable.

For the search for XPs to be useful, there must be a large stockpile of explanations that have been proven useful in the past. People are inherently lazy in their use of these old patterns: if some new situation can be seen as similar enough to an old one, people will try to use the explanation for the old failure and make it suffice. Much of the point behind the work on memory reported upon in [Schank 82] was to show how people use past experiences to help them interpret new experiences. If Burger King can be seen as somehow like McDonald's, the rules for operating in McDonald's can be adopted and modified. Also, seeing one situation as an instance of another is helpful in making generalizations and thus in learning about both situations. The fact that generalizations are often inaccurate does not stop us from making them and does not lessen their value. In explanation, a similar phenomenon occurs.

We rely upon explanation patterns to create new explanations from old explanations. This at once makes the process of explanation easier and makes its precision considerably less than ideal. It also makes creativity possible.

Explanation Patterns: Indexing

The key to inventing creative explanations lies primarily in intelligently indexing the XPs. One way to explain something unusual is by reference to something different for which there exists an explanation. So, one way to find a candidate set of XPs is by changing the event that is to be explained into one that is like the original event but is enough different that it might bring up a new idea that is relevant. In this way we have the possibility of finding additional XPs that are not connected to all the indices at hand, but might be relevant. To find these additional XPs, we attempt to change the event that needs to be explained into another event that we know we can explain.

Thus creative explanation depends on having a large library of XPs intelligently indexed so that they come to mind at just the right time. We get them to come to mind by reformulating the original problem so that more indices are available. Again, we see how school systems have de-emphasized creativity: students are not supposed to change the problems that teachers give them. They are not encouraged to relate new problems to their previous experience: they are just supposed to absorb the new material.

Explanation Patterns: an Example

To see how explanation patterns work, let's look at a standard explanation for death, KILLED FOR THE INSURANCE MONEY. This pattern is useful when it is necessary to explain why someone was killed. If the person who is dead is worth more money dead than alive, it is reasonable to assume that he was killed for the insurance money. We would expect that many people might have such a pattern and that using that pattern would not require a great deal of explanation in itself. In other words, it is like the scripts we introduced in Schank and Abelson (1977) in that one needn't think much about it in order to operate with it.

Even when a pattern is idiosyncratic and ill-formed, it is useful for reasoning. That much is obvious, since people do use such rules to explain the behavior for which they were concocted. The question with respect to explanation patterns however, is whether and how those patterns can be applied to situations other then those for which they were originally intended. The real issue in the use of explanation patterns is how to misapply them.

Explanation Patterns: Misapplication

The issue in misapplication is embodied in the notion of a partial match. It is all well and good to establish that someone has been killed and that he had a large estate that was to go to his evil son-in-law and to thus suspect his son-in-law. Such a suspicion comes from matching the pattern, KILLED FOR THE INSURANCE MONEY, and there is really no more to say about such a match other than such patterns must exist in memory in order for such matches to occur.

But the really interesting case is when KILLED FOR THE INSURANCE MONEY is used to explain a set of circumstances that are superficially quite different than those of son-in-law, rich man killed, and so on. Suppose, for example, we were trying to explain why Swale, the best thoroughbred racehorse of 1984, the one who had been winning the most important races for 3 year-old horses, suddenly died in his stall. KILLED FOR THE INSURANCE MONEY might come to mind even though we know that racehorses don't usually inherit money. But this seemingly unusable XP might bring to mind that owners do have insurance policies on their horses, so this hypothesis might yet be viable.

Following our previous analysis, the way to retrieve useful explanations is to transform the situation and thereby generate unusual indices. The original anomaly of a successful racehorse dying young can be transformed by looking at Swale not as simply a thorougbred racehorse, but as a number of other related categories. SWALE could be seen as a successful performer to access an XP like PERFORMANCE-RELATED-OVERDOSE or Swale could be seen as a competetitor in a contest, suggesting DISABLE-OPPONENT-IN-CONTEST. Being open-minded about how to view a problem is one of the tricks to creative thinking.

The idea is that frozen patterns that otherwise look simple and uncreative, taken out of context, can shed light on new domains of inquiry. That is the philosophy behind the misapplication of Explanation Patterns. When one applies patterns where they do not obviously belong, interesting things can result. Therefore, when teaching creativity, we must teach children to think of alternate ways of viewing problems so as to access solutions that might be tweaked into being useful.

We can see then that the major role of Explanation Patterns ironically, is not in their intended use. XPs are fossilized reasoning. They represent our intention not to think very profoundly about a subject. When we use an XP in its intended role, we are deciding to forego complex reasoning of our own in favor of using a well established reasoning chain that is in favor amongst a particular group of reasoners.

Explanation Patterns: Modification

After a hypothesized explanation has been derived from an XP, we need to begin to modify aspects of the hypotheses to make it fit the current case. Thus, we must take the hypotheses created in the last step and see if they make sense. The hypotheses, were created by finding what seemed to be relevant Explanation Patterns. Now the task is to see if they are truly relevant. To do this, we must alter the parts of the Explanation Pattern that have nothing to do with the actual case. The result is an explanation that applies to the situation at hand and is also suitable to be added to our storehouse of common XPs. Thus tweaking the XP into relevance can create a new general pattern. In the case of SWALE, for example, we might create a pattern for owners of expensive performers killing their entries and therefore sacraficing fame for money.

Creativity: Getting Started

If our theory of creative thinking is correct, then creativity depends on two primary factors: a set of methods for getting reminded and a set of methods for adapting remindings in such a way as to fit the new situation. Plus, one must have a creative attitude. This attitude manifests itself in the ability to keep alive obviously errant remindings or obviously irrelevant XPs long enough to see if they really are useless. Rejecting all ideas when they are not immediately applicable is a certain way to avoid creativity.

Finding Anomalies

There is a certain amount of creativity to the explanation process itself, but perhaps the most important part of this process is getting it all started. To be creative, one must notice that something is wrong, that somewhere, your expectations are not being met. Creativity and learning derive from the need to correct failures and understand anomalies in the world. We can create solutions, correct failures and explain anomalies only by identifying where we have been wrong.

When our theories fail, we feel compelled to wonder why, and to attempt to explain the failure. When we explain something to ourselves, we have the potential to learn from the explanation. Explanation is a kind of self-taught learning. In order to learn we must have something we want to learn. In order to want to know something we require a failure of some sort to show us our own lack of knowledge. Failure in this sense means a failure to have had the knowledge that would have eliminated the failure.

Explanation is a process of accounting for our failed previous explanations. When our explanations and theories break down, we must explain the failure in such a way as to modify the thought process or the aspect of our behavior that was in error. Finding just which aspect is most significant can be a serious problem however.

To teach students to be creative, we must teach them to become aware of just how wrong everything is. They must notice when things around them don't work. They must seek out anomalies in the world around them, in people's behavior, in their own behavior. They must wonder why they do what they do every day. If they have been going through school thinking that everything is fine, this might be a shock to them.

A person's previous understanding of the world or any part of it is obviously crucial to his future understanding of it. When a person's new experiences fit nicely into the framework of expectations he or she has derived from experience, the understanding process seems simple enough.

But new experiences don't always fit nicely into our expectations. Experience is often anomalous in some way, which simply means it fails to correspond with our expectations. When this occurs, we must re-evaluate what is going on. We must attempt to explain why we were wrong in our expectations. The explaining of our failed expectations is the only way we can grow as a result of our experiences. Learning thus requires expectation failure followed by explanation.

Expectation failure is not always simple. A curious or anomalous situation may require that we re-evaluate a tremendous number of expectations, rules, theories and beliefs that relate to the situation. People are constantly questioning themselves and each other, to find out why someone has done what he has done, and what the consequences of an action are likely to be. In order to find out how we learn, we must find out how we know that we need learn. How do we discover anomalies? How do we know that something doesn't work?

Asking Questions

The answer is questions. When we see a pattern in the world we ask a set of questions about. If we get the standard answers back, no more processing needs to be done. But if we get new answers back, we need to do some thiking. Questions lead to new thoughts, answers only to the end of thinking.

Schools should teach children the questions, not answers. Explanations are only creative when we come up with them ourselves. And what does this tell us about how teaching should be conducted? Every time we want a child to learn something, we must figure a way to make him ask about what he wants to know by himself. And, we must encourage him to speculate for himself about what the answer might be. In other words, in order to learn, a child must want to know, question, create an explanation, check to see if its right, and if it isn't, he must try again. This is how we learn naturally, and it should be how we learn in school as well.

From what we have seen, we can derive a set of 5 components beyond those needed for normal understanding, that are essential to being creative:

1. Not being afraid of failure

2. Seeking out anomalies by asking questions

3. Knowing ways to transform problems so as to get reminded

4. Keeping alive errant hypotheses

5. Knowing ways of adapting hypotheses to new situations

Teaching Creative Thinking

These components must be integrated into the school curriculum if we are to produce creative thinkers. First we will look at the problems associated with introducing these components into the schools and then we will try to show how computers can be a solution to these problems.

Why Aren't Children Creative in School?

Many children come to think that school is a place where you are asked questions and you must quickly produce "the correct answer" by simply recalling it from memory - no thinking allowed. In school, the teacher expects answers to questions. She asks who discovered America and wants the name Columbus, not some hedging about Vikings, or comments about American Indians.

This stereotyped role for school thinking can be seen most clearly in this quote from a five year old:

> *"Wendy was taking a ride with her father. They passed a cement truck, and Wendy said: "That's a cement mixer. It has sand and water and cement in the round part and that part goes around so the sand and water and cement will get mixed up and stay mixed up. " She went on to tell where the cement mixer might be going and what would happen when it got there. After giving all this information, she looked at her father and said, "Do you know how I know this? I thinked it." Then she said thoughtfully, "You don't have to think in school. The teacher tells you."*

Source: Mary Lee Marksberry, "Sizing up Assertions," *Elementary School Journal,* vol. 76 (1976) p. 289

In the schools, hypothesis-making is referred to as fanatasizing and is strongly associated with lying. If something isn't right, it's wrong. Children aren't supposed to speculate on what a cement mixer might be used for or how it works. They are supposed to just "know". The result of this approach to teaching car be felt most in deprived neighborhoods, as evidenced by the studies of Bereiter and Englemann in *Teaching Disadvantaged Children in the Preschool.* They studied a group of children from low-income communities and noticed that children had the tendency to answer questions without breaking them down into seperable components or making hypotheses. They called this "one-shot thinking" because the children took one shot at each question rather than thinking them through.

The childrens' background did not teach them to perform conceptual operations - to compare, combine, and translate. The children had not developed the technique of internal dialogue, whereby they ask themselves questions - "Is this true?" "Does it have that characteristic?" - in proceeding through a step-by-step sequence of deductions. Thus they could easily memorize the rule "If it has a beginning and an ending it is a word", yet even after learning to identify beginnings and endings, they could not consistenly apply this simple rule. They were not in the habit of systematically asking the question "Is it a word?" and breaking it into subquestions "Does it have a beginning" "Does it have an ending?" "Did I answer yes to both questions?".

Source: *Intelligence Can be Taught,* Whimbey and Whimbey

With these children, if the answer to a problem was not immediately available, if it required any decomposition or step-by-step reasoning, the problem was abaondoned immediately. In fact, Bereiter and Engelmann reported that the children believed the answer to a question *should* be given immediately, rather than after a certain amount of thinking!

The empirical research on the difference between successful thinkers and unsuccessful ones is not limited to young children. In a largely-ignored study by Bloom and Broder (1950), thinking-aloud protocols were taken of high IQ students and low IQ students in several colleges across the country. There were two key features they found different in the reasoning of the student groups - the low group performed one-shot thinking rather than question-asking and answering, and the students allowed gaps of knowledge to exist without exploring them, so they didn't notice anomalies and didn't even think of asking questions to resolve gaps in their knowledge.

From these and other studies, the evidence is clear: lack of creative thinking comes from one-shot thinking that is perpetuated by schools and by parents, particulary in low-income neighborhoods. Students are afraid of not getting the right answer, they don't take note of gaps in their knowledge, and they immediately reject any hypothesis that is not already known to be correct.

Teaching Creativity: What Doesn't Work?

How can this go on? Educators and psychologists have been worrying about the role that creativity plays in classroom study for several decades. Why does the school system remain largely indifferent?

Determining The Factors

Several researchers over the years have had an interest in fostering creativity in the classroom. Torrance gave a set of guidlines for teachers to let creativity flourish in the classroom (Torrence 65). He suggested developing a tolerance for new ideas, providing for active and quiet periods, being aware of forcing a set pattern upon students, teaching skills for avoiding peer sanctions, and dispelling the sense of awe of masterpieces. These absence of these factors were seen to be detrimental to creative students, but researchers could no justify their guidelines because they had no theory of the creative process. The best they could do was to increase factors they found promoted creativity and decrease those factors that seemed to stifle creativity. Which factors were important could only be determined by a large amount of experimentation and testing.

Testing for Creativity

There have been a number of tests that attempt to measure creative potential, The Torrance Test of Creative Thinking and Mednick Test of Remote Associates, for example. Torrance and others wanted to understand creativity qualitatively and quantitatively by designing tests that correlated with our intuitive feelings about creative aptitutude. But being able to measure creative aptitude is quite a different matter from being able to understand and manipulate it through schooling. Torrance and others only wanted to provide opportunities for children that were already creative. And to a large extend they were successful: there are many gifted and talented programs available in public schools today. But we are suggesting something much more grand and yet within reach: an established curriculum for increasing creative potential through the use of educational software.

Applying Cognitive and Educational Theories

The fact that previous endeavors did not ever make a major change in our school systems stems directly from 3 key problems:

1. *Cognitive psychology has been producing non-prescriptive theories.*

The non-prescriptive cognitive theories allow us to evaluate students or put them into categories, but they don't give advice about how to generate useful curricula. These theories only tell us what the classes of behavior are, not how to change a person that is in one class into another. For example, it is nice to know that short-term memory is limited to a capacity of 7 digits and that brighter students have larger short-term stores, but what does this tell us in general about teaching? Is short-term memory capacity a cause of learning problems or is it a reflection of some deeper problem? Should we try to increase kids' short-term memories? If this theory just tells us that we shouldn't overload our kids with information, we already knew that!

2. *Educational psychology has been giving advice grounded in theories unrelated to cognitive functioning.*

Educational theories have largely been oriented at increasing classroom variables that increase scores on standardized intelligence tests. However, we now know that standardized intelligence tests don't measure individual aspects of intelligence [Sternberg 85], they only give us numbers for factors that are correlated with success on other tests and on performance in school and in the workplace. The tests give no way of delineating the various cognitive abilities that constitute intelligence. Hence, educational psychologists' advice usually comes from intuition about what changes in the classroom will improve scores on the tests and on subsequent testing to see if those changes actually do improve scores. It is not surprising that educational psychologists have problems making a major impact on the school system with such a slow and indirect method of making recommendations.

3. *Advice from cognitive and educational psychology rarely makes an impact on actual classroom instruction.*

Even if a psychological theory does generate advice, to an educator it is just another theory. Even well-proven psychological results are extremely slow to be applied in the schools. Teachers rarely consider the cognitive psychological aspects of their teaching; they simply wait for books and teachers guides to arrive that embody the new ideas.

Teaching Creativity With Computers

Fortunately, the computer has the promise of changing all of these impediments to progress. First, computers make testable process models possible and hence prescriptive theories are easier to construct. Second, ideas from the artificial intelligence laboratory and psychology laboratory can be translated into programs directly and these quickly become available as tools for teachers. In addition, by making the computer record its interactions with students, psychologists can get feedback from the educational world more directly and hence can be more aware of the actual teaching performance of their programs.

But what has really been missing is a detailed analysis of the mechanisms behind creativity that can allow us to be more confident that our attempts at fostering creativity will work. We think that the analysis of how people understand creatively given in this paper will suggest important and significant changes to the current school curricula and we think that these changes can be most quickly brought about by computer technology. To see why, let's look at some ways we can change the way things are done.

Create a Fail-easy Environment

We agree with DeBono (1969) that the fear of being wrong is the greatest deterrent to the ability to come up with new ideas. Making children start learning for learning's sake and not for high scores on standardized tests will take a major effort on the part of teachers, parents, and students. There are a number of ways this can be accomplished.

UNTESTED LEARNING

With computers monitoring students' creation of ideas, tests will not be as critical as they have been in current educational practice. Teachers can review the progress of individual students by watching a trace of their performance, recorded by the program.

If creative thinking is the goal, then reducing the quality of students' ideas to a few numbers is self-defeating. Instead, teachers should play the role of coaches: helping students along with ideas they are having trouble grasping, and encouraging them to try new things. Tests and graded homework can be used less because of their decreased importance in the learning process. Tests should be disassociated from the "idea creation" time with computers. With appropriate support from the home and an enthusiastic and patient teacher, this way of utilizing computers should result in children who are not afraid of failing.

ENCOURAGING FAILURE

Once students have learned that failure is OK, they should actually be *encouraged* to fail. Encouraged students to fail? That's ridiculous, you say! That is counter to our entire educational philosophy!

We contend that part of the fear of actually helping students to fail at tasks is tied up in the behaviorist tradition in psychology. Skinner, in his dominating book, *The Technology of Teaching,* argued that failure should never be encouraged. Rewarding failure, he argued, confuses students and decreases their learning rate.

Skinner was probably right when the tasks being done are tests that drill students on answers to questions with clear-cut answers. But answers, especially those produced by learners, are rarely right or wrong. Seen as hypotheses constructed by students, there is often a grain of truth in incorrect answers. We must learn the reasons behind students' answers and reward the reason if it is right, not the result.

By encouraging students to generate ideas and test them, students will fail more. This failure will only be profitable if the student understands why he failed. Teachers should get students into the habit of explaining their failures back to them and later to themselves. Only by explaining failures can those failures be turned into learning experiences. Students should be guessing outcomes in history class and comparing their guesses against what really happened, not simply learning the facts. They should be predicting the results in chemistry class and then seeing if their predictions come true. The thinking done before and after studying, experimenting, or solving is the most important part of the overall process and we must encourage it.

NOT ALL ANSWERS KNOWN

There are really 2 types of questions that crop up at home and in the schools. First, there are questions from students who want the answer so they won't have to think, so they can spit out that answer later on a test. Then there are inquisitive questions that could lead to learning a new concept, fact, or rule. Often teachers and parents mistake the second type of question for the first type of question. They quickly give the student the answer without making a learning experience out of it.

Adults who quickly answer "Where do laws come from?" with "The Congress", with an emphasis on heading off more questions, are just encouraging the idea that there is one right answer to every question. Children come to think that the right way to learn is to catalog the answer to each question they might be asked in school. When the teacher asks "Where do laws come from?" and Johnny replies "The Congress", everyone is happy - Johnny, the teacher, the parents. But unfortunately, Johnny hasn't learned anything about laws by this process; he doesn't really know what "The Congress" is. Johnny is likely to tune out from the subsequent discussion of the 3 branches of the government because he already knows that laws are made by "the Congress" and this is the answer he must know on the test.

REWARDING QUESTION-ASKING

For children between the ages of 3 and 4, questions constitute between 11 and 28 percent of all utterances [Ross 74] . What happens to all of these questions as children grow older? Do older children want to know less? Have they learned everything they want to know by the age of four? Of course not. When children first come to school, they want to ask questions. But the fear of asking a "dumb" question in front of the whole class overtakes them. Children who ask too many questions are scolded for wasting valuable class time. What would a teacher do with a whole class of kids asking questions?

As Papert points out in *Mindstorms*, with computers children could ask questions and explore answers on their own, without the fear that their questions are "wrong" and without reducing the teacher's effectiveness. If some child wonders how an airplane behaves when it is at top speed, they can just try it, with a simulated airplane on a simulated flight. The bottleneck in software development then becomes the communication gap between student's questions and the machine's ability to understand those questions enough to help the student explore the answers to them. This is a matter of technological advance, but the materials for success are here.

Teachers should never discourage question-asking. They should try and understand the inquisitive intent behind each and every question and think of what the student can do to learn the answer on their own. If a child asks how many meters are in a foot, the teacher shouldn't reply "That's not right. Meters are bigger than feet. There are 3 feet in a meter." Instead, they should say "That's a good question, let's figure it out" and give the child a yardstick and a meter stick to play with in the corner of the room. Or they should pick out a computer program that let's kids explore measurement. Or they should assign a homework project to measure their living room in meters and feet. Any way they can, children should be allowed to explore the answers to their own questions.

Encourage Hypothesis Generation

Hypothesis generation is an important part of the creative process. It is only through hypothesis generation that our previous experience has impact on our current problems. Students need to be taught to notice and seek out problems and once they have these problems, generate hypotheses about them.

ASKING FOR HYPOTHESES

A first step is to continually challenge students with anomalies. One good way to start doing this is to ask children questions that, if they answer, they will discover anomalies and ask furthur questions for themselves. Questions from programs like this should not be used to get children to know the answer, but rather to prompt them to discover interesting relationships and ask why they are true. Thus, these questions should never be asked again on a test and students should know they are not being asked to memorize answers. Rather, they are examples to the student of how to find and explore interesting relationships.

PROVIDING OPEN-ENDED PROBLEMS

Another way to keep students asking questions is to give them difficult problems and make them generate hypotheses in response to these problems. The basic cycle for the interaction would be:

1. Computer poses hard question for which there may be no right answer

2. Student generates a hypothesis.

3. Computer responds with a counterexample from its data base.

4. Until there are no counterexamples, go to step 2

The main point of this process is that the student fails to get the right answer. The computer is continually trying to point out holes in the student's answers. We maintain that a computer can get away with this, but a teacher in a classroom can't. The reason is simple, but compelling: the computer is not judgmental. Children recognize that there is no social stigma attached to being corrected by a computer (especially if everyone is treated that way). However, children are very sensitive to the attitudes of teachers and other students. No child want to continually be singled out as unable to answer a question. The computer has greater latitude. Its interaction is private. The other students aren't aware of the mistakes that any other child has made. Neither is the teacher, at least immediately. Computers offer students a great and important luxury: the opportunity to fail privately.

STUDENTS GENERATING PROBLEMS

Once students have got the hang of hypothesizing answers to problems that are presented to them with no fear of failure, they should start generating problems on their own. The emphasis should be getting the students to generate original problems, so the main job of the teacher is to make sure that problems are not ones that have been solved before by the same student.

Once students have found hypothesis generation stimulating, they will easily catch on to the game of proposing their own questions and problems. They will enjoy "being the teacher". In english class, students should submit inquisitive questions they have about a book they've read and then the teacher can mix these up and give them back to students to generate hypotheses in response to those questions. In chemistry class, students should be free to propose experiments that they and other students can try out on a computer simulation. To bolster reading comprehension, students should read murder mysteries and prompt the class with questions that might lead to identifying the culprit. The first one to get it right and explain why gets to pick the next murder mystery book to read.

The whole point of these exercises is to get students in the habit of posing problems. With no fear of a failure, they should have no fear of proposing problems to solve. Gradually, they will learn to pose problems to themselves and then generate hypotheses in response to them. These are the tools of creativity.

HYPOTHESES AS LEARNING EXPERIENCES

Once students have mastered creating problems and generating hypotheses, they should learn to manipulate hypotheses. They should learn that answers are not to be thrown out immediately if they seem wrong, but are to be manipulated to generate other answers or new problems.

Once children get the idea of hypothesis generation, some children will invariably suggest strange or unrelated hypotheses. Each of these hypotheses should be treated as a learning experience: teachers should point out the truth in each one and from this true part, find new problems for the class. If the question is "Who harnessed electric power?" and a student calls out "Edison!" instead of "Franklin", it might be a good time to introduce Edison and the invention of the lightbulb. This type of interaction takes an enormous amount of time and patience and this is one reason why it is a marvelous application of computer technology. The machine asks a question, the student answers, and the machine tries to use the student's answer to explain a point and introduce another question.

INTRODUCING ALTERNATVE HYPOTHESES

Remember that a key part of creativity is keeping alive errant hypotheses and knowing ways of tweaking these hypotheses to fit new situations. To practice tweaking hypotheses, students must generate hypotheses or be given hypotheses that are seemingly irrelevant and practice working them through. Once students have the idea that they should be generating hypotheses, they should be moved to a program that generates seemingly irrelevant hypotheses that can be subsequently adapted by the student. The program's hypothesis should be designed to lead the students to better generalizations, not to answer their questions completely. Later, students will learn to generate their own hypotheses.

Creative Students

Finally students should be allowed to pursue their own projects in a creative fashion. Once students have accepted that being creative is OK, the results of their creative endeavors should be put on display; children should explain to others the strength of their compositions. The "student art show" should be extended to the "student math show" and the "student history show" - each piece of work should be respected as something important because it is something original.

If our program of training creative students is followed, we predict that students will begin suggesting their own projects, hunting out potential problems, generating hypotheses, adopting the results, failing, and making new suggestions - all on their own. The teacher will become the acquirer of resources to support student projects, a friend to bounce ideas off, and an overall manager of student activities, trying to start students on projects in new domains to extend their creative attitude toward more and more projects.

The main point of all the exercises we have given is that creativity must be taught in manageable pieces. You can't just tell students to "be creative"; you must provide an environment where they can propose new problems, fail to solve those problems, ask questions, explore anomalies, and create explanations. And you can't tell students how to do each of these things: you must provide an environment where they can learn these skills by experience. Educational software should be used to enhance that experience. With the right environment, the attitude needed for creative work will come naturally because students will realize that their failures are simply learning experiences and are not being criticized. They will learn that the rules they are learning are really hypotheses, meant to be examined, compared, and adapted to new situations.

Computers can be a vehicle for introducing creative thinking into the schools, but first we must break out of our current model of what schools is all about. The computer revolution can be an education revolution, but only if we make it one.

REFERENCES

[BereiterEng 66] Bereiter, C. and Engelmann, S.
Teaching Disadvantaged Children in the Preschool
Prentice Hall, 1966.

[BloomBrod 50] Bloom, B.S. and Broder, L.
Problem-Solving Processes of College Students
University of Chicago Press, 1950.

[DeBono 69] DeBono, E.
The Mechanisms of Mind.
Simon and Schuster, 1969.

[Goldstein 82] Ira P. Goldstein
The Genetic Graph: a representation for the evolution of procedural knowledge
In Sleeman D. and Brown J.S., eds.,
Intelligent Tutoring Systems
Academic Press, 1982.

[Lepper 85] Lepper, M.R.
Microcomputers in Education: Motivational and Social Issues
in American Psychologist
Vol. 10, No. 1, pp. 1-18, 1985.

[Marks. 76] Marksberry, M.L.
Sizing up Assertions ,
in Elementary School Journal
Vol. 76, P. 289, 1976.

[Papert 80] Papert, S.
Mindstorms: Children, Computers, and Powerful Ideas
Basic Books, Inc., 1980.

[Ross 74] Ross, H.S.
Forms of Exploratory Behaviour in Young Children
in Foss B. ed., New Perspectives in Child Development
Penguin Education, 1974.

[SchnkAbl 77] Schank, R.C. and Abelson, R.P.
Scripts, Plans, Goals, and Understanding
Lawrence Erlbaum Associates, 1977.

[Schank 82] Schank, R.C.
Dynamic Memory: A Theory of Learning in Computers and People
Cambridge University Press, 1982.

[Schank 86] Schank, R.C.
Explanation Patterns
Lawrence Erlbaum Associates, 1986.

[Skinner 69] Skinner, B.F.
The Technology of Teaching
Prentice Hall, 1968.

[Sternberg 85] Sternberg, R.J.
Beyond IQ: A Triarchic Theory of Human Intelligence
Cambridge University Press, 1985.

[Torrance 65] Torrance, E.P.
Rewarding Creative Behavior:
Experiments in Classroom Creativity
Prentice-Hall, 1965.

[Whimbey 75] Whimbey, A. and Whimbey, L.S.
Intelligence Can Be Taught
E.P. Dutton and Co., 1975.

END

6 — 87

DTIC