TESOL Teachers of English to Speakers of Other Languages

COMPUTER-ASSISTED LANGUAGE LEARNING-INTEREST SECTION

Volume III

December 1986

TIME AND DESIGN CONSTRAINTS OF DEVELOPING MULTI-BRANCHING LANGUAGE INSTRUCTION

by Christopher G. Johnson
The University of Arizona

Robert Taylor (1980, p. 3) describes the "computer as tutor" as follows:

> To function as a **tutor** in some subject (such as language instruction), the computer must be programmed by "experts" in programming and in that subject. The student is then tutored by the computer executing the program(s).... With appropriately well designed software, the computer tutor can easily and swiftly tailor its presentation to accommodate a wide range of student differences.

How do we ensure the development of "appropriately well-designed software" for language instruction? One continually hears statements that indicate that software designers are not always adequately addressing this question (e.g. "the software doesn't cover what I need to cover"; "the program isn't factually correct"; "the range of material covered is too narrow"; "it is inappropriate for my students"). In general, such software deficiencies

> tend to fall into two primary categories. The first category consists of deficiencies that are technological or functional in nature, that is, deficiencies or defects associated with the operation of the software as well as the extent to which programs utilize the potential contained in the hardware. The second category includes deficiencies of a pedagogical nature. In this category we place concerns about learning theory, diagnostic and individualizing components, and the congruence of subject matter, mode of delivery, and developmental level of the students (Helm 1984, p. 10).

Many of these deficiencies stem from some common misconceptions held by novice software developers. One is that creating software is a relatively simple process: all you need to do is sit down, write out a few ideas, and begin programming. Others tend to bite off more than they can chew. Most new software developers tend to underestimate the time and effort involved in creating just one hour of instruction and thus do not fully utilize the potential of computer-aided instruction (CAI). The purpose of this paper will, therefore, be to examine the development of CAI, to explore what is involved in creating software for language learning, and to make some suggestions for software development.

One of the major selling points of CAI is that it is individualized instruction. Suppes states that one benefit of using computers for instruction is "the sense of individualization that can be achieved by computer-assisted instruction, both in terms of actual rate of progress of the student and also in terms of the convenience of time and place for the student" (cited in Taylor, 1980, p. 19); however, just how individualized is current CAI? In the past twenty years researchers have identified three dimensions of learning styles or "preferences." These styles include cognitive style, information processing habits representing the learner's typical mode of perceiving, thinking, problem solving, and remembering (Messick, 1976); affective style, the learner's typical mode of arousing, directing, and sustaining behavior; and physiological style, biological based modes of response that are founded on gender-related differences, personal nutrition and health, and accustomed reaction to the physical environment (Keefe, 1979).

References

Chaudron, C. (1985). "Intake: On models and methods for discovering learners' processing of input." **Studies in Second Language Acquisition, 7,** 1-14.

Clark, R. E. (1983). "Reconsidering research on learning from the media." **Review of Educational Research, 53,** 445-459.

de Bot, K., & Mailfert, K. (1982). "The teaching of intonation: Fundamental research and classroom applications." **TESOL Quarterly, 16,** 71-77.

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

Krashen, S. (1981). **Second language acquisition and second language learning.** Oxford: Pergamon Press.

Pennington, M. C. (1984). "Review of ALA/regents grammar mastery Series." **CALL-IS Newsletter 1,** 1,3.

Richards, J. C. (1986). "Focus on the learner." **Working Papers 5, University of Hawaii at Manoa: Department of English as a Second Language,** 61-86.

Stevick, E. W. (1976). **Memory meaning and method: Some psychological perspectives on language learning.** Rowley, MA: Newbury House Publishers.

SOFTWARE REVIEWS

CLUE IN (Regents/ALA) and **THE GAME SHOW** (Advanced Ideas)

At the 1984 TESOL Convention, Ferreira, Sklar, and Kagan gave a presentation on adapting commercial software for language learning. This was my first introduction to **The Game Show,** and since then, Regents/ALA has come out with **Clue In,** which, though designed for language learning, adopted essentially the same concept.

Both programs are games in which questions are put to students and clues given, one by one, to prompt the answers. Both come with starter games, but both are really meant to be adapted to individual situations, and accordingly, both include authoring features that allow for easy input of clues and responses.

The games differ in details of format. **The Game Show** closely emulates the old TV game show. A moderator and two contestant interrogators are shown on the screen and converse in cartoon bubbles (one person can play alone). The moderator asks the contestants' (students') names and then prompts a category. When this is selected, one interrogator presents a statement to which the first student partner must type a response. If that student misses, then the other cartoon contestant provides a second clue to which the second student attempts an answer. The game proceeds until someone guesses the word or phrase the computer is looking for or until the clues are exhausted. At the end of the game, the player with the most points wins.

This format differs slightly from that of **Clue In.** First, **Clue In** is played by one lone player either at his/her own pace or against the clock. The latter mode allows more points, which is useful if the player wishes his or her score registered in the Hall of Fame, where the three highest scores for that disk are kept permanently for conspicuous display. Second, players select a correct answer from three possibilities. Selecting a correct answer eliminates frustration over insignificant errors in a typed-out response (along with opportunities for active production), and which narrows the response set (which could be anything up to 36 characters in **The Game Show).**

In **Clue In,** the player has only two chances to guess the correct answer. This is done as follows: on being given the first clue, the player is challenged to attempt an answer. If the challenge is accepted, then the three possible answers are displayed, but these may be ignored. The point value is then decreased, a second clue is displayed, and

the challenge is repeated. If the player accepts the challenge at this point, he/she will likely find that one of the possible answers is no longer logical, but that it is still a matter of guessing between the remaining two. Declining the challenge until the third clue is the safest strategy since that clue should distinguish the correct answer from the two incorrect ones.

For adults, the games are initially enjoyable, but both suffer from two disadvantages: (1) question presentation is strictly linear, and (2) the quality of any game (hence motivation for playing it) depends on the thought that has gone into preparing the clues. The first of these, strict linearity, means that the games are both programmed so that for any given category, play will proceed in exactly the same way each time, with the same questions and corresponding clues presented in identical order. The ability to achieve seemingly infinite variety through random selection from a copious database is one advantage distinguishing computers from other instructional media, but neither of these games has this feature. Adult learners find that the novelty of the presentations quickly fades once the game proves to be predictable. The success of these programs, therefore, depends on the ability of teacherss (or the students themselves) to come up with numerous variations.

On the other hand, my observations of children using these games suggest that youngsters appreciate the security of predictable linearity. Young learners, who find the formats of these games highly motivating, enjoy the advantage over the computer that knowing the answer gives them, and these learners will play one game over and over if they like the subject matter.

As pointed out by Ferreira and her colleagues, such games are best utilized in ESL when students themselves prepare the questions. Either of these games would lend itself well to this purpose. The authoring programs are transparently easy to use, and the games can easily be created, eliminated, or changed later. Such an activity would likely cover several days or class periods. At least one period would have to be devoted to preparing appropriate clues, and perhaps another to collaborating on which clues to use. Another period would be devoted to input, and still another to reaping the reward of playing each other's games.

In summary, I recommend both programs, subject to the constraints mentioned above. Both are professionally presented and are particularly motivating for children. With adults, their effectiveness will be attenuated by repetition unless fresh exercises are constantly introduced via the authoring system. One imaginative way to use the programs is to have the students themselves create and key in the lessons and to try them out on one another. Used in this way, these programs would contribute to a student-centered, collaborative setting for learning.

Reference

Ferreira, L., Sklar, S., & Kagan, A. (1984, March). **Computers as realia: Using existing software to develop ESL communication skills.** Paper presented at the 18th annual Teachers of English to Speakers of Other Languages' Convention, Houston, TX.

Reviewed by **Vance Stevens**
Sultan Qaboos University

ESL Picture Grammar

Product Description
Program Title: ESL Picture Grammar
System Requirements: Apple II+,IIe,//c, 48K, one disk drive, color or mono-chrome monitor
Publisher: Gessler Educational Software, 900 Broadway, New York, NY 10003
Price: $59.95
Content: One diskette, teacher's guide
Audience: Elementary School Age.
ESL/EFL Level: Beginning to Intermediate. The materials are designed to stand on their own.
Summary

ESL Picture Grammar is a primitive piece of software which suffers from a very restricted vocabulary range and has limited pedagogical value. It requires the user to construct contrived and nonsensical sentences following the traditional grammatical paradigm.

ESL Picture Grammar is a drill-and-practice program which gives practice in producing simple affirmative/negative statements and questions in the present, past, and future verb tenses. The formation of statements is framed in the traditional grammar paradigm of verb parsing **a la** I am kick-