A Critical Analysis of the Use of Computers in Education*

Michael J. Streibel
Educational Technology
University of Wisconsin-Madison

Microcomputers are being introduced into public and private schools at an exponential rate and this trend shows no signs of abating (Becker, 1983). It is therefore crucial that we examine the nature of computer use in education and what implications this has for the future of education (Sloan, 1984). In this article, I will present a critical analysis of the three major approaches to the use of computers in education: the drill-and-practice approach, the tutorial approach, and the simulation and programming approach. I will do this by presenting the characteristics of these approaches and then by examining their assumptions and contexts of use. The focus will not be just on the nature of instructional strategies, but on the nature of these strategies in a computer environment. The computer, as I will try to show, is not just another "delivery system" but an environment that has certain values and biases associated with it. Finally, I will try to uncover a common framework that runs throughout the three approaches and discuss the implications of this framework for education. The analysis will not, however, deal with the social, political, or economic dimensions of education even

though such dimensions are ultimately involved (Apple, 1979, 1982).

A number of philosophical questions helped guide me in the course of this analysis. For example, what is the logic behind each of the computer-based approaches to education mentioned above, and how does this logic express itself in a learning situation? These and other questions helped me understand that computerized drill-and-practice approaches to education are a behavioral form of learning technology which may not be the best way to supplement instruction. That is, the logic of computerized drill-and-practice programs may run counter to the dialectics of learning. I also asked myself how the various approaches mentioned above treated the human learner, and what consequences this could have for education. My analysis revealed that a concept of an individual as a generic information-processor was embedded in each of the three approaches. This, in turn, revealed how systems' goals (*e.g.*, efficient production of learner performance) were invariably masked behind the rhetoric of individual "mastery" and individual "needs." It also revealed that the learner's personal intellectual agency was decreased rather than increased in the case of computer-based tutorials even though the rhetoric promised the exact opposite.

Finally, I asked myself how the "intellectual tool" use of computers helped or hindered us to formulate, understand, and solve problems. That is, what kinds of "solutions" were legitimized when we formulated problems in computable form? This was part of the larger question about the potential that computer languages and simulations had for thinking and learning about problems. My answer to this question required a very careful analysis of the "expressive potential" of computer languages.

I have explored the underlying assumptions of the three most common approaches to the use of computers in education in order to understand the potentials and limitations of this technology within education. This should provide a basis for further critical study of the subject.

A paper such a
some external goal
personal values an
inextricably woven i
statement of my valu

I teach a vari
computers in educat
line of inquiry into t
teacher educator,
technology. In each
teach others how to t
to think) and how to

Empowered
liberation in so far
ability to encode and
ability to *create* me
This, in turn, presu
its realization.

My episten
critical theory which
construction of real
to be real is the
meaning-making,
making (Goody, 1
reality applies to o
and technological
social reality (i.e.,
(Kuhn, 1962; Berge
construction of rea
and dialogic engag
1982b). This col
space" in the mind
further evolution
systems. In a s

A Personal Statement of Assumptions and Values

A paper such as this is as much a journey outward toward some external goal as it is a journey inward toward one's personal values and assumptions. Since the latter are so inextricably woven into the fabric of the entire paper, a brief statement of my values is in order.[1]

I teach a variety of courses on the use of the media and computers in education and carry on an empirical and critical line of inquiry into this area. I therefore see myself as a teacher, teacher educator, and scholar in the area of educational technology. In each of these roles, I am motivated by a desire to teach others how to think about media and computers (but not what to think) and how to become empowered, life-long learners.

Empowered learning, in my view, is a form of human liberation in so far as the latter is possible. It goes beyond the ability to encode and decode abstract meanings and focuses on the ability to *create* meanings within an interpretive community. This, in turn, presumes certain epistemological commitments for its realization.

My epistemological commitments are derived from critical theory which states that knowledge is the result of a social construction of reality (Habermas, 1981). That is, what we *know* to be real is the result of historical and social processes of meaning-making, language-making, and symbol-system making (Goody, 1977; Ong, 1982). This social construction of reality applies to our knowledge of physical realty (i.e., scientific and technological knowledge) as well as to our knowledge of social reality (i.e., what we know about ourselves and others) (Kuhn, 1962; Berger & Luckmann, 1966). In both cases, the social construction of reality entails a process of human collaboration and dialogic engagement with interpretive communities (Greene, 1982b). This collaboration and engagement creates a "public space" in the minds of the people involved which then leads to the further evolution of shared meanings and shared symbol systems. In a sense, collaboration and engagement within

---

[1] I would like to thank an anonymous reviewer for suggesting that I add this section and also for making many helpful comments.

interpretive communities define our very humanity because we are participating in the creation of history, our language, and our values (Greene, 1982a).

What is the role of education within this framework? I see formal education as a social process of nurturing cognitive, affective, physical, aesthetic, and moral growth and diversity in children (Kohlberg & Mayer, 1972). The latter are dimensions of "meaning-making" and therefore can only happen within the community. The development of factual knowledge and skills is important, of course, but always in a subordinate role to personal and communal growth and development. A simple way of saying this is to state that "a teacher teaches people and not subjects"— even within the daily routine of classroom schedules, organizational structures, and curricular guidelines (Berlak & Berlak, 1981).

Learning, in my view, is an intentional and constructive act on the part of the learner. Here again, dialogue and critical thinking with the members of one's interpretive community play a central role. Hence, even though a student may be learning a simple fact or skill in a drill-and-practice activity, he or she is still using the linguistic and intellectual tools of the larger community in order to make sense out of the world. Mindfulness and meaningfulness are interconnected as are mind and the social construction of reality. Otherwise, skill-building and knowledge acquisition become mechanical and mindless processes.

Students are active participants in their own knowledge-creation in another sense. Learning is as much a process of accommodation as assimilation (Piaget, 1970; Furth, 1969). Accommodation requires the active creation of new meanings and the active construction of new symbol systems to give expression to these meanings. Assimilation is also an intentional and constructive act even though it entails a process of plugging new information into existing schemas. My reason for believing the latter derives from an ontology which views all events—both physical and social—as unique, historical and irreversible (Whitehead, 1929). A student who is learning facts or skills is more than a receiver of knowledge or a manipulator of

predefined puzzle
beings me to the h
    I believe
discrete, objectiv
describe some of
cognition entails
explicit knowlec
processes are alv
turn takes place
reality.   Since
always open to th
(Greene, 1982a).
I am always ren
century physicist:
a complete under:
    There ar
objective knowl
importance of ol
objective knowl
knowledge. Som
    1.   Tac
         kno
    2.   "An
         a v
         wo1
         exp
         am
         (e.£
    3.   Prc
         poi
         19£
         hel
         wo:
         kn
         res
         "so
         Fr

predefined puzzles. He or she is an intentional agent. This beings me to the heart of my epistemological commitments.

I believe that our knowledge encompasses more than discrete, objective, rational symbolic representations. I will describe some of these other types below. I also believe that cognition entails more than formal operations which operate on explicit knowledge structures. An individual's cognitive processes are always subordinate to meaning-making which in turn takes place within the context of the social construction of reality. Since cognition deals with meaning-making, it is always open to the future and new ways of representing meaning (Greene, 1982a). Even quasi-formal systems are never complete. I am always reminded of this when I recall how many 19th century physicists believed that Newtonian Mechanics gave them a complete understanding of the world.

There are many types of knowledge besides discrete, objective knowledge. This is not meant to diminish the importance of objective knowledge but rather to indicate that objective knowledge is interconnected with other forms of knowledge. Some other forms of knowledge are:

1. *Tacit Knowledge*—through which explicit knowledge is viewed (Polanyi, 1958, 1966).

2. *"Ambiguous" Knowledge*—which is grasped through a variety of "intelligences" (Gardner, 1983). The word ambiguous is in quotes only because what is explicit in one symbol system (*e.g.,* music) may be ambiguous and inexplicable in another system (*e.g.,* verbal).

3. *Problematic Knowledge*—which is often the starting point for new paradigms of knowledge (Greene, 1982a). Reflection, dialogue, and critical thinking help make certain routine assumptions about the world problematical. This is the case for scientific knowledge, as well as for social knowledge, and it results in "revolutionary science" in science and "social empowerment" in society (Kuhn, 1962; Freire, 1970).

4. *Experiential Knowledge*—which can be combined with rigorous reflection, analysis, and inquiry and results in robust and ecologically valid understanding (Kolb, 1984).
5. *Knowledge of Individual Differences*—which leads to the tolerance, understanding, and furtherance of individual and cultural diversity and not just to finer-grained control of the learner (Bloom, 1976).

Finally, I believe that formal education is more than a "rationally managed process." Schooling should not be modeled on an industrial assembly line because schooling deals with "meaning-making" (and ultimately with "people-making") and not with widget production (Apple, 1975). I therefore do not view a teacher as an "instructional manager" who equates certainty with predictability and power with control over process and product. Such concepts (predictability and control) are appropriate for the efficient production of identical products.

In my view, a teacher is a central agent in a dialectical community of learning and one who forms a triadic relationship with the learner and the subject matter (Greene, 1978). This maintains a learning community that extends backward and forward in time. I do not wish to ignore the daily routine of teaching. I merely wish to reiterate the craft status of teaching—a status that is particularly important today because a technological environment tends to leave most people "powerless" and unable to articulate their state of being (Greene, 1982a; Ellul, 1980a). Learning, empowerment, and community, in my view, all go together.

A teacher is also someone who nurtures various types of intelligence in students. A teacher is not someone who stands over and above the learning process or who separates the conception from the execution of learning (Apple, 1982). In short, a teacher is not someone who adopts a theoretical stance toward the learner. Rather, a teacher is a life-long learner who engages and guides others along a similar path.

One can see from the foregoing discussion that my deepest commitments center on valuing individual and cultural uniqueness and diversity while at the same time valuing the

knowledge and skills th
reality. Maintaining
uniformity (of person,
struggle for a teacher
computer (Berlak & Ber
and values that I now
computers in education

Drill and Pr

Drill-and-pract
programs that guid
instructional strategy.
education today (Suppe
Watt & Weiner, 1982)
time-sharing, and mi
Alderman, 1978; Stewa
describing the major
approach and then by
latter issues, however,

The most gene
courseware programs
learning. I will de
courseware programs
the mastery-learning
learning, and the con
will then show how
dimensional form of
structures, reward
educational events to
This, in turn, underm
performances into oth

Finally, I wi
practice courseware p
legitimizes behavio
educational goals.
cultures must still
interpersonal interac
alternate cultures (a
goals). Drill-and-pra

h can be combined
is, and inquiry and
cologically valid

rences—which leads
, and furtherance of
ity and not just to
ner (Bloom, 1976).
tion is more than a
ould not be modeled
chooling deals with
eople-making") and
erefore do not view a
quates certainty with
process and product.
e appropriate for the

gent in a dialectical
triadic relationship
Greene, 1978). This
ends backward and
the daily routine of
status of teaching—a
ecause a technological
erless" and unable to
982a; Ellul, 1980a).
in my view, all go

ures various types of
someone who stands
who separates the
pple, 1982). In short,
ical stance toward the
ner who engages and

ssion that my deepest
idual and cultural
me time valuing the

knowledge and skills that contribute to the social construction of reality. Maintaining a balance between uniqueness and uniformity (of person, culture, knowledge, and skills) is a daily struggle for a teacher that will be profoundly affected by the computer (Berlak & Berlak, 1981). It is with these personal beliefs and values that I now turn to a critical analysis of the use of computers in education.

Drill and Practice Computer Programs

Drill-and-practice courseware programs (*i.e.*, computer programs that guide learning with a drill-and-practice instructional strategy) are the dominant use of computers in education today (Suppes, 1966; Coburn, Kelman, Roberts, Snyder, Watt & Weiner, 1982). They are currently run on mainframe, time-sharing, and microcomputers (Murphey & Appel, 1977; Alderman, 1978; Stewart, 1979). I will deal with this topic by first describing the major characteristics of the drill-and-practice approach and then by discussing a number of related issues. The latter issues, however, warrant some preliminary remarks.

The most general issue deals with how drill-and-practice courseware programs are biased toward behavioral forms of learning. I will demonstrate this by showing how such courseware programs combine several broad traditions such as the mastery-learning paradigm, the philosophy of individualized learning, and the concepts of educational work and efficiency. I will then show how such programs represent a very one-dimensional form of education because they restrict the goal structures, reward structures, and meaning structures of educational events to the domain of educational productivity. This, in turn, undermines the possibility of integrating sub-skill performances into other educational experiences.

Finally, I will deal with the issue of how drill-and-practice courseware programs work best in a learning culture that legitimizes behavioral performances over other types of educational goals. Ironically, behaviorally oriented learning cultures must still be created, mediated, and sustained by interpersonal interactions which have the potential for forming alternate cultures (and thereby alternate types of educational goals). Drill-and-practice courseware programs, however, do not

permit these alternate goals to develop. My discussion of the importance of dialogue will then lead to an analysis of the "dialogue" component of human-computer tutorial interactions.

Description of Drill-and-Practice Courseware

Drill-and-practice courseware programs make a number of assumptions about instruction (Bunderson, 1981; Salisbury, 1984):

a. Previous instruction in the concept or skill has already taken place,

b. Regular instruction is only being supplemented and not replaced,

c. Instruction is to follow a controlled, step-by-step linear sequence of sub-skills according to an algorithm embedded in the computer program. This algorithm does *not* constitute a model of a student or an expert but constitutes a model of: i) rote skill-building in the case of drill, and ii) patterned skill-building according to the logic of the content and an instructional theory in the case of practice (Skinner, 1968; Gagné, Wager & Rojas, 1981),

d. There exists a right/wrong answer dichotomy in the logic of the content,

e. The basic unit of instructional interaction is a question-answer-branch episode (Dennis, 1979a). Continuous learner responses in the form of correct answers are therefore expected, and

f. The best feedback by the program from an instructional point of view is an immediate check on a student's responses according to the logic of the content:

  i. Positive feedback when the answer is correct,

  ii. corrective (rather than judgmental) feedback when the answer is incorrect.

The characteristics described above make several things clear: drill-and-practice courseware programs are designed to provide immediate corrective interventions in the learning process when continuously monitored performance measures indicate incorrect responses. The learner is viewed as a "black

box" and his or her mechanical process (i. feedback mechanisms behavioral goal). therefore constitute a (Skinner, 1968). Th building but not for h 1984). Within the l drill-and-practice pro gains (Kulik, Kulik, &

The character thing clear: drill-an dimensional form of and take us into broad

a. They c *structur*

  i.

  ii.

  iii.

  iv.

b. They c *structul*

  i.

  ii.

c. They *structu*

My discussion of the
to an analysis of the
tutorial interactions.

ourseware

ograms make a number
derson, 1981; Salisbury,

e concept or skill has

being supplemented and

controlled, step-by-step
kills according to an
computer program. This
e a model of a student or
model of: i) rote skill-
, and ii) patterned skill-
ic of the content and an
ase of practice (Skinner,
s, 1981),
answer dichotomy in the

tional interaction is a
isode (Dennis, 1979a).
es in the form of correct
ed, and
he program from an
an immediate check on
ding to the logic of the

en the answer is correct,
than judgmental)
nswer is incorrect.
ve make several things
ograms are designed to
tions in the learning
performance measures
r is viewed as a "black

box" and his or her behaviors are shaped by an external, mechanical process (i.e., by an instructional algorithm that uses feedback mechanisms to guide the learner towards a pre-specified behavioral goal). Drill-and-practice courseware programs therefore constitute a deterministic form of behavioral technology (Skinner, 1968). This may be adequate for beginning skill-building but not for higher levels of learning (Dreyfus & Dreyfus, 1984). Within the beginning skill-building domain, however, drill-and-practice programs do result in significant performance gains (Kulik, Kulik, & Cohen, 1980).

The characteristics described above also make another thing clear: drill-and-practice courseware programs are a one-dimensional form of education. The reasons for this are simple and take us into broader issues:

a. They contain a uni-dimensionality of *goal structures* since:

    i. They only focus on *pre-specified* behavioral performances,

    ii. They are designed for *convergent uniformity of outcomes* and not a multiplicity of divergent outcomes,

    iii. They exclude non-behavioral educational goals (*e.g.*, emergent attitudinal and cognitive strategy outcomes),

    iv. They are conceptualized and designed before any interaction with the actual learners takes place,

b. They contain a uni-dimensionality of *reward structures* because:

    i. They only reward successful performance on one sub-skill with an opportunity to work on the next sub-skill,

    ii. They define "individualized" feedback in generic terms rather than in terms of personal, semantic, and affective engagement,

c. They contain a uni-dimensionality of *meaning structures* because:

i. They only define "mastery" (i.e., success at any point) in terms of objective performance (*e.g.,* total number of correct responses),

ii. They do not acknowledge the legitimacy of qualitative aspects of performance (*e.g.,* expert performance) except in so far as it results in better quantitative performance.

In order to understand the consequences of this one-dimensionality for education, I will delve into the philosophy behind drill-and-practice courseware programs as well as into the context of their use. I will do this by examining the mastery learning paradigm, the philosophy of individualized learning, and the concepts of educational work and efficiency.

Mastery Learning and Drill-and-Practice Courseware

The mastery learning paradigm assumes that most students can learn most things to a specific level of competence in varying amounts of time (Bloom, 1976). It therefore bases differences in the *amount* of learner performances at any point in time on differences in the *rates of learning.* Second, the mastery learning paradigm assumes that instruction can be consciously designed to guarantee specific outcomes. It therefore places a heavy emphasis on the *quality of instructional materials.* Finally, the mastery learning approach uses criterion-referenced tests with each objective to decide whether a student has met the criterion of success. Students are then permitted to go on to the next set of objectives.

You can see from this description that the mastery learning paradigm closely resembles a rationally managed input/output model of educational performance (Apple, 1979, 1982). It therefore makes a number of assumptions about the pedagogical principles, classroom practices, and instructional arrangements involved (Barr & Dreeben, 1978; Bolvin, 1982). These assumptions elaborate the input/output model.

a. *Pedagogical principles:*

The mastery learning paradigm assumes that students vary in their aptitude, ability to understand instruction, motivation, and

(i.e., success
of objective
ber of correct

e legitimacy of
rmance (*e.g.,*
in so far as it
performance.
s of this one-
he philosophy
vell as into the
g the mastery
lized learning,
y.

Courseware
es that most
competence in
herefore bases
at any point in
d, the mastery
be consciously
refore places a
al materials.
rion-referenced
ent has met the
to go on to the

t the mastery
ally managed
le, 1979, 1982).
the pedagogical
arrangements
1982). These

digm assumes
ude, ability to
ivation, and

perseverance.   These factors contribute to the
distribution of performances by students at any
given point in time. Furthermore, these factors are
only considered in so far as they affect educational
performance. They are not, for example, considered
in terms of their contribution to non-performance
educational goals (*e.g.,* growth in consciousness,
growth in aesthetic appreciation, moral growth,
etc.).

Although all students are believed to be able
to achieve mastery *given enough time,* the amount of
time it takes to achieve mastery is normally
distributed. In comparison, in a norm-referenced
learning paradigm, the actual performance scores
are also distributed normally at any particular point
in time, but there is no claim that low-achievement
students will *ever* reach higher levels of
performance (Bloom, 1971, 1976).

The pedagogical assumption that all students
are able to achieve mastery given enough time
allows learning to be treated as a rationally
managed process since only time and resources
have to be considered in order to guarantee a
predictable performance outcome. Considerations
such as the dialectics of learning, accommodation to
individual uniqueness, and the possibility of
emergent goals have been factored out of the process.
I will show later that these considerations are
essential for learning—even at the level of simple
skill-building.

b.   *Classroom and school practices:*

The   mastery   learning   paradigm
manipulates the *time* allowed for learning and the
quality of instructional stimuli as the main factors
to help students achieve mastery. It therefore entails
the rational planning of classroom time, schedules,
organization, and conditions of instruction, as well
as the rational design of instructional materials

(Nunan, 1983). Such design and planning activities are only called rational to the extent that they are guided by the pragmatics of instructional and organizational theories. Such design and planning are *not* guided by the pragmatics of classroom teaching (Wolcott, 1977). For example, predictability and manageability of process and product are a prime consideration and not whether some unique classroom event becomes an occasion for further learning. Hence, the conception of instructional events is separated from the execution of such events, and the conceptual phase directs and controls the execution phase (Apple, 1982). In this scheme of things, we would eventually expect teacher performance (and ultimately administrator and even system performance) to be evaluated in terms of student performance since correct student behavior is the ultimate output. However, I will show later that this in fact serves the system's needs more than the learner's needs.

c. *Instructional arrangements:*

The mastery learning paradigm follows a number of procedures to guarantee that students will perform at pre-specified levels (Bloom, 1976):

i. Pre-instructional assessment procedures are used to measure the presence or absence of prerequisite knowledge in the learner,

ii. Initial teaching methods are used to inform the learner of the objectives and prerequisite knowledge,

iii. Training procedures are used to help the learner acquire the appropriate knowledge and skills,

iv. Continual assessment procedures are used to assure the presence of sub-skills,

v. Immediate remediation procedures are used if the sub-skills are not present, and

vi.

You can s
learning paradig
quality control te
procedure and all
that is left for hun

The mast
provides a broa
courseware. It s
(*i.e.*, measurable
be achieved (*i.*
instructional sti
as the trainin
framework. Al
mastery paradi
individualized.
individualizatio

Individualiza

Individu
study, individu
educational outc
recognition that
Messick, 1976; (
forms of indivi
outcomes for ge
unique individu
The phil
within individu
specific assump

a. A
ch
th
pa

b. A
se

vi.   Certification of mastery is added when some predetermined criterion performance is reached by the learner.

You can see from all of these techniques that the mastery learning paradigm conceptualizes the instructional process in quality control terms. Each step in the paradigm is expressed as a procedure and all instruction is arranged to maximize output. All that is left for humans to do is to get to work!

The mastery learning paradigm described above therefore provides a broad theoretical framework for drill-and-practice courseware. It specifies *what kinds* of things are to be achieved (*i.e.*, measurable performance gains) and *how* these things are to be achieved (*i.e.*, through the manipulation of time and instructional stimuli). Drill-and-practice courseware functions as the training and remediation component within this framework. Although group work is permissible within the mastery paradigm, drill-and-practice courseware is usually individualized. This leads me to the philosophy of individualization.

Individualization and Drill-and-Practice Courseware

Individualization can mean many things: independent study, individual pacing, individual diagnosis, individual educational outcomes, etc. (Bolvin, 1982). It arises out of the larger recognition that individuals differ from each other (Sperry, 1972; Messick, 1976; Cronbach & Snow, 1977). Within computer-based forms of individualized learning, however, it refers to generic outcomes for generic individuals rather than to personal goals for unique individuals. More of this later.

The philosophy of individualization, as it has developed within individualized systems of education, contains a number of specific assumptions (Lukes, 1973; Talmadge, 1975):

a.   A belief that each person has a unique set of characteristics or aptitudes that ultimately influence the *rate* at which competent performance in a particular skill is achieved,

b.   A belief that a well-defined and well-structured sequence of instructional events can be designed for

each individual to facilitate his or her progress toward preplanned outcomes,

c. A belief that time and quality of instructional materials will influence successful completion of an objective,

d. A belief that assessment procedures of student "needs" and characteristics will indicate a readiness for the objectives by the learner,

e. A belief that an evaluation mechanism can be found for constantly monitoring the student's progress toward a pre-planned outcome (these same evaluation mechanisms also provide data for the instructional system's performance), and

f. A shift in the role of the teacher *from* a pedagogical one *toward* that of instructional decision-maker (i.e., placement of students; selection, use and allocation of space, time, and materials; data collection and report-writing, etc.)

These assumptions are implemented in a number of ways. In some systems of individualized learning such as the Keller Plan, students have a great amount of flexibility in setting schedules, getting help from student tutors, and following any number of paths toward a predefined goal (Keller, 1968). Each of these factors is adjusted for the sake of individual rates of learning. In the Skinnerian version of the philosophy of individualization, on the other hand, instruction is broken down into much smaller units, and instructional events are more controlled and automated (Skinner, 1968). It is this latter version that is most often implemented in drill-and-practice courseware.

Drill-and-practice courseware programs overwhelmingly use rate of progress as their major dimension of individualization although they sometimes also include level of difficulty (Suppes, 1966). Other dimensions such as cognitive style are often called for but rarely implemented because of the difficulty of specifying these factors in computable form (Scriven, 1975).

Drill-and-practice courseware programs also break the instructional process into very small steps. They then assess each

response by the learner
learner to follow. The
narrow sense of the t
meaning of individual:
measurable and comp
progress along finite
specified, measurable
*presentation of inform
with that information,*
during the time that th

Finally, drill-a
the teacher to a man:
exception handler) (Bo:
a teacher or school sys
classroom accordin
individualization, o
programs are biased
schools, the shift tov
place without the comp

In many wa
courseware is consis
field toward the "tec
Franklin, 1974, 1982
under the rubric of tl
rest of my discussion.
help us begin to und
courseware.

Courseware as Re

Victor Bunde
analyses of the co
(Bunderson, 1981).
analysis-and perhap:
he argues, has botl
product, courseware
instruction] that op:
instructional deliv
constitutes (Bunder:

r her progress

f instructional
 completion of

res of student
l indicate a
rner,
m can be found
lent's progress
(these same
e data for the
and
a pedagogical
lecision-maker
tion, use and
aterials; data

umber of ways.
 as the Keller
lity in setting
following any
1968). Each of
vidual rates of
philosophy of
is broken down
ents are more
s latter version
ce courseware.
verwhelmingly
dimension of
nclude level of
h as cognitive
because of the
form (Scriven,

also break the
hen assess each

response by the learner and specify a finite number of paths for the learner to follow. They are therefore "individualized" in a very narrow sense of the term. That is, they have restricted the meaning of individualization to a finite set of choices within a measurable and computable domain (i.e., individual rates of progress along finite, forced-choice paths that lead to pre-specified, measurable outcomes). Since they control both the *presentation of information* as well as the learner's *interactions with that information,* they control the individual's total attention during the time that they are used.

Finally, drill-and-practice courseware programs relegate the teacher to a managerial function (*e.g.,* resource manager or exception handler) (Boyd, 1983a, 1983b). This is not to suggest that a teacher or school system is forced by the computer to organize the classroom according to the Skinnerian philosophy of individualization, only that drill-and-practice courseware programs are biased towards such an orientation. In many schools, the shift toward teacher-as-manager has already taken place without the computer (Apple, 1975; Berliner, 1982).

In many ways, the philosophy of drill-and-practice courseware is consistent with the movement in the curriculum field toward the "technical control" of learning (Kliebard, 1971; Franklin, 1974, 1982; Carlson, 1982). I will describe this notion under the rubric of the "technological framework" through-out the rest of my discussion. For now, the concept of replicated work will help us begin to understand the concept of "technical control" in courseware.

Courseware as Replicated Work

Victor Bunderson has developed one of the most thorough analyses of the concept of courseware as replicated work (Bunderson, 1981). The following discussion owes much to his analysis and perhaps will add to his ideas. Computer courseware, he argues, has both a product and a process dimension. As a *product,* courseware consists of the "consumables [materials of instruction] that operate on and with a technologically mediated instructional delivery system." As a *process,* courseware constitutes (Bunderson, 1981):

an economically replicable and easily portable package which when used in combination with a technologically mediated instructional delivery system, is capable of performing work related to training and performance improvement.

An instructional delivery system, on the other hand, consists of both *the physical objects and structures* "designed to perform or facilitate the work necessary to achieve educational and training goals," and, a *human culture* of "traditions, values, and habits that inform and constrain the use of the physical artifacts" (Bunderson, 1981). The instructional delivery system is therefore the broadest category within which all other components of the ensemble are situated and from which they derive their meaning.

We can see from the description above that the concept of computer courseware contains a "technological control" orientation. That is, the technical structure of the delivery system shapes the form and function of the human culture and the physical artifacts. The technical structure also orients these components toward some external goal (i.e., educational performance) and then tries to maximize the levels of this goal. A technological delivery system will therefore ultimately influence the nature of the classroom culture—unless, of course, the classroom is already organized as a work culture.

From my earlier discussion, we can begin to see that the student's control over the pace of learning is really a form of pseudo-control because he or she can only choose from a finite number of paths toward a pre-determined goal. Bunderson acknowledges this somewhat by saying that "learner-centered will emphasize learner productivity, not necessarily learner-control" (Bunderson, 1981). This restricts the meaning of "individualized learning" to that of "individualized productivity level."

Bunderson continues his discussion of educational work by criticizing the inability of current teacher-centered "delivery systems" to be more "productive." The teacher-centered culture, he argues, has reached the "limits of [its] improvability." His solution is a technological one: "when education is analyzed into

the work that is requir
make a fundament
Bunderson's argument
based technology (*i.e.,*
than a teacher without
very carefully, however
technologies.

Before turning
are in order. First o
"instructional delive
instructional setting i
respected and unique i
can happen in such a s
place for training and
mastery learning pa
workplace for both te
classroom as a workpl
the work potential o
discussion by Bunder:
systems lays the conc
courseware as more ef

Another gener
as "productivity tools"
also apply to computer

a.   What re:
     product:
     intellige
     the pres
     intellige:
     is the k
     Hence,
     better of
     fact, be
     assume

b.   The fac
     more th:
     the "w
     determi

sily portable
ation with a
nal delivery
·k related to

n the other hand,
ctures "designed to
achieve educational
"traditions, values,
use of the physical
nal delivery system
a which all other
d from which they

that the concept of
nological control"
the delivery system
an culture and the
 also orients these
 (i.e., educational
evels of this goal. A
ltimately influence
ss, of course, the
ture.

egin to see that the
is really a form of
choose from a finite
d goal. Bunderson
at "learner-centered
aecessarily learner-
ts the meaning of
ualized productivity

of educational work
r-centered "delivery
-centered culture, he
mprovability." His
tion is analyzed into

the work that is required, technology is seen as the *only* way to make a fundamental difference" (Bunderson, 1981). Bunderson's argument is very general and even applies to book-based technology (*i.e.,* a teacher with books can accomplish more than a teacher without books). His argument has to be analyzed very carefully, however, for book-based as well as computer-based technologies.

Before turning to my analysis, some general comments are in order. First of all, conceiving of the classroom as an "instructional delivery system" (rather than, say, an instructional setting for the dialectical encounter of mutually respected and unique individuals) narrows the debate about what can happen in such a setting. The classroom, in effect, becomes a place for training and development. I mentioned earlier that the mastery learning paradigm turned the classroom into a workplace for both teachers and students. The concept of the classroom as a workplace is further compounded by highlighting the work potential of classroom technologies. Second, the discussion by Bunderson about classroom instructional delivery systems lays the conceptual groundwork for accepting computer courseware as more efficient versions of the same thing.

Another general theme deals with books. Looking at books as "productivity tools" has a number of problems—some of which also apply to computers:

a.   What really counts in books (even for the restricted productivity paradigm of education) is the *intelligence* embodied in the print medium and *not* the presence or quantity of print information. This intelligence goes beyond procedural knowledge and is the key to what is transmitted during learning. Hence, teachers with unintelligent books are not better off than teachers without books. They may, in fact, be worse off because they might be tempted to assume that they can accomplish more than before.

b.   The fact that teachers with books can accomplish more than teachers without books does *not* mean that the "work potential" of book-based technology *determines* what a teacher does. True, reading

skills might be required simply because books are used, but this does not mean that books need only be used for training. Conceptualizing books in terms of their work potential, however, forces them into a productivity scheme.

c. Analyzing books in terms of their work potential diverts attention away from their dialectical potential (*i.e.*, their ability to confront one with alternate points of view from one's taken-for-granted reality). Books, for example, also have the ability to reveal how other humans have integrated the dialectical tension between, say, justice and love or between wisdom and courage. In these instances, books become guides for experiential learning that point the way to personal and communal integration. All of this is hidden when we look at books in terms of their work potential. We are then only left with an image of a book as a training aid.

d. Finally, books as such leave open the choice of interactions which a person can carry out. Hence, books leave open the way an individual uses and makes sense out of the information. They are therefore not a good analogy with computer courseware programs which control both the presentation of information as well as the user's interactions with that information.

Bunderson's argument permits one to compare teacher-delivered instruction and technology-delivered instruction because both are conceptualized in similar terms. Computer courseware is then seen as a more efficient mechanism. I must admit that when I observe certain classrooms which contain hierarchical authority, rigid schedules, and mindless workbooks, I am inclined to agree with Bunderson. But I must also point out that even in such classrooms, students still have some opportunity for personal integration of experiences and skills. That is, students can still integrate their drill-and-practice activities with exploration, planning, negotiation, and collaboration—even if it is done as a subterfuge. These latter characteristics are essential

for education (Kolb, :
courseware, on the c
attention, and interac
integrations are no lo
this loss worth the p
only the acquisition o

When we e
Bunderson's concept
embody the *extensioı*
and procedural).   I
teacher: presents in:
students, provides sı
"individual needs"
motivate learning, t
system, assesses stı
information, and ma
is there a teacher's
students beyond m
Bunderson describε
processing functionε
functions *can* be caı
technology (*e.g.*, vi
means maximizing
financial cost. Eff
process and the pı
effectiveness, in effe
criteria of excellenc
area but to the quaı
No wonder teachers

There is a
point of view, howe
individual, but the
systems concerns.]
from the intension
toward the exteı
information-procesı
emphasized:

ply because books are
hat books need only be
alizing books in terms
ver, forces them into a

f their work potential
om their dialectical
 to confront one with
from one's taken-for-
example, also have the
umans have integrated
n, say, justice and love
ge. In these instances,
eriential learning that
nal and communal
idden when we look at
potential. We are then
ook as a training aid.
ve open the choice of
can carry out. Hence,
n individual uses and
nformation. They are
alogy with computer
ich control both the
 as well as the user's
ation.
e to compare teacher-
-delivered instruction
ilar terms. Computer
nt mechanism. I must
ssrooms which contain
d mindless workbooks,
t I must also point out
l have some opportunity
s and skills. That is,
-practice activities with
ollaboration—even if it
racteristics are essential

for education (Kolb, 1984). In individualized drill-and-practice courseware, on the other hand, where a student's total time, attention, and interactions are controlled by the computer, such integrations are no longer possible (Dreyfus & Dreyfus, 1984). Is this loss worth the price—even for low levels of learning where only the acquisition of procedural rules is involved?

When we examine the actual characteristics of Bunderson's concept of educational work, we find that they all embody the *extensional* side of education (i.e., the measurable and procedural). For example, according to Bunderson, a teacher: presents information to students, models processes for students, provides students with trials and feedback, discusses "individual needs" with students, uses affective appeals to motivate learning, trains students in the use of the delivery system, assesses student performance, manages the assessed information, and manages classroom interactions. Where, here, is there a teacher's affective and semantic engagement with students beyond maximizing performance gains? All that Bunderson describes are procedural skills and information processing functions. When conceptualized in this way, such functions *can* be carried out more efficiently and effectively by technology (*e.g.*, video, microcomputer, etc.). Efficiency here means maximizing educational productivity at the lowest financial cost. Effectiveness means reliably reproducing the process and the product (Bunderson, 1981). Efficiency and effectiveness, in effect, are no longer subject to the qualitative criteria of excellence and expertise within a particular subject area but to the quantitative criteria of economics (Eisner, 1979). No wonder teachers cannot compete!

There is a fundamental contradiction in Bunderson's point of view, however. The rhetoric stresses the "needs" of the individual, but the terms of the debate emphasize instructional systems concerns. By shifting the educational interactions away from the intensional logic of interpersonal interactions and toward the extensional logic of procedural skills and information-processing functions, the following criteria are emphasized:

a. *Systems efficiency* (*i.e.,* maximizing the throughput of students for the time and resources invested—rather than developing individual talents),

b. *Systems reliability* (*i.e.,* quality control and replicability of input—rather than the establishment of individual, communal, or cultural diversity), and,

c. *Systems economy* (*i.e.,* more scholar for the dollar—rather than personally-determined pursuit of excellence).

Hence, *only those "individual needs" amenable to systems logic are served.*

There is another contradiction in Bunderson's concept of educational work. The very work culture which has to exist in a classroom for a technological instructional system to operate can only result from intensional human engagement, negotiation, and interaction. Students and teachers are therefore essential and continuous agents in the creation of a classroom work culture (Sarason, 1982). However, the very processes that are required to *produce* the work culture then have to be *denied* because they contradict the technological framework. This is the case because an instructional delivery system embodies a technological culture that tries to shape the human culture to its own ends whereas human cultures shape their own ends (Nunan, 1983).

A good illustration of the contradiction described above can be seen in the teacher roles which Bunderson believes will predominate in a technological environment: corrector of imperfect and outdated information, illustrator and augmenter of delivered instruction, illustrator and augmenter of the expert algorithms embodied in the instructional system, creator of a technologically acceptable setting, and interpreter of automatically tested and recorded results. Each of these roles shows how the technological delivery system has become the central organizing factor in classroom life. The classroom has thereby been structured as a workplace by someone other than the teacher (Wolcott, 1977; Nunan, 1983). A simple drill-and-practice courseware program is therefore not all that innocent an aid to teaching in a classroom community (Benne, 1975). In fact, such

programs ma
because teac
affair.[2]

Summary o

Drill-
deterministic
embody narr
philosophy o
work and ef
into a form
and they res
and level of
productivity
    Com
may also no;
they maxir
simple: dr:
of interact
domain an
skills.

    Fin
of a beha
behavioral
thinking
arises whe
whether t
philosoph:
argued th
limitatio:
1981).

_____
[2]Some sch
our school
coursewar
population
1979; Ok
evidence {

The left column of this page shows fragments of running text with the left portion cut off:

ng the throughput
ources invested—
talents),
lity control and
the establishment
ltural diversity),

ar for the dollar—
ined pursuit of

ds" amenable to

lerson's concept of
h has to exist in a
em to operate can
ment, negotiation,
herefore essential
room work culture
hat are required to
lied because they
is the case because
s a technological
re to its own ends
(Nunan, 1983).
n described above
erson believes will
ent: corrector of
r and augmenter of
nter of the expert
stem, creator of a
d interpreter of
Each of these roles
m has become the
The classroom has
eone other than the
le drill-and-practice
innocent an aid to
975). In fact, such

programs may ultimately conflict with the nature of teaching because teaching is *not* a highly rational decision-making affair.[2]

Summary of Drill-and-Practice Courseware

Drill-and-practice courseware programs embody a deterministic form of behavioral learning technology. They also embody narrow aspects of the mastery-learning paradigm, the philosophy of individualization, and the concepts of educational work and efficiency. That is, they convert the learning process into a form of work that tries to maximize performance-gains, and they restrict the meaning of individualism to rate of progress and level of difficulty (and ultimately to individualized levels of productivity).

Computerized drill-and-practice courseware programs may also *not* be the best way to supplement instruction even though they maximize sub-skill performance. The reason for this is simple: drill-and-practice courseware programs restrict the type of interactions involved to a decontextualized performance domain and diminish integration of sub-skills with higher-level skills.

Finally, drill-and-practice courseware programs are part of a behavioral learning culture that mitigates against non-behavioral goals. Hence, such programs do not lead to critical thinking or personal empowerment. The question therefore arises whether computer-based tutorials have such a potential or whether they simply develop the behaviorally oriented learning philosophy in a more sophisticated way. Many authors have argued that computer-based tutorials do in fact solve some of the limitations of drill-and-practice programs (O'Neil & Paris, 1981).

---

[2]Some scholars have suggested that the prevalence of drill-and-practice in our schools is not incidental. The hidden curriculum of drill-and-practice courseware, they claim, prepares a certain segment of the student population for similar types of computer-controlled jobs in society (Apple, 1979; Olds, 1982). However, such an argument involves socio-political evidence and is beyond the scope of this paper.

Tutorial Computer Programs

Do tutorial courseware programs go beyond drill-and-practice approaches to education? They do in an obvious sense because they are intentionally *designed* to "take total responsibility for instruction" and to contain a "mixed-initiative dialogue" (Dennis, 1979b; Bork, 1980a; O'Shea & Self, 1983). But what exactly do the terms "dialogue" and "initiative" mean here? I will examine the nature of "dialogue" in human-computer interactions as a way of analyzing the nature of tutorial programs. My reason for this is simple: dialogue is seen by many authors to be the basic building block for higher levels of learning (Freire, 1973; Greene, 1978; Shor, 1980). I will also examine the *types* of "quality-control" procedures which are used and the *nature* of the tutorial engagement. I will then show that many of the themes which I uncovered in my analysis of computerized drill-and-practice re-emerge in computerized tutorials in a more sophisticated form. This will stand in sharp contrast to the rhetoric about tutorial courseware programs which claims that such tutorials resemble real conversations and real teaching (Bork & Franklin, 1979; Dennis, 1979b; Bork, 1980b).

Before dealing with these larger issues, however, I would like to describe the various types of human-computer interactions in tutorial courseware programs: on-line tests, remedial dialogues, and interactive proofs (Bork, 1980b).

On-Line tests are initiated by a computer as part of the tutorial interaction. They involve a comparison between a model of the student (which was either pre-programmed into the computer or constructed by the computer on the basis of student performances) and a model of an expert. In simpler tutorials, on-line tests only involve a comparison between student performances and pre-specified, content-determined performance levels (O'Shea & Self, 1983). In either case, however, on-line tests provide continual diagnoses of students' performances.

An immediate consequence of having on-line tests in computerized tutorials is that the learner is subject to constant "quality control." This does not seem unreasonable since, in interpersonal interactions, humans also check out their

inferences abou
a computer do
such monitori
engagement a
interactions, or
external ager
programmer) i:
interaction. T
negotiable, ar
learner. Cons
understand tl
interpersonal
behavioral out
rather obvious
is not always s
The col
in on-line tests

    a. I:
      c
      n
      t
      s
      i:
      e
      u
    b. I
      a
      i
      c
      ŗ
      c
      ɪ
      c
      t
      (
      (

      ]

grams

go beyond drill-and-
o in an obvious sense
rned to "take total
in a "mixed-initiative
ea & Self, 1983). But
initiative" mean here?
in human-computer
e nature of tutorial
dialogue is seen by
k for higher levels of
r, 1980). I will also
dures which are used
I will then show that
d in my analysis of
ge in computerized
is will stand in sharp
eware programs which
nversations and real
79b; Bork, 1980b).
ues, however, I would
computer interactions
line tests, remedial
b).
mputer as part of the
ison between a model
med into the computer
e basis of student
simpler tutorials, on-
between student
ontent-determined
either case, however,
noses of students'

ring on-line tests in
subject to constant
reasonable since, in
o check out their

inferences about each other (Nisbett & Ross, 1980). Why shouldn't a computer do the same? However, in interpersonal dialogues, such monitoring takes place in the context of semantic engagement and conjoint intentions. In human-computer interactions, on the other hand, such monitoring is guided by an external agent's (i.e., author, instructional designer, or programmer) intentions which are fixed *for the duration of the interaction.* These external intentions establish pre-set, non-negotiable, and measurable performance outcomes *for the learner.* Constant monitoring is therefore not intended to understand the learner and his or her messages (as in interpersonal dialogue) but rather intended to guarantee a behavioral outcome. In drill-and-practice courseware, this was rather obvious. In "mixed-initiative" computer "dialogues," this is not always so evident.

The constancy and immediacy of diagnosis and feedback in on-line tests has several other consequences:

a. It emphasizes accretion learning because the computer is constantly assessing evidence of normal progress toward a pre-specified goal. This tends to discourage "messing around" with the subject matter because "messing around behavior" is not evidence that a learner is building up an experiential basis for a quantum leap of understanding (Hawkins, 1974).

b. It tends to focus learning on generic *means* as well as generic *ends* in spite of the fact that tutorials are individualized. The reason for this is simple: by continually measuring and diagnosing educational performances with context-free response analysis, on-line tests pre-empt personally constructed means. The computer, in effect, *controls the means as well as the ends* and constitutes a powerful "other" that structures and dominates the entire interaction (Weizenbaum, 1976; Scheibe & Erwin, 1979; Gardner, 1979; Turkle, 1984).

In interpersonal interactions, on the other hand, learning tends to focus on the ends-in-view

and not on the means (Polanyi, 1958; Greene, 1978). Hence, learning incorporates personally constructed means and meanings. This is not to suggest that human teachers cannot dominate an interaction with a learner but only that learners have the opportunity to develop personal ways to reach a particular goal.

c.   It tends to accelerate the learning process because it creates a set of temporal expectations (notice that rate of learning was the major dimension of individuality, and faster rates were considered better because of the efficiency orientation of the system. This in turn, biases the tutorial interaction *against* reflectiveness and critical thinking (Freire, 1973; Shor, 1980). Some courseware authors have suggested that this bias can be countered by using "individualized" wait loops in the courseware programs (Schneidermann, 1980). However, reflectiveness is *not* a matter of waiting longer.

Remedial dialogues are initiated by the computer when the learner's performance does not match some pre-specified performance criteria (Bork, 1980b). They assume that the student already knows the area and can work with the information which is presented. Here again, this parallels what happens in interpersonal dialogues but with some very important distinctions:

a.   In interpersonal tutorials, remedial dialogues are initiated by a teacher on the basis of his or her tacit knowledge about the unique characteristics of the learner. Furthermore, the teacher tries to understand the learner's state of mind by "thinking like the student" in order to unravel the student's conceptual bind or misunderstanding (Hawkins, 1974). This is a unique, constructive, and intentional act of empathy and engagement by the teacher and *only nominally entails the student's behaviors.*

b.   In I
are
com
algc
Self
effe
eng
The
"di£
gen
wh
bas
res
ori
the
typ
bel
Interactii
permit the learn
choices (Bork, 1
work through a
even construct t
nature of the co
the debate. Th
from mathemat
parallels the 1
computable for
author (or inst
interactive pro
mathematical s
interactive pro
can be recast ir
that relates an
"proof" then pr
ultimately tre
computerized
mathematics.

digms Regained

; Greene, 1978).
es   personally
This is not to
ot dominate an
ly that learners
ersonal ways to

rocess because it
(notice that rate
dimension   of
were considered
rientation of the
torial interaction
itical   thinking
rseware authors
be countered by
the   courseware
0).   However,
iting longer.
mputer when the
e   pre-specified
that the student
nformation which
hat   happens   in
very   important

ial dialogues are
of his or her tacit
acteristics   of   the
eacher   tries   to
nind   by "thinking
vel   the   student's
nding   (Hawkins,
nstructive,   and
gagement   by   the
ils   the   student's

b.   In human-computer tutorials, remedial "dialogues" are initiated on the basis of a set of explicit and computable rules (expert or content-related algorithms) (Sleeman & Brown, 1982; O'Shea & Self, 1983). The tutorial courseware program, in effect, constitutes a generic, rule-driven process that engages an internal, *generic* model of the learner. The actual human agent (i.e., student) in this "dialogue" only provides the data for the computer's generic model of the learner. This is even the case when the computer builds up a model of the student based on a history of student performances. The resulting model of the student is still a formal, rule-oriented, generic model. Remedial "dialogues" therefore do not involve *this* student but rather this *type* of student. I will explain this in greater detail below.

Interactive proofs are a type of computer-based tutorial that permit the learner to make decisions beyond a pre-defined set of choices (Bork, 1980b). Hence, students can ask for information, work through a variety of examples that embody some concept, and even construct their own models of the problem. However, the very nature of the computing environment still constrains the terms of the debate. The best examples of interactive proofs usually come from mathematics and science where the nature of the content parallels the nature of the computing environment (i.e., a computable formula captures both). This in turn permits an author (or instructional designer or programmer) to create an interactive proof out of the formula. Of course, even non-mathematical subject areas can be reformulated to be amenable to interactive proofs. Hence, the socio-political problem of hunger can be recast into economic terms and then reduced to a formula that relates an arbitrarily chosen set of variables. The interactive "proof" then proceeds *as if* it were a mathematical problem. This ultimately treats a problem such as hunger as if it were a computerized numbers game—no matter how complex the mathematics. However, computational complexity will *never*

match real-world complexity unless the processes in the world are controlled by comparable mathematical or procedural rules.

Human-Computer "Dialogues" in Tutorial Courseware

We now come to the central assumption of computer-based tutorials. Namely, that human-computer "dialogues" should resemble interpersonal conversations. Bork has modified this claim somewhat by saying that the student is really dialoguing with the author of the computer tutorial rather than with the computer itself, but this is a facile reformulation. Bork also admits that the author of a computer tutorial is trying to manipulate the student by "stimulat[ing] meaningful responses which contribute to learning (Bork, 1980b). Hence, human-computer "dialogues" are a form of behavioral technology where dialogic interactions are controlled by an author who *is not part of the actual interaction.* Responses are only meaningful in light of their contribution to educational performance-gains. The actual confrontations between humans and computing are therefore one-sided affairs because the computers have fixed goal structures, interactive strategies, and deductive capabilities.

What *should* human-computer interactions in a tutorial be called? To answer this we have to compare them with interpersonal interactions. Interpersonal dialogues contain an essential component of *conjoint control* (in spite of the power differentials that may exist between students and teachers). Such conjointness is missing in human-computer interactions. In human-computer "dialogues," students only control the:

a. *Rate (i.e.,* pacing of pre-defined sequences),
b. *Route (i.e.,* any one of a finite number of predefined, or algorithmically constrained, paths toward a pre-defined goal), and
c. *Timing (i.e.,* speed of individual responses).

All other control resides in the courseware program. Students therefore only have a form of pseudo-control because the actual interaction follows a pre-planned, goal-oriented procedural network. Hence, human-computer tutorial interactions are best called *"utilogs"* rather than "dialogues" (Shneidermann, 1980). Of course, utility here is defined by a courseware author who in

turn is restricted to
framework (Ellul, 19

What are tl
interactions in a
intentions of an au
implications for edu
interactions:

a. *Humar*
symbol
the con
actual
only o
algorit
cannot
beings.
of the
that fr
to ada

engage
person
into a
this is
The p
progra
formal
a mea
model
progra
implic

intell
(Gree
tutori
The o
pseud
More

:esses in the world are
procedural rules.

:torial Courseware
:ion of computer-based
r "dialogues" should
ork has modified this
it is really dialoguing
rather than with the
mulation. Bork also
tutorial is trying to
meaningful responses
30b). Hence, human-
oral technology where
thor who *is not part of*
meaningful in light of
ice-gains. The actual
:ing are therefore one-
fixed goal structures,
lities.
actions in a tutorial be
compare them with
dialogues contain an
in spite of the power
ts and teachers). Such
ter interactions. In
control the:
d sequences),
number of predefined,
, paths toward a pre-

ual responses).
e program. Students
rol because the actual
-oriented procedural
interactions are best
Shneidermann, 1980).
seware author who in

turn is restricted to certain categories within the technological framework (Ellul, 1980b; Turkle, 1984).

What are the deeper implications of having utilogic interactions in a computer tutorial shaped by the external intentions of an author? I will summarize a number of these implications for education by comparing them with interpersonal interactions:

a.   *Humans are treated as data-based, rule-following, symbol-manipulating, information processors:*

This implication emerges from the nature of the computer technology that is used to carry out the actual tutorial interactions. Machine processes can only operate on explicit information according to algorithmic rules (Weizenbaum, 1976). Computers cannot semantically or affectively engage human beings. Humans therefore have to adapt to the *nature* of the computational environment—although within that framework, computer processes can be designed to adapt to the "individual differences" of humans.

Earlier, I mentioned that computers only engage *data* from an individual and not the actual person. These data are organized by the program into a model of the learner (in simpler programs, this is merely a data-base of variables and values). The particular model of an individual which the program contains (or builds up) always remains a formal and abstract *type*. Furthermore, this model is a *means* to an end. This is even the case when many models of students (i.e., many generic types) are programmed into the computer. This has serious implications for education.

Since human beings develop personal intellectual agency through dialogic interactions (Greene, 1983), the learner in computer-based tutorial interactions can *never* develop such agency. The only control a learner ever has is a form of pseudo-control within a technological framework. More significantly, since human beings tend to

model the "other" in dialogic interactions (Scheibe & Erwin, 1979), computer-based tutorials may actually teach students to treat other dialogic partners as *anthropomorphized processes* and *means* rather than as ends (Turkle, 1984).

In interpersonal dialogic interactions, on the other hand, individuals encounter, confront, accept (to a greater or lesser degree), and engage each other as unique individuals. True discourse, in this case, requires the acceptance of the "other" in the discourse as a unique and intentional being. It also requires a similar image of self. The individual, in interpersonal dialogues, is therefore a *unique ontological entity* (rather than a generic type) and an *end* (rather than a means). This sets the stage for personal agency in learning. Interpersonal dialogue can, of course, become mechanical if the humans involved act on the basis of some stereotypical inferences about each other. However, the potential for true discourse is *always* present in interpersonal interactions. This potential can *never* exist in human-computer interactions (Weizenbaum, 1976).

Another consequence of viewing humans as rule-following information-processors (as opposed to individuals with unique intentionalities) is that uniform education goals, methods, and outcomes are legitimized. Uniformity in education is enforced not only because the instructional systems attempt to shape a uniform product (*i.e.*, pre-specified learning outcomes) but also because the very conceptualization of the individual places "semantic and syntactical constraints on acceptable language for the discussion of human beings" (Strike, 1974). This in turn makes it impossible to express and legitimize other conceptions of human beings, educational goals, and methods outside of the technological

eractions (Scheibe &
torials may actually
dialogic partners as
d *means* rather than

interactions, on the
ter, confront, accept
nd engage each other
scourse, in this case,
ther" in the discourse
ng. It also requires a
he individual, in
herefore a *unique*
generic type) and an
is sets the stage for
ng. Interpersonal
e mechanical if the
he basis of some
ach other. However,
is *always* present in
s potential can *never*
ter interactions

viewing humans as
essors (as opposed to
ntionalities) is that
ds, and outcomes are
ducation is enforced
al systems attempt to
re-specified learning
very conceptualization
ntic and syntactical
language for the
trike, 1974). This in
xpress and legitimize
beings, educational
of the technological

framework (Ellul, 1980b). A simple example of an alternative framework can make this clear.

In order to create a community and carry out community action, we must (Newmann & Oliver, 1967):

     i. Unconditionally accept individual uniqueness and diversity.
     ii. Carry out an on-going dialectical synthesis of opposing viewpoints with the actual members of the community, and
     iii. Respect emergent community goals.

The technological framework makes this perspective impossible to conceptualize (let alone operate) because it is based on an opposing set of assumptions. The community framework builds on three things: the uniqueness of each individual and his or her viewpoint, a dialectical rationality that tries to synthesize opposing views, and emergent community goals. The technological framework, on the other hand, builds on: the generic characteristics of individuals, a means-end rationality, and a pre-determined set of performance goals.

b. *Machine processes will eventually match human processes:*

This second implication derives from the first implication: if humans are ultimately rule-following information processors, then computers will eventually do everything that humans can do. Human-computer utilogs will then in fact become dialogues because both sides of the interaction will have identical ontological status.

This statement has some serious problems for education, however, even if we only restrict ourselves to the cognitive domain. If we as educators accept the responsibility for the growth of young minds, then we are obligated to ask how such minds do in fact grow. Furthermore, if we find that mental development at all levels requires a dialectical

synthesis of personally and socially constructed meanings, then we can see that the very ontology of the technological framework (*i.e.,* the world is made up of specifiable and controllable processes) is inadequate for the whole domain of intentions and interpersonal meanings (Greene, 1983). *Machine processes*, in this case, *will never replace interpersonal interactions*!

　　　　Finally, if we find that human skills and knowledge are ultimately based on tacit beliefs and judgments which cannot be analyzed into components, then computational processes (which by their nature reduce similarity *judgments* to computable comparisons of component identities) will never match human processes (Dreyfus, 1979).

c.　*Education will be viewed as a form of training and will be subject to explicit, extensional logic:*

　　　　Having an expert author design the goals, rules, and actual messages for a human-computer interaction means that the logic of prediction and control (i.e., the technological framework) is applied to developing pre-planned performance outcomes. The whole educational enterprise is then reduced to means-end rationality because the ends are specified first and then the most efficient means are employed to guarantee a quality product (Nunan, 1983; Apple, 1982). The resulting mechanization of interaction is sometimes transparent when computers carry out the actual interaction because of the sophistication, speed, and variety of media involved (Weizenbaum, 1976). But we should never confuse sophisticated technique with sophisticated instruction (Amarel, 1983). Technique does not have a tacit dimension whereas all human knowledge and learning does (Polanyi, 1966). Technique is solely subject to extensional logic whereas knowledge is subject to *both* intensional and extensional logic.

ocially constructed
he very ontology of
, the world is made
able processes) is
n of intentions and
e, 1983). *Machine
l never replace*

human skills and
on tacit beliefs and
e analyzed into
processes (which by
ity *judgments* to
mponent identities)
es (Dreyfus, 1979).
*orm of training and
ional logic:*

r design the goals,
a human-computer
ic of prediction and
amework) is applied
formance outcomes.
se is then reduced to
ase the ends are
t efficient means are
ity product (Nunan,
ing mechanization of
transparent when
nteraction because of
d variety of media
But we should never
e with sophisticated
hnique does not have
l human knowledge
966). Technique is
nal logic whereas
h intensional and

We can see the implications of this view for education most clearly when we examine the nature of experiential learning within the technological and non-technological frameworks (Kolb, 1984). I will do this by comparing the following notions: the nature of experience, events, and activities; the concept of individual; the methods of knowledge; and the types of thinking involved. My discussion will necessarily be brief.

When restricted to a computational environment, "experiences" take on the form of puzzles *of the same type* (*i.e.*, computable, quantifiable, procedural). Events are non-historical because they are reversible (*i.e.*, declarative and procedural) and activities are restricted to a non-dialectical "artificial reality" (Kreuger, 1983). Furthermore, an individual is only trivially unique (i.e., the variables of a student model in the computer are generic; only the *values* of the variables are unique). Finally, an individual needs only the ability to decode abstract symbols because the "text" and "context" are pre-determined by an external agent, and because knowledge is expressed in an explicit, abstract form. Critical and dialectical thinking are not needed because they make too many things problematical, ambiguous, and non-controllable.

In a natural environment, on the other hand, "experiences" are made up of indefinite *types*. Events are ambiguous, historical, and irreversible (Whitehead, 1929). Activity involves a confrontation between persons and events, and meanings are personally and interpersonally constructed. Experiences and actions are dialectical and historical. Furthermore, natural experiences entail an accommodation to, and assimilation of, an indefinite variety of uniqueness in persons, ideas, and events (Piaget, 1970). These in turn become the

experiential basis for further critical and dialectical thinking. Finally, individuals need interpretive as well as decoding skills because they are forced to construct as well as deconstruct the meanings-in-use of others (Greene, 1978). Interpersonal dialogue plays a central role here because knowledge is dialectical, historical, and subject to transformation.

This brief discussion of experiential learning points out the tremendous restrictions that the technological framework places on the variety of educational experiences. Computer-based tutorials therefore seem to rule out everything that is of value to the individual in the natural and social worlds.

Summary of Tutorial Courseware Programs

I began my discussion about tutorial courseware programs with the question of whether human-computer "dialogues" go beyond drill-and-practice approaches. The answer is both yes and no: such tutorials do go beyond because they are a more sophisticated form of interaction, but they also stay well within the bounds of behavioral and technological framework. That is: behavioral outcomes are still pre-specified by expert agents outside of the actual interaction, "quality control" procedures are still used to guarantee that the learner will reach the intended outcomes, and learners are still only given a form of pseudo-control (*i.e.,* rate, route, and timing). Furthermore, although the actual interaction is less rigid than a drill-and-practice courseware, the interaction is still constrained by a computable algorithm, is still focused on maximizing education performance gains, and still treats the learner as a means toward someone else's end. Computer-based tutorials are biased against experiential learning (outside of the technological framework), quantum leaps in learning, and reflective thinking. Their value in education is therefore very limited.

The question now arises about the case where computers are used as "intellectual tools." Does *this* use of computers go beyond the limitations discussed so far? On first reflection, personal intellectual agency seems to be a natural concomitant of

the "tool" use of con careful analysis.

Comp

What is the (Luehrmann, 1980)? earlier discussion an an intellectual, pro computer as a perso rather with the comp This brings me in simulations.

So far, I h tutorial courseware into the learning building (the terms efficiency and per systematically de Furthermore, know they are conceptu design and the con execution of instru a fine-grained contr system's performa process and makes

Does the sit the learner progr (Critchfield, 1979)? rationality of an e managing the lear control of the whole

The gener student who contro framework of the computer and the computer). My ans that they be used define the user's n Computational in

the "tool" use of computers, but this conclusion requires a more careful analysis.

Computers as Intellectual Tools

What is the intellectual dimension, if any, of computers (Luehrmann, 1980)? To answer this question, I will build on my earlier discussion and then examine the nature of the computer as an intellectual, problem-solving tool. I will *not* deal with the computer as a personal productivity tool (*e.g.*, word processor) but rather with the computer as an "object to think with" (Papert, 1980). This brings me into the realm of computer languages and simulations.

So far, I have described how drill-and-practice and tutorial courseware programs introduce a means-end rationality into the learning process. Knowledge acquisition and skill building (the terms themselves are revealing) become subject to efficiency and performance criteria, and learning becomes a systematically designed and rationally managed process. Furthermore, knowledge and skills become commodified because they are conceptualized in utilitarian terms and because the design and the conception of instruction are separated from the execution of instruction. This commodification, in turn, permits a fine-grained control of the learner's, the teacher's, and even the system's performance. The computer formalizes this whole process and makes it capital intensive.

Does the situation described above apply to the case where the learner programs his or her own solutions to problems (Critchfield, 1979)? Surely here, we will not see the means-end rationality of an external agent conceptualizing, designing, and managing the learning process. After all, the *learner* is now in control of the whole process!

The general question therefore becomes whether the student who controls the computer can go beyond the technological framework of the computer (the values associated with the computer and the symbol systems that can be manipulated by the computer). My answer will proceed as follows: tools tend to insist that they be used in certain ways and intellectual tools tend to define the user's mental landscape (Bruner, 1975; Greene, 1978). Computational intellectual tools (i.e., computer programming

and simulations) therefore bias our ways of knowing and dealing with the world toward extensional knowledge (i.e., the quantitative, declarative, and procedural kinds of knowledge) and hide other kinds of knowledge. Intensional knowledge, of course, will not go away. It will only be delegitimized by computers.

The Computer as an Intellectual Problem-Solving Tool

Before answering the general question, I would like to discuss how a computer might bias our ways of knowing. A computer is basically a box that manipulates symbols (and information) according to a plan. When someone else writes the plan (i.e., the program), then we are forced to follow their set of procedures. When we write the plan, then we are forced to use the computer's language. In both cases, we are confronted with a question about the nature of the plans and the types of symbols which the computer can manipulate. We therefore need to explore how programming a particular problem in a computer language helps us learn and think about a problem (Taylor, 1980).

Computers, as I mentioned, are boxes that manipulate symbols according to a plan. The symbols are actually only energy states in an electronic machine which are transformed according to formal, algorithmic rules. Hence, a *computer does not add* "1+1" to get "2." Rather, the computer initiates an electronic process where two energy states, which we identify as representing the numeric unit "1," are transformed according to an internal structural process (hardwired or software, it matters not) into another energy state which we identify as representing the numeric unit "2." I am belaboring this point because of its generality. If a computer manipulates two high-level representational constructs such as "All men are mortal" plus "Socrates is a man" and ends up with "Therefore Socrates is a mortal," then *nothing has been added by the computer beyond my first example*. The computer only manipulates semantically empty energy states (which we call input or data or symbols) according to syntactical rules—no matter how high-level those rules. It is *we* who actively construct and ascribe meaning to these semantically empty mechanisms. A computer language is therefore *not* a language in the traditional sense of the term (i.e.,

expressive, intentio
based on qualitativ
notations to contrc
1980). Hence, *a co
the syntactical dim*
those who equate
potential of comput
because "all relev
syntactic distinctio
latter case, becom
that act on symboli

What are
computer language
that if humans ar
then they must v
these tools (Mow
manipulate expli
syntactical rules,
knowledge that fi
types of knowledg
can only be proces:
terms. I describec
example. Hence,
characteristics of l
governed order,
ambiguity, non
contradiction (i.e.,
aspects. They als
the only acceptabl

By way of
following characte
goals, self-constri
and tacitness, a
simultaneity of i
finally, they ten
methods: abduct
dialectical synthe
more computers a

expressive, intentional, and connotative as well as denotative and based on qualitative knowing, etc.) but rather a set of syntactical notations to control computer operations (Wirth, 1976; Iverson, 1980). Hence, *a computer's expressive potential only extends over the syntactical dimension of its formal operations.* Of course, for those who equate cognition with computation, the expressive potential of computer languages extends into the semantic domain because "all relevant semantic distinctions [are] mirrored by syntactic distinctions" (Pylyshyn, 1980, 1984). Semantics, in the latter case, become a set of rule-governed, cognitive operations that act on symbolic representations.

What are the epistemological implications of using computer languages to represent events in our world? It is clear that if humans are going to use computers as intellectual tools, then they must work within the epistemological limitations of these tools (Mowshowitz, 1976). Since computers can only manipulate explicit data and symbols according to formal, syntactical rules, computers tend to legitimize those types of knowledge that fit into their framework and delegitimize other types of knowledge (Strike, 1974). The latter types of knowledge can only be processed when they are reformulated into computable terms. I described such a transformation earlier with the hunger example. Hence, computers tend to *legitimize* the following characteristics of knowledge (Broughton, 1984; Olson, 1985): rule-governed order, objective systematicity, explicit clarity, non-ambiguity, non-redundancy, internal consistency, non-contradiction (i.e., logic of the excluded middle), and quantitative aspects. They also tend to legitimize deduction and induction as the only acceptable epistemological methods.

By way of contrast, computers tend to *delegitimize* the following characteristics of knowledge (Streibel, 1983): emergent goals, self-constructed order, organic systematicity, connotation and tacitness, ambiguity, redundancy, dialectical rationality, simultaneity of multiple logics, and qualitative aspects. And finally, they tend to delegitimize the following epistemological methods: abduction, interpretation, intuition, introspection, and dialectical synthesis of multiple and contradictory realities. The more computers are used as intellectual tools, therefore, the more

this process of legitimization and delegitimization takes hold. The more we rely on the formal characteristics of knowledge, the less we rely on the tacit and interpretative dimensions of knowledge. It is almost as if the technological framework is not only incompatible with other ways of knowing, but inevitably excludes them from our mental landscapes as well (Ellul, 1980b). Of course, the formal and the tacit dimensions of knowledge can never be separated from each other (Polanyi, 1966). The tacit dimension can only become hidden.

The foregoing discussion brings us back to an earlier conclusion: computers force us to act *as if* we were rule-governed information processors. They also force us to construe thinking as " 'cognitive problem-solving' where the 'solutions' are arrived at by formal calculation, computation, and rational analysis" (Broughton, 1984). Even if we are active and constructive and intuitive in our approach to the world, we must still analyze and reduce problems into explicit and procedural terms. Hence, we must restrict our thinking to cognitive operations. The concept of the computer as an intellectual tool is therefore *not* a neutral formulation because it forces us to objectify ourselves as agents of prediction, calculation, and control (Weizenbaum, 1976). Personal intellectual agency has thereby been limited to the technological framework. This has serious consequences for education.

Computer Programming and Computer Simulations

We can easily see how programming is a paradigm of *thinking* in the context of the tool use of computers. If the only legitimate knowledge entails objective facts, explicit representations of facts as data, and formal operations on these representations, then programming is the ideal way to process such knowledge. The same can be said of programming as a paradigm for *learning*. If the only way to think about things is through analysis and procedural debugging, then programming is also the ideal way to *learn* how to deal with the world. After all, we are not just learning to act *as if* we were computers, we are developing operational and representational cognitive structures to deal with *any* aspect of the world. Gone are aesthetic,

metaphoric, artistic, affective, interpretive, and moral structures for dealing with the world!

We can therefore understand how many of the chief advocates of the tool use of computers see computer literacy as the ability to "do computing" (Luehrmann, 1981) and see computer programming as the best way to shape a child's cognitive development (Papert, 1980). However, in this scheme of things, we can also see that our rational life is thereby reduced to a set of operational, problem-solving skills—to say nothing about our emotional life.

Is there anything positive to be gained from programming aside from the actual technical skills? In several studies, very little positive transfer was found from programming to other domains of cognitive problem-solving (Coburn et al., 1982; Pea & Kurland, 1983). However, this conclusion is only tentative because the field is still too new (Linn, 1985). We must therefore fall back on an analysis of the nature of programming in order to see what is possible with this approach.

Computing, as Arthur Luehrmann, one of the chief advocates of programming, argues (Luehrmann, 1981):

> belongs as a regular school subject for the same reason as reading, writing, and mathematics. Each gives the student a basic intellectual tool with wide areas of application. Each gives the student a distinctive means of thinking about and representing a problem, of writing his or her thoughts down, of studying and criticizing the thoughts of others, whether they are embodied in a paragraph of English, a set of mathematical equations, or a computer program. Students need practice and instruction in all these basic modes of expressing and communicating ideas.

This certainly is an admirable statement because it integrates computing (*i.e.,* algorithmic, procedural thinking) into the other "basics" of education (*i.e.,* reading, writing, and arithmetic) (Kroener, 1981; Ershov, 1981). Luehrmann's argument also casts programming as an aid to understanding.

Given the arguments described above, how could anyone possibly object to programming as a subject matter in our schools?

The answer is very simple and applies to the other "distinctive means of thinking and representing a problem": whenever technique is emphasized over grappling with content, then the innermost principles of that content are lost (Ershov, 1972). Although this is also true for reading and writing, this is especially true for computer programming because the computer is an instrument of technique *par excellence*. The computer can *only* manipulate content-free symbols according to formal procedures. Hence, although computer programming may force one to structure *information* in precise and systematic ways and carry out *logical operations* on abstract representations of that information, it tells us *nothing* about *what* information should be treated in this way. It also tells us nothing about the *nature* of the real world. A simple computer-simulation example should make this clear. The same argument applies to programming the simulation.

Oregon Trail is a popular computer simulation that records the problems which the pioneers had in crossing the American frontier (Grady, 1983). It provides a simplified environment for elementary-aged school children where they can make "decisions" and watch the consequences of their "actions." Hence, forgetting to "buy" enough bullets inevitably leads to program termination. A student can "win" if he or she keeps a careful record of the "purchases" and analyzes the relationships between events, supplies, and mileage. What we have here, however, is quantitative, artificial reality with no hint of the lived reality (Grady, 1983). The simulation, in fact, represents the abstract world of algorithmic logic rather than the lived-experience of historical logic. Hence, whether one is simply using the finished simulation to learn about history or whether one is programming the simulation, historical logic is incapable of being represented. It would be more justifiable to say that winning here (i.e., solving the problem) is more the result of looking for patterns among the numbers than developing a sense of history (Grady, 1983). The simulation is simply a well-disguised numbers game.

One might object to the foregoing discussion on a number of grounds:

a. The al[
    parallel
    as matl
    a place
b. All lea
    unknov
    that sin
c. Person;
    within
    reality-

Each of these objec
warrants our attenti
also be interpreted
analysis.

The first obj
case that many real
procedural structure
Hence, learning to
program because prc
subtraction (Dreyfus
about the *reasons* for

Brown & Bu
tutor that recognize;
subtraction (Brown &
models one way in v
subtraction. This is
may be very useful i
procedures surroun
say: fine, this w:
subtraction skill m
wait a minute—sub
skill that leads to no
to competence in *usi
maximum, it shoul
both cases, it shoul
generate personal e
procedures because
As Dreyfus and Dre

a. The algorithmic logic of simulations does in fact parallel a similar logic in *some* content areas (such as mathematics) so that computer simulations have a place in education.

b. All learning proceeds from the known to the unknown (and from the simple to the complex) so that simulations are a stepping stone to life.

c. Persons can learn to become autonomous inquirers within the limitations of a safe and simple artificial reality—a skill that they can later use in real life.

Each of these objections has an intuitive appeal and therefore warrants our attention. Each of these statements, however, can also be interpreted in several ways, so each deserves a careful analysis.

The first objection is easy to handle. It is certainly the case that many real-world activities contain the same logical and procedural structure that is found in the realm of computation. Hence, learning to subtract can be modeled in a computer program because procedural rules are all there is to the *process* of subtraction (Dreyfus & Dreyfus, 1984). But what does this tell one about the *reasons* for these procedures?

Brown & Burton have developed an "intelligent" computer tutor that recognizes over ninety ways to make a mistake during subtraction (Brown & Burton, 1979). Each of these procedural bugs models one way in which a person can go wrong in the process of subtraction. This is certainly a very sophisticated approach and may be very useful in some cases. However, it only elaborates the procedures surrounding subtraction. A logical positivist would say: fine, this will help establish the automaticity of the subtraction skill more efficiently. But an educator would say: wait a minute—subtraction is not an isolated, de-contextualized skill that leads to nothing but itself. At a minimum, it should lead to competence in *using* subtraction with real-world problems. At a maximum, it should lead to mathematical understanding. In both cases, it should be connected to experiences that ultimately generate personal expertise. And expertise *cannot* be reduced to procedures because it involves judgment as well as calculations. As Dreyfus and Dreyfus (1984) conclude:

at the higher stages of skill acquisition, even if there
are rules underlying expertise, the rules that the expert
has access to are not the rules that generate his
expertise . . . [Hence], trying to find rules or
procedures in a domain often stands in the way of
learning *even at the earliest stages* [my emphasis].

Developing procedure-following skills, therefore, does *not*
facilitate broader learning. I have used subtraction as an
example in this discussion because it involves a procedural skill.
My argument applies even more for non-procedural kinds of
expertise and understanding (*e.g.*, historical expertise and
understanding).

The second objection is more difficult to handle: all
learning proceeds from the known to the unknown and from the
simple to the complex. But we have to be very careful how we
define simple so that we do not prejudge the *nature* of the complex.
This problem is a perennial concern in the philosophy of science:
should we base our scientific concepts on our intuitions and lived
experiences, or should we base them on counter-intuitive
conceptual constructions that happen to fit empirical facts (Kuhn,
1962)? This problem emerges in education in a number of forms.
For example, in science education, should we teach young
children to be Artistotelians before Newtonians—let alone before
Einsteinians (DiSessa, 1982)?

In the context of this discussion, the simplicity question
becomes: is a simple, context-free, quantitative, and procedural
simulation *ever* an adequate preparation for a complex,
contextual, qualitative, and non-procedural lived-experience
(Megarry, 1983)? If we wanted to prepare children to understand
and deal with the real world, shouldn't we develop simple
learning situations *of the same kind* as those they will later
encounter in a more complex form? Isn't problem-solving, in
fact, domain-specific no matter how high-level the activity
(Newell & Simon, 1982; Lester, 1980; Pea & Kurland, 1983)? The
"ivory tower" aspect of schooling might be just the right protection
against the harsh realities of life, but this does *not* mean that
schools should become "artificial realities." Using the computer
to develop problem-solving skills, however, sets up just such a

dichotomy between "s
natural reality." Notice
the above dichotomy is u
(Noble, 1984). Learning
not be the best way to pr

The final objecti
persons develop anal
limitations of a computa
in real life? After all,
general and more like "
(Streibel, 1985). The q
reveal what has been
skills which are dev
dialectical, and judg
useful within the lived
ambiguity, interpersc
dialectical thinking, th
and judgment-based act
in this way, a positive
this is twofold: the com
that crowds out other fo
problems, and thinking
(*i.e.*, problem-solving
analysis and inquiry ca
of dealing with similar
so well for this reason-
event are controlled
Furthermore, the flight
the nature of the real wo

A final answer t
does seem, however, tha
utilitarian, problem-sol
under our control does
have delegitimized oth
skills display "intelli
intelligence itself has b
Broughton laments (Bro

dichotomy between *"simple* artificial reality" and *"complex* natural reality." Notice that the *artificial-natural* dimension of the above dichotomy is usually hidden in the debate on the matter (Noble, 1984). Learning to program the computer may therefore not be the best way to prepare children for real life.

The final objection is the most difficult to answer: can persons develop analytical and inquiry skills within the limitations of a computational environment that can then be used in real life? After all, analytical and inquiry skills are very general and more like "frames of mind" than simple procedures (Streibel, 1985). The question can be reformulated, however, to reveal what has been hidden: are the analytical and inquiry skills which are developed within a non-contextual, non-dialectical, and judgment-free computational environment useful within the lived environment that requires tolerance for ambiguity, interpersonal construction of new meanings, dialectical thinking, the acceptability of incomplete solutions, and judgment-based actions? When the question is reformulated in this way, a positive answer becomes doubtful. The reason for this is twofold: the computer embodies a technological framework that crowds out other forms of conceptualizing and understanding problems, and thinking is only ever thinking about something (*i.e.,* problem-solving is domain-specific). Hence, mature analysis and inquiry can therefore only be the result of a history of dealing with similar *kinds* of things. Flight simulators work so well for this reason—both the simulation and the real-world event are controlled by the same kinds of procedures. Furthermore, the flight simulator is *both* simpler as well as true to the nature of the real world situation.

A final answer to the third objection remains to be seen. It does seem, however, that the computer restricts our rational life to utilitarian, problem-solving skills. Saying that such skills are under our control does not help very much because these skills have delegitimized other ways of knowing. Saying that such skills display "intelligence" does not help either because intelligence itself has been redefined in a restricted manner. As Broughton laments (Broughton, 1984):

one can measure the educational impact of computers,
and particularly of learning to program them, in
terms of what is lost in the process. To the curriculum
is lost the arts and the humanities. To pedagogy is lost
the hermeneutic art and language that allows us to ask
about the meaning of things and of life, to interpret
them in their many and various cultural horizons. To
both is lost the self and the autonomous capacity to
examine critically what we interpret.

Hence, although problem-solving with a computer appears more
desirable and high-level than computerized drill-and-practice,
programming still limits us to the technological framework.
Using a computer as an intellectual tool is therefore a more subtle
form of behavioral learning technology because the computer and
its "languages" (*i.e.*, information, symbol systems, and
activities) shape the very categories with which we apprehend and
think about the world, and because it is also done with the active
consent and participation of the learner. It represents, as one
author has called it, the "industrialization of intellectual work"
(Ershov, 1972). This is particularly disturbing because
programming (as well as drill-and-practice and tutorial
courseware) is being introduced to children in their most plastic
and formative years (Bitter, 1982/83; Cuffaro, 1984).

Summary of the Tool Use of Computers

I began this section with the question of whether the
"intellectual tool use" of computers went beyond the limitations of
tutorial interactions. The answer again is both yes and no.
Computers do help us develop a limited personal intellectual
agency by forcing us to structure information in precise,
systematic ways and specific logical operations on that
information (Linn, 1985). However, this agency only develops
within the computational domain. Hence, we are left with an
under-developed intellectual agency within the qualitative,
dialectical, and experiential domain of natural and social
events. Learning to program is therefore only a good way to learn
and think about procedural problems, although even here there are
some limitations.

The root of the difficulty seems to reside in the nature of computer languages: the expressive potential of computer language only extends over the syntactical dimension of computer operations. This contrasts sharply with the expressive potential of natural languages which extend over the aesthetic, metaphoric, artistic, affective, and moral domains (Ong, 1982). Why can't these various languages co-exist? The answer boils down to this: computer languages are part of a technological framework, which, when applied to a number of problems, delegitimizes other frameworks. We are then left with a very restricted mental landscape.

General Summary

I have examined the three major approaches to the use of computers in education and found serious limitations with each approach. The drill-and-practice approach was shown to embody a deterministic, behavioral technology that turned learning into a systematically designed and quality-controlled form of work. Although drill-and-practice courseware programs were only intended to supplement instruction, they in fact introduced a technological framework into the classroom culture that mitigated against non-behavioral educational goals. Computerized tutorial programs were shown to extend the behavioral and technological approach to learning even further. That is, in tutorial courseware programs, interactions were still shaped by an external agent's intentions in order to maximize the learner's performance gains and still constrained by computable algorithms. Furthermore, the human learner was still treated as a means toward someone else's ends and only given a form of pseudo-control in the interaction. Most seriously, computerized tutorial interactions pre-empted personal intellectual agency and ultimately inner-directed learning. Finally, the use of computer programming and simulations in education was shown to limit the learner's mental landscape to objective, quantitative, and procedural "intellectual tools." This left the learner with an under-developed intellectual agency within the qualitative, dialectical, and experiential domains of natural and social events.

Each of the approaches described above may have some short-term gain associated with them, but taken together, they represent a shift toward technologizing education. Drill-and-practice courseware programs alter the nature of sub-skill acquisition, tutorial courseware programs restrict the full range of personal intellectual agency, and computer programming and simulations delegitimize non-technological ways of learning and thinking about problems. Taken together, is this worth the price?

REFERENCES

Alderman, D. L. (1978). *Evaluation of the TICCIT computer-assisted instructional system in the community college.* Princeton, N.J.: Educational Testing Service.

Amarel, M. (1983). The classroom: An instructional setting for teachers, students, and the computer. In A. C. Wilkinson (Ed.), *Classroom computers and cognitive science.* New York: Academic Press.

Apple, M. W. (1975). The adequacy of systems management procedures in education. In R. H. Smith (Ed.), *Regaining educational leadership.* New York: John Wiley & Sons.

Apple, M. W. (1979). *Ideology and curriculum.* London: Routledge & Kegan Paul.

Apple, M. W. (1982). *Education and power.* London: Routledge & Kegan Paul.

Barr, R., & Dreeben, R. (1978). Instruction in classrooms. In L. Shulman (Ed.), *Review of research in education* (Vol. 5). Ithaca, IL: F. E. Peacock.

Becker, H. J. (1983). *School uses of microcomputer: Reports from a national survey.* Nos. 1-5. Baltimore, MD: Center for Social Organization of Schools, the Johns Hopkins University.

Benne, K. D. (1975). Technology and community: Conflicting bases of educational authority. In W. Feinberg & H. Rosemont, Jr. (Eds.), *Work, technology, and education: Dissenting essays in the intellectual foundations of*

American education. Urbana, IL: University of Illinois Press.

Berger, P. L., & Luckmann, T. (1966). *The social construction of reality: A treatise in the sociology of knowledge.* New York: Doubleday.

Berlak, A., & Berlak, H. (1981). *The dilemmas of schooling.* New York: Methuen.

Berliner, D. (1982). Viewing the teacher as manager. *Educational Digest, 47,* 20-23.

Bitter, G. S. (1982/93). The road to computer literacy. *Electronic Learning.* Sep., 60-63; Oct., 34-68; Nov.-Dec., 41-91; Jan. 40-48; Feb, 54-60.

Bloom, B. S. (1971). Mastery learning. In J. H. Block (Ed.), *Mastery learning: Theory and practice.* New York: Holt, Rinehart, and Winston.

Bloom, B. S. (1976). *Human characteristics and school Learning.* New York: McGraw-Hill.

Bolvin, J. O. (1982). Classroom organization. In H. M. Mitzel (Ed.), *Encyclopedia of educational research. 5th edition* Vol. 1. New York: Macmillan, pp. 265-274.

Bork, A. (1980a). Interactive learning. In R. Taylor (Ed.), *The computer in the school: Tutor, tool, and tutee.* New York: Teachers College Press.

Bork, A. (1980b). Preparing student-computer dialogs: Advice to teachers. In R. Taylor (Ed.), *The computer in the school: Tutor, tool, and tutee.* New York: Teachers College Press, pp. 15-52.

Bork, A., & Franklin, S. D. (1979). The role of personal computer systems in education. *AEDS Journal,* Fall.

Boyd, G. M. (1983a). Education and miseducation by computer. In J. Megarry, E. R. F. Walker, S. Nisbet, & E. Hoyle (Eds.), *World yearbook of education, 1982/83: Computer and education,* pp. 50-54.

Boyd, G. M. (1983b). Four ways of providing computer-assisted learning and their probable impacts. *Computers and Education, 6,* 305-310.

Broughton, J. M. (1984). Computer literacy as political socialization. Paper presented at the annual meeting of

the *American Educational Research Association*. New Orleans, April 23-27.

Brown, J. S., & Burton, R. R. (1979). Diagnostic model for procedural bugs in basic mathematical skills. *Cognitive Science, 2,* 155-192.

Bruner, J. S. (1975). Language as an instrument of thought. In A. Davis (Ed.), *Problems of language and learning.* London: Heinemann. pp. 61-80.

Bunderson, V. (1981). Courseware. In H. F. O'Neil (Ed.), *Computer-assisted instruction: A state of the art assessment.* New York: Academic Press.

Carlson, D. (1982). 'Updating individualism and the work ethic: Corporate logic in the classroom. *Curriculum Inquiry, 12(2),* 125-160.

Coburn, P., Kelman, P., Roberts, W., Snyder, T., Watt, D., & Weiner, C. (1982). *Practical guide to computers in education.* Reading, MA: Addison-Wesley.

Critchfield, M. (1979). Beyond CAI: Computers as personal intellectual tools. *Educational Technology, 19(10),* 18-25.

Cronbach, L. J., & Snow, R. E. (1977). *Aptitudes and instructional methods.* New York: Irvington.

Cuffaro, H. K. (1984). Microcomputers in education: Why is earlier better? *Teachers College Record, 85(4),* 559-568.

Dennis, J. R. (1979a). The question-episode: Building block of teaching with a computer. *The Illinois series on educational applications of computers,* Vol. 4e. Urbana, IL: College of Education, the University of Illinois.

Dennis, J. R. (1979b). Tutorial instruction on a computer. *Illinois series on educational applications of computers,* Vol. 6e. Urbana, IL: College of Education, The University of Illinois.

DiSessa, A. (1982). Unlearning Aristotelian physics: A study of knowledge-based learning. *Cognitive Science, 6(1),* 37-75.

Dreyfus, H. H. (1979). *What computers can't do* (2nd Ed.). New York: Harper & Row.

Dreyfus, H. H., and Dreyfus, S. E. (1984). Putting computers i their proper place: Intuition in the classroom. *Teacher College Record, 85(4),* 578-601.

Eisner, E. W. (1979). *The educational imagination.* New York Macmillan.

Ellul, J. (1980a). The power of technique in the ethics of non power. In K Woodward (Ed.), *The myth of information Technology and post-industrial culture.* Madison, WI Coda Press.

Ellul, J. (1980b). *The technological system.* New York Continuum.

Ershov, A. P. (1972). Aesthetics and the human factor ii programming. *Datamation, 18(7),* 62-67.

Ershov, A. P. (1981). Programming: the second literacy. In R Lewis & E. D. Tagg (Eds.), *Computers in education Proceedings of the 3rd IFIP world conference.* Lausanne Switzerland, July.

Franklin, B. (1974). *The curriculum field and the problem of social control, 1918-1938. A study in critical theory.* Unpublished Ph.D. Dissertation. Madison, WI University of Wisconsin.

Freire, P. (1970). *Pedagogy of the oppressed.* New York: The Seabury Press.

Freire, P. (1973). *Education for critical consciousness.* New York: Seabury Press.

Furth, H. (1969). *Piaget and knowledge: Theoretical foundations.* Englewood Cliffs, NJ: Prentice-Hall.

Gagné, R. M., Wager, W., & Rojas, A. (1981). Planning and authoring computer-assisted instruction lessons. *Educational Technology, 21,* 17-26.

Gardner, H. (1979). Toys with a mind of their own. *Psychology Today,* Nov., 101.

Gardner, H. (1983). *Frames of mind: The theory of multiple intelligences.* New York: Basic Books.

Goody, J. (1977). *The domestication of the savage mind.* London: Cambridge University Press.

Grady, D. (1983). What every teacher should know about computer simulations. *Learning, 11(8),* 34-46.

Gray, L. (1983). Teacher's unions and the impact of computer-based technologies. In J. Megarry, E. R. F. Walker, S. Nisbet, & E. Hoyle (Eds.), *World yearbook of education, 1982/83: Computers and education*, pp. 29-41.

Greene, M. (1978). *Landscapes of learning.* New York: Teachers College Press.

Greene, M. (1982a). Literacy for what? *Phi Delta Kappan, 63*(5), 326-329.

Greene, M. (1982b). The University and the public. Lecture given at the Pennsylvania State University, University Park, PA. October 14-15.

Greene, M. (1983). The literacy that liberates. Tape No. 612-20312. *Association for Supervision and Curriculum Development.* Alexandria, VA.

Habermas, J. (1981). *The theory of communicative action. Vol. 1. Reason and the rationalization of society.* Boston: Beacon Press.

Hawkins, D. (1974). *The informed vision: Essays on learning and human nature.* New York: Agathon Press.

Iverson, K. (1980). Notation as a tool for thought. *Communications of the ACM, 23,* 444-465.

Jackson, P. W. (1968). *Life in classrooms.* New York: Holt, Rinehart, and Winston.

Keller, F. S. (1968). Goodbye teacher. *Journal of Applied Behavior Analysis, 1*(1), 79-89.

Kliebard, H. M. (1971). Bureaucracy and curriculum theory. In V. F. Haubrich (Ed.), *Freedom, Bureaucracy, & Schooling. Yearbook of the Association for Supervision and Curriculum Development.* New York: National Education Association. pp.74-93.

Kohlberg, L., & Mayer, R. (1972). Development as the aim of education. *Harvard Educational Review, 42*(4), 449-496.

Kolb, D. A. (1984). *Experiential learning: Experience as the source of learning and development.* Englewood Cliffs, NJ: Prentice-Hall.

Kroener, J. D. (Ed.). (1981). *The new liberal arts: An exchange of views.* New York: Alfred P. Sloan Foundation.

Krueger, M. W. (1983). *Artificial reality.* Reading, MA: Addison-Wesley.

Kuhn, T. S. (1962). *The structure of scientific revolutions.* Chicago: University of Chicago Press.

Kulik, J. A., Kulik, C. L. C., & Cohen, P. A. (1980). Effectiveness of computer-based college teaching: A meta-analysis of findings. *Review of Educational Research, 50,* 525-544.

Lester, F. K. (1980). Research on mathematical problem solving. In R. J. Shumway (Ed.), *Research in Mathematics Education.* Reston, VA: National Council of Teachers of Mathematics.

Linn, M. C. (1985). The cognitive consequences of programming instruction in classrooms. *Educational Researcher, 14*(5) 14-29.

Luehrmann, A. (1980). Should the computer teach the student or vice versa? In R. Taylor (Ed.), *The computer in the school: Tutor, tool, tutee.* New York: Teachers College Press, pp 129-140.

Luehrmann, A. (1981). Computer literacy: What should it be? *Mathematics Teacher, 74*(9), 682-686.

Lukes, S. (1973). *Individualism.* Oxford: Basil Blackwell.

Megarry, J. (1983). Thinking, learning, and educating: The role of the computer. In J. Megarry, E. R. F. Walker, S. Nisbet, & E. Hoyle (Eds.), *World yearbook of education, 1982/83: Computers and education.* New York: Kogan Page, pp. 15-28.

Messick, S. (Ed.). (1976). *Individuality in learning.* San Francisco: Jossey-Bass.

Mowshowitz, A. (1976). *The conquest of will: Information processing in human affairs.* Reading, MA: Addison-Wesley.

Murphey, R. T., & Appel, L. R. (1977). *Evaluation of the PLATO IV computer-based educational system in the community college.* Princeton, NJ: Educational Testing Service.

Newell, A., & Simon, H. (1972). *Human problem solving.* Englewood Cliffs, NJ: Prentice-Hall.

Newmann, F., & Oliver, D. (1967). Education and community. *Harvard Educational Review, 37*(1), 61-106.

Nisbett, R., & Ross, L. (1980). *Human inference: Strategies and shortcomings of social judgment*. Englewood Cliffs, NJ: Prentice-Hall.

Noble, D. (1984). Computer literacy and ideology. *Teachers College Record, 85*(4), 602-614.

Nunan, T. (1983). *Countering educational design*. New York: Nichols Publishing.

Olds, H. F., Jr. (1982). The microcomputer—an environment that teaches: Exploring the hidden curriculum. In *Proceedings. The computer: Extension of the human mind*. Third Annual Summer Conference. College of Education. University of Oregon. Eugene, Oregon, pp. 73-85.

Olson, D. R. (1985). Computers as tools of the intellect. *Educational Researcher, 14*(5), 5-8.

O'Neil, H. F., Jr., & Paris, J. (1981). Introduction and overview of computer-based instruction. In H. F. O'Neil, Jr. (Ed.), *Computer-based instruction: A state of the art assessment*. New York: Academic Press.

Ong, W. (1982). *Orality and literacy. The technologizing of the word*. New York: Methuen.

O'Shea, T., & Self, T. (1983). *Learning and teaching with computers*. Englewood Cliffs, NJ: Prentice-Hall.

Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. New York: Basic Books.

Pea, R. D., & Kurland, D. M. (1983). On the cognitive effects of learning computer programming. *Bank Street technical report #18*. New York: Bank Street College.

Piaget, J. (1970). *Genetic epistemology*. New York: W. W. Norton.

Polanyi, M. (1958). *The tacit dimension*. Garden City: Doubleday.

Pylyshyn, Z. W. (1980). Computation and cognition: Issues in the foundations of cognitive science. *The Behavioral and Brain Sciences, 3*, 111-169.

Pylyshyn, Z. W. (1984). *Computation and cognition*. Cambridge, MA: MIT Press.

Salisbury, D. F. (1984). Cognitive psychology and it: implications for designing drill-and-practice program: for computers. Presented at the annual conference of th American Educational Research Association, Nev Orleans, April.

Sarason, S. B. (1982). *The culture of the school and the problem o change.* 2nd Edition. Boston: Allyn & Bacon.

Scheibe, K. E., & Erwin, M. (1979). The computer as altar *Journal of Social Psychology, 108*, 103-109.

Scriven, M. (1975). Problems and prospects fo: individualization. In H. Talmadge (Ed.), *Systems o. individualized instruction*. Berkeley, CA: McCutchan pp. 199-210.

Schneidermann, B. (1980). *Software psychology: Human factor: in computer and information systems*. Cambridge, MA Winthrop.

Shor, I. (1980). *Critical teaching and everyday life*. Boston South End Press.

Skinner, B. F. (1968). *The technology of teaching*. Englewoo Cliffs, NJ: Prentice-Hall.

Sleeman, D., & Brown, J. S. (1982). *Intelligent tutoring systems* New York: Academic Press.

Sloan, D. (1984). On raising critical questions about the compute: in education. *Teachers College Record, 85*(4), 539-547.

Sperry, L. (Ed.). (1972). *Learning performance and individua differences*. Glenview, IL: Scott Foresman.

Stewart, J. T. (1979). Drill-and-practice on a computer. *Illinoi series on educational applications of computers*. Vol. 7e Urbana, IL: College of Education, The University o Illinois.

Streibel, M. J. (1983). The educational utility of LOGO. *Schoo. Science and Mathematics, 83*(6), 474-484.

Streibel, M. J. (1985). Beyond computer literacy: Analytica skills, inquiry skills, and personal empowerment *Technical Horizons in Education, 12*(10), 69-73.

Strike, K. A. (1974). On the expressive potential of behaviorisl language. *American Educational Research Journal, 11*(2), 103-120.

Suppes, P. (1966). The uses of computers in education. *Scientific American, 215*(3), 206-220.

Talmadge, H. (1975). *Systems of individualized instruction.* Berkeley, CA: McCutchan Publishing.

Taylor, R. (Ed.). (1980). *The computer in the school: Tutor, tool, tutee.* New York: Teachers College Press.

Turkle, S. (1984). *The second self: Computers and the human spirit.* New York: Simon & Schuster.

Weizenbaum, J. (1976). *Computer power and human reason.* San Francisco: W. H. Freeman.

Whitehead, A. N. (1929). *The aims of education and other essays.* New York: Free Press.

Wirth, N. (1976). *Algorithms & data structures = programs.* Englewood Cliffs, NJ: Prentice-Hall.

Wolcott, H. F. (1977). *Teachers versus technocrats.* Eugene, OR: University of Oregon.

17

The Use of Computers in Education:
A Response to Streibel*

Robert Heinich
Instructional Systems Technology
Indiana University

This article is a reply to Michael Streibel's "Critical Analysis of the Use of Computers in Education" [*ECTJ* 34(3), Fall 1986]. I congratulate Streibel for forcing us to consider issues we prefer to sweep under the intellectual rug. Too often we act as "mindless technicians," as James D. Finn once put it, with little thought given to our philosophical assumptions about learning, the organization of instruction, and instructional vehicles (including teachers). The level of discourse in the article is at a consistently high level.

However, I am afraid that Streibel has fallen into the trap described by Charles Kettering as using logic to go wrong systematically. It is a trap that I have been caught in myself on more than one occasion. There is an irresistible urge to push an argument too far, with a consequent loss of touch with experience. For example, Streibel's analysis of the logic of computer-based instruction (p. 285) is driven by his own assumptions, a common enough beginning and one we all engage in. But what evidence (as opposed to the testimony, speculation, assertions, or logical inferences of others) is offered that these assumptions are correct and the conclusions confirmed?

Paradigms Regained

The Uses of Illuminative, Semiotic and Post-Modern Criticism as Modes of Inquiry in Educational Technology

A Book of Readings

edited by

Denis Hlynka
University of Manitoba

and

John C. Belland
The Ohio State University

Educational Technology Publications
Englewood Cliffs, New Jersey 07632

The paradigm
1962 landmark work t:
has been slowly dr:
alternative methodolog
educational technology
the vanguard of chang
behind other fields an(
already spent a dec:
genuine research alter
have become stuck fa
which goes by a variet;
instructional developi
plain *educational tec.*
positivistic conception
formal experiments w
would predict and
technologically-based
      Yet it would
technologists have
alternatives. It is just t
recorded such explor
"qualitative methodc