R E P O R T   R E S U M E S

ED 015 679   56   EM 005 974

BRANCHING PROGRAMS IN AUTOMATED INSTRUCTION--A SIMPLIFIED FORMAT.

BY- ROE, ARNOLD
CALIFORNIA UNIV., LOS ANGELES

REPORT NUMBER R-61-72   PUB DATE   AUG 61
GRANT   OEG-7-04-138-01
EDRS PRICE   MF-$0.25   HC-$0.72   16P.

DESCRIPTORS- *PROGRAMED INSTRUCTION, *BRANCHING, *EVALUATION, *PACING, *TIME FACTORS (LEARNING)

THIS STUDY DISCUSSES DIFFERENT PROGRAM BRANCHING PROCEDURES AND DESCRIBES AN INEXPENSIVE CARDBOARD DEVICE TO EVALUATE STUDENT PERFORMANCE ON PROGRAMS VARYING THE TECHNIQUES. (LH)

EM005 974

NDEA VII
PA 56

A

ED015679

AUGUST 1961

DEPARTMENT OF ENGINEERING

branching programs in automated instruction: a simplified format

REPORT NO. 61-72

UNIVERSITY OF CALIFORNIA, LOS ANGELES

Report No. 61-72
August 1961

BRANCHING PROGRAMS IN AUTOMATED INSTRUCTION:

A SIMPLIFIED FORMAT

Arnold Roe

TEACHING SYSTEMS RESEARCH PROJECT
H. W. Case A. Roe

University of California
Department of Engineering
Los Angeles, California

FOREWORD

The research described in this report, *Branching Programs in Automated Instructions: A Simplified Format,* by Arnold Roe was s supported in part by a grant from the United States Office of Education, Department of Health, Education, and Welfare under Title VII of the National Defense Education Act (NDEA Grant 7-04-138.01).

J. M. English
Director - Engineering Research

UNIVERSITY OF CALIFORNIA
DEPARTMENT OF ENGINEERING
LOS ANGELES, CALIFORNIA

In our early experimental work[1, 2, 3] we were interested in discovering which method of presenting programmed instruction would be most effective. We compared the use of multiple choice machines, free-response machines, programmed texts with overt responses, and programmed texts with no responses. Linear programs, where each student had to go through all the items in a predetermined sequence, were used in these early studies. We failed to find any significant differences in the terminal performance of the students using the various learning methods. However, we did find significant differences in the amount of time required to complete the various kinds of learning sessions.

We thus discovered, as most experimenters in the educational process eventually do, that it is difficult to find both statistically and socially significant differences in students' performance scores (particularly after some time has elapsed since the learning session) which can be attributed to the different methods of instruction under experimental investigation. The failure to find such significant differences may in part be attributable to the weak and indeterminate measuring scales we use in the human behavioral sciences.[4]

This same difficulty is not encountered in measuring the amount of time a student takes in studying specified course material, and here significant differences have been experimentally determined. The question of interest then becomes: What method of instruction will require least student time without jeopardizing his (weakly measurable) performance test scores?

Using linear programs, we had already found significant differences in learning time, apparently determined by the time required for the physical manipulation of the learning material or devices used to display the material. However, linear programs did not seem to be ideally suited to take advantage of the learning time variable.

If the same linear program is used for all students, one way to insure that the greatest number will have high terminal performance is to write the program at the slowest student's level, so that he will have adequate repetition and practice opportunities. However, a program fixed at the slow student's level annoys and eventually bores the bright student because of the great number of small items he is forced to go through. It has been claimed that individual differences are taken care of by the possibility for bright students to move at a faster pace through the linear program than dull students. This is simply not true. A certain amount of time is required just to read and respond to each item. Prior knowledge of the subject matter, or aptitude, does not save much time. We felt that significant savings in time could only be achieved by some branching technique, especially in the early stages of a program where an evaluation should be made of the prior knowledge and other abilities that a student brings to the learning situation.

But which branching technique should we use. Many kinds of branching procedures have been suggested: backward branching to missed items, backward branching to review an entire sequence of items, backward branching to alternate form items, lateral branching to supplemental or prerequisite material, lateral branching to supplemental practice items, branching down to a lower level or more detailed items for slow students, branching up to a faster program for bright students, and finally, forward branching by skipping items.

Some of these branching procedures, in the way they have been diagrammatically illustrated, seemed to require of the user a digital computer oriented background (see Figures 1,* 2,** and 3**).

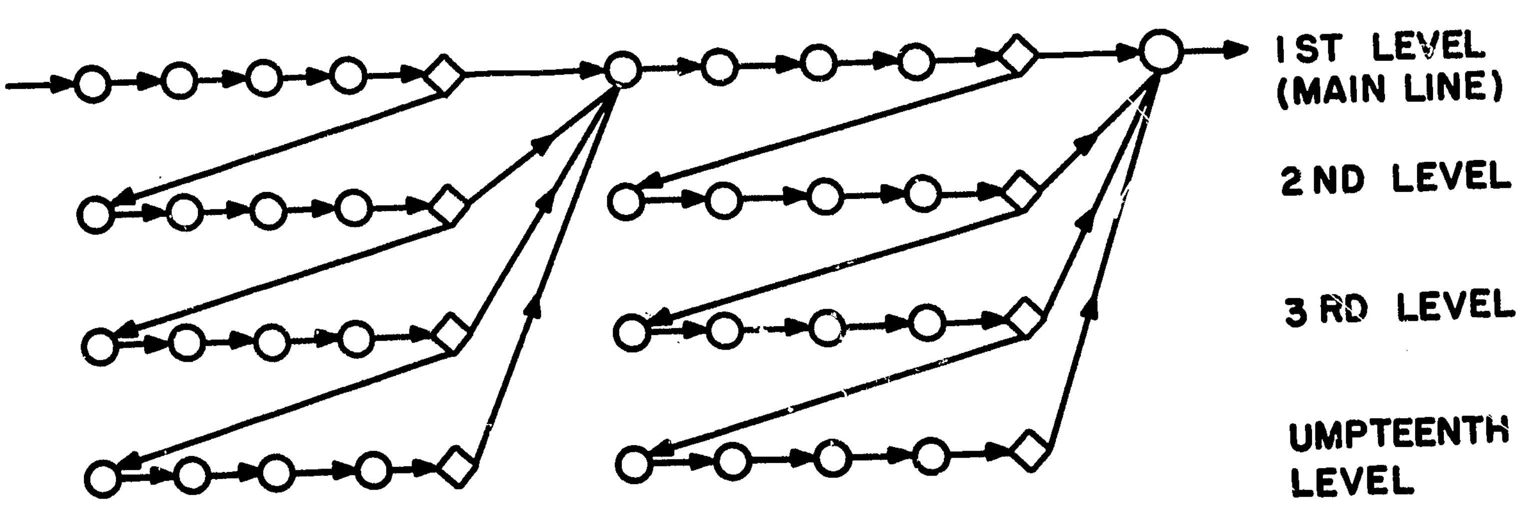


FIGURE 1

*Suggested by John Coulsen and Harry Silberman of Systems Development Corp.
**Suggested by Norman Crowder of United States Industries.

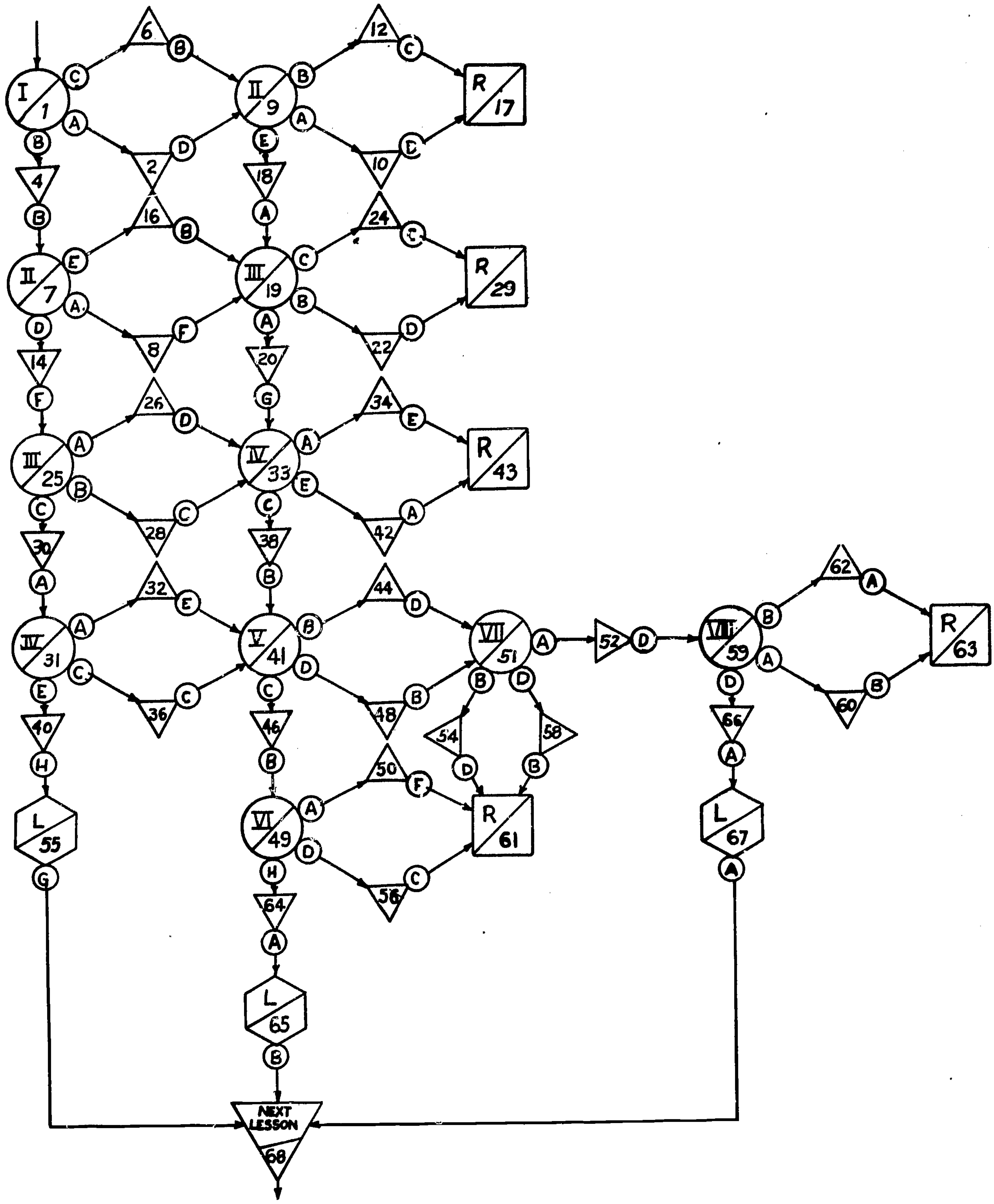

I 1
II 9
R 17
II 7
III 19
R 29
III 25
IV 33
R 43
IV 31
V 41
VII 51
52
VIII 59
R 63
VI 49
R 61
L 55
L 65
L 67
NEXT LESSON 68

FIGURE 2

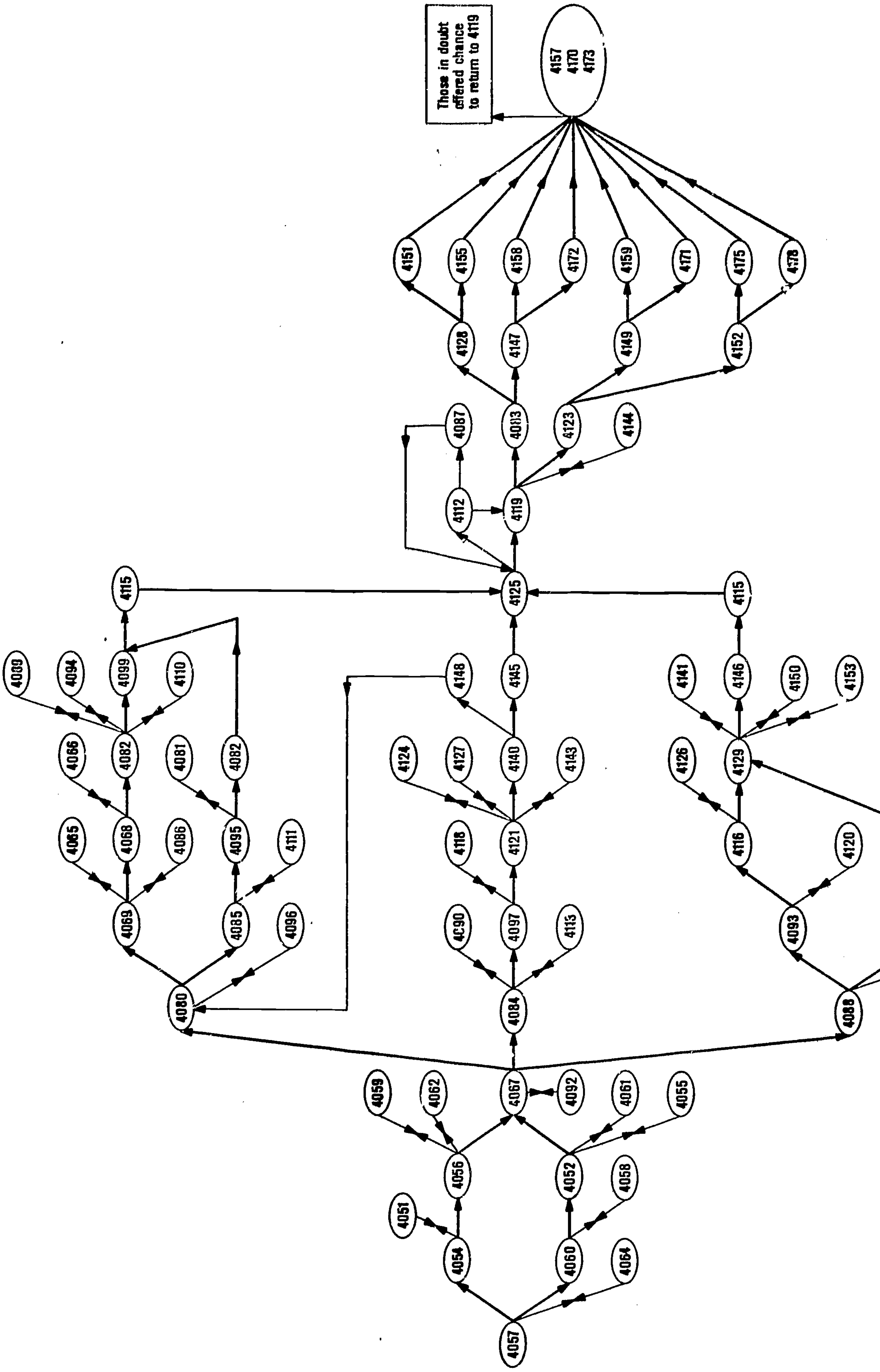
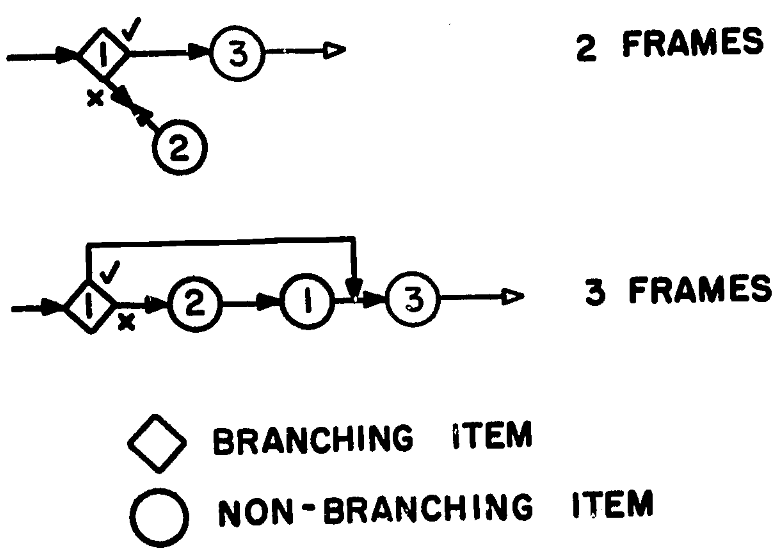

Those in doubt offered chance to return to 4119
4157
4170
4173

FIGURE 3

The complexity of these diagrams, and the belief that rather expensive equipment for random access to program items is required, apparently has dissuaded many investigators from exploring the potentialities of branching programs.

However, it is possible to look at branching programs in a slightly different way, and by so doing, reduce all types of branching programs to one procedure, namely, forward branching. Furthermore, the items in such a forward branching program can be strung out in a line, similar to a linear program, making it evident that random access devices are not required for branching programs.

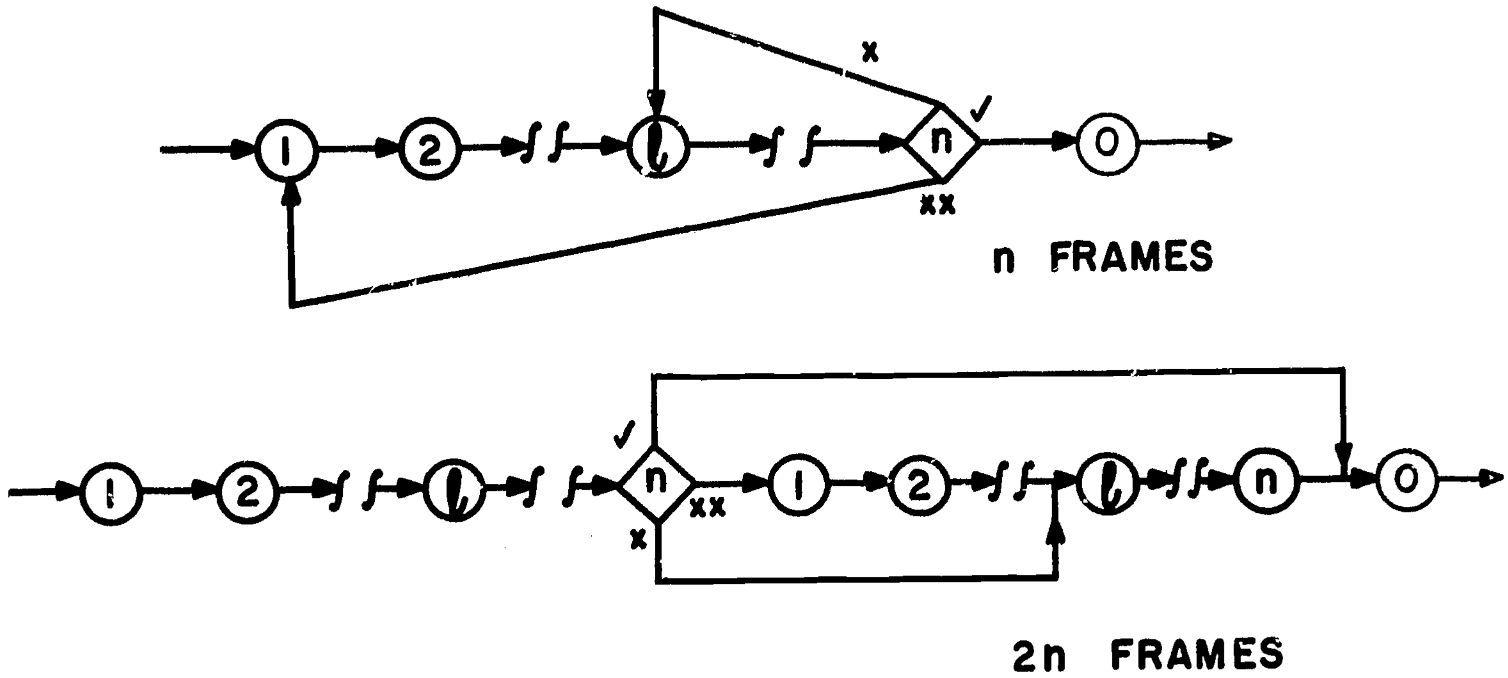
For example, let us look at the "return item", of the type profusely illustrated in Figure 3. Here, if a student makes an incorrect response on item 1 (see upper diagram, Figure 4), he is branched to a remedial item 2, from which he must return to, and repeat, item 1. This "return" type of branching program can be reduced to a linear sequence, with a forward branching (or skipping) provision if the information on the branching item is repeated once (see lower diagram, Figure 4).



RETURN ITEM

FIGURE 4

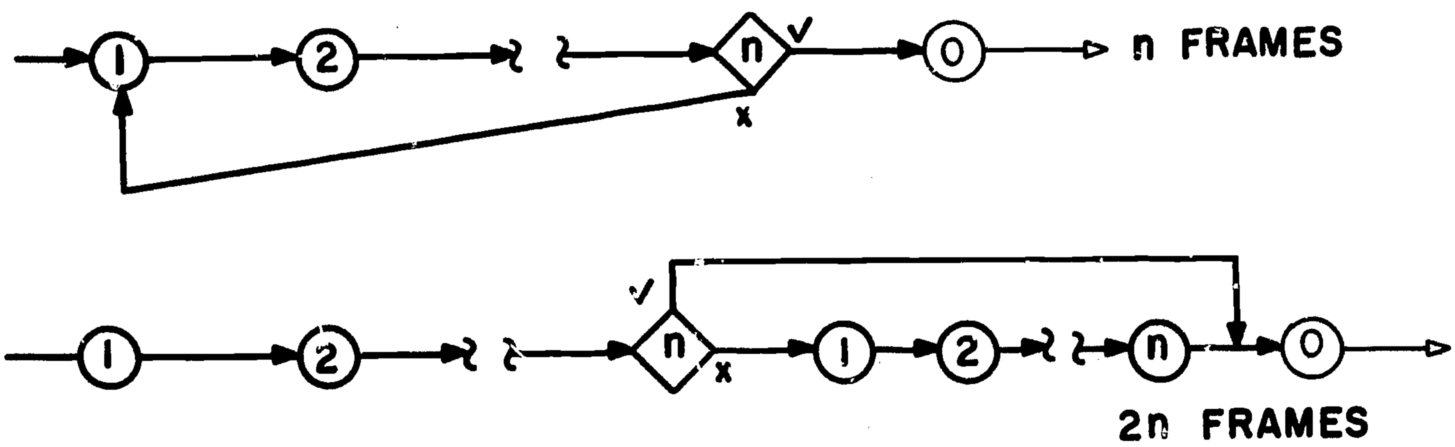
Similarly, for a backward loop (see Figure 5), a repetition of the items will permit transforming a backward branching procedure into a forward branching format.



BACKWARD LOOP

FIGURE 5

For selective backward loops, where the branching items are used to judge the severity of an incorrect response and thus determine how far back in the program the student should be sent, a repetition of the items and selective forward branching again reduces the program to the required format. (See Figure 6.)



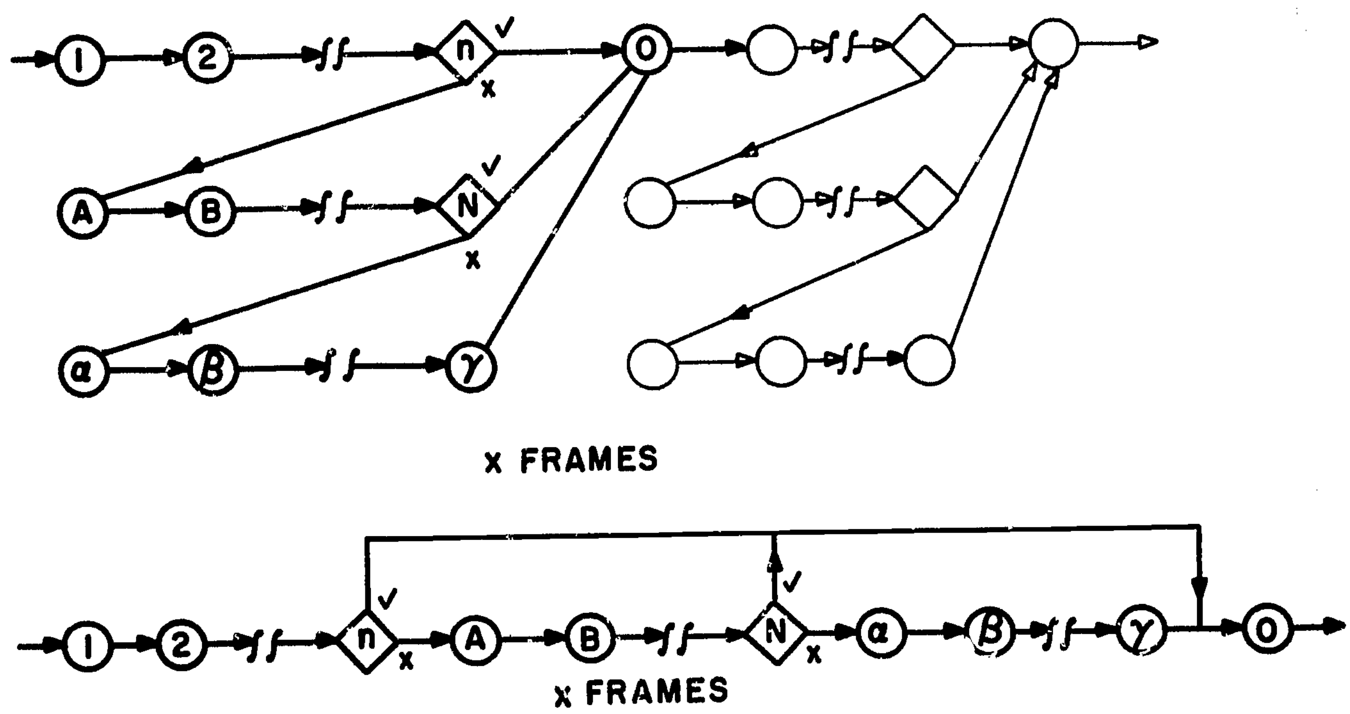
SELECTIVE BACKWARD LOOP

FIGURE 6

In converting the various types of backward branching programs into forward branching programs, as illustrated above, two assumptions are made. Firstly, that it is more economical to physically repeat items than to re-display them by means of a random access device or a forward-backward counting device. Secondly, that the repetition of the same item is not required more than one time. This second requirement is made from a practical point of view, since theoretically it is possible to allow multiple repetition of the same items by duplicating the items as many times as desired. In the types of backward looping illustrated in Figure 3, and in the upper diagrams of Figures 4, 5 and 6 it is possible to endlessly traverse the same loop, if you cannot successfully master the branching item controlling that loop. For most non-rote type of learning tasks it seems silly to ask a student to repeat an item more than once. If the second time around has not helped him, it is doubtful whether additional repetitions will be effective.

In general, all types of backward branching programs are subject to the criticism that repetition of items is not as effective as branching to alternate items. If this criticism is valid,[5] then the procedure recommended here for arranging branching programs becomes more attractive, since the physical repetition of the items which is necessary for backward branching programs is not required for other types of branching.

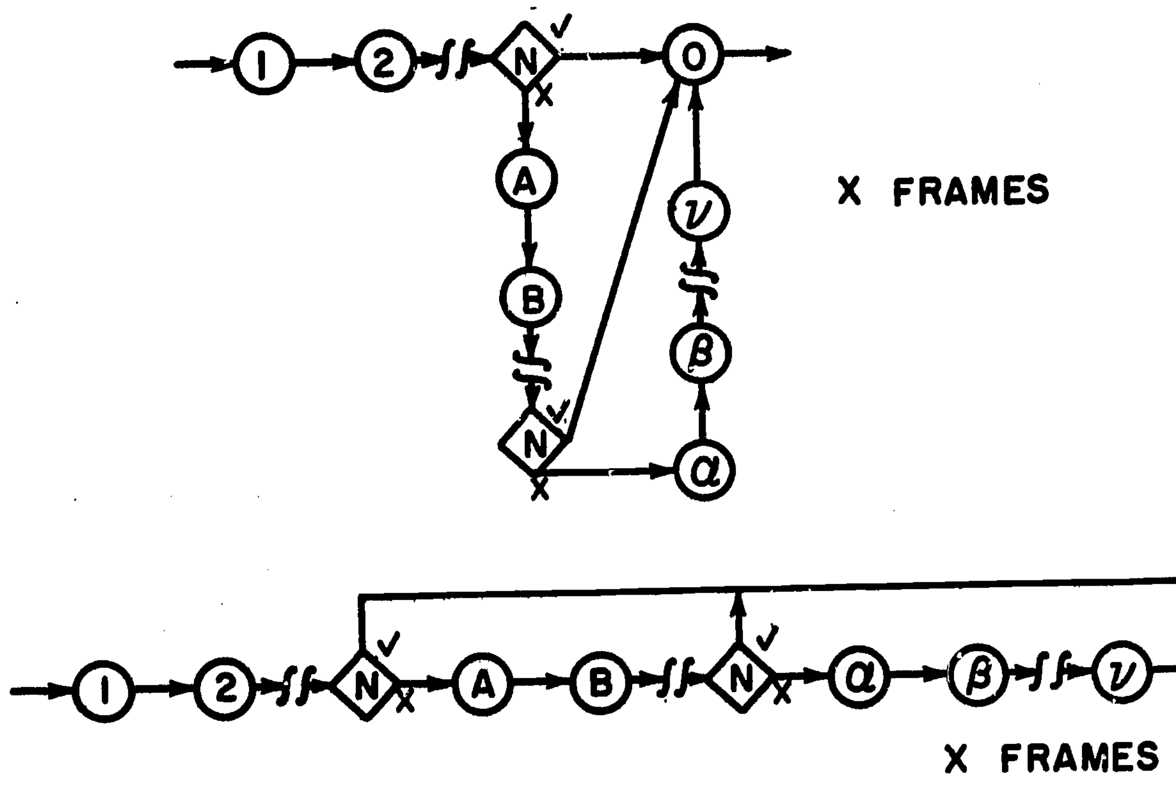
In the multiple level branching procedure, instead of being sent back thru the same items after making an incorrect response on a branching item, the student is sent to alternate items. As shown in Figure 7, this multiple-level program can readily be transformed into a linearized forward branching program, and the number of item frames is identical in both formats.



BRANCHING TO LOWER LEVELS

FIGURE 7

It is interesting to note that programs containing so-called "remedial loops", as illustrated in the upper diagram of Figure 8, are identical to multiple-level programs shown in the upper diagram of Figure 7. This becomes apparent when comparing the lower diagrams of both Figures 7 and 8.



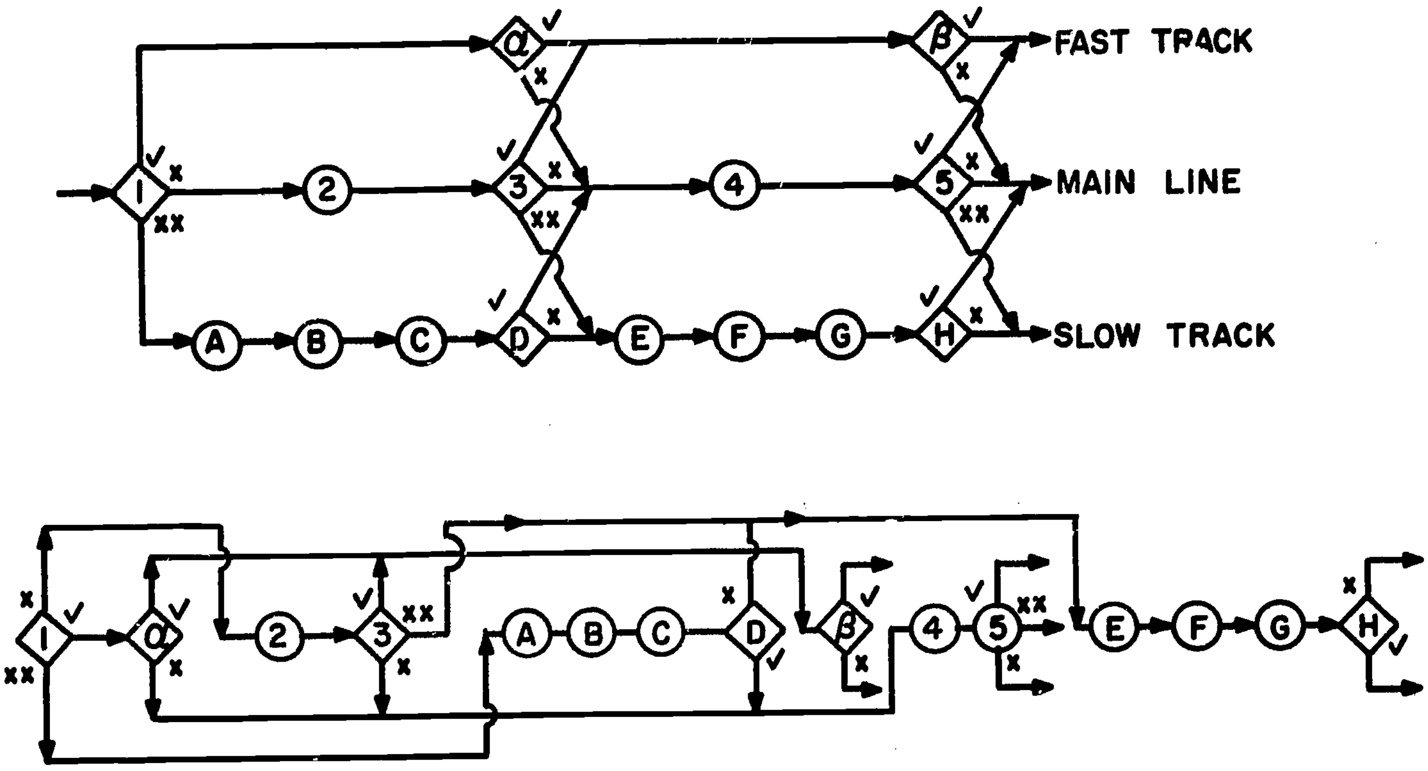
REMEDIAL LOOP

FIGURE 8

Another interesting multiple-level branching procedure is the "fast-slow track" program, shown in the upper diagram of Figure 9. Depending on the student's prior performance, he is directed to a version of the program having either more or less items covering a given concept. If all the items are strung out in a line, the forward branching paths could become fairly complex, but it does indicate that this type of program can also be handled in a simple forward branching device.

Instead of using the student's response on a single branching item to determine which level in a multiple-level program he should be sent to, two or more test items may be used, as shown in the multiple-level diagram of Figure 10.* Also, the use of more than one test item to control forward branching** seems particularly appropriate where there is a possibility that the student may guess the correct answer and thereby skip over a sequence of items containing information which he does not know. The use of more than one test item to determine the branching path is also reducible to the linearized forward branching format, as shown in the bottom diagram of Figure 10.

* Suggested by Jacques Gilly of Systems Development Corporation.
**Suggested by Richard Nazro, formerly with the Teaching Systems Research Project, Department of Engineering, UCLA.



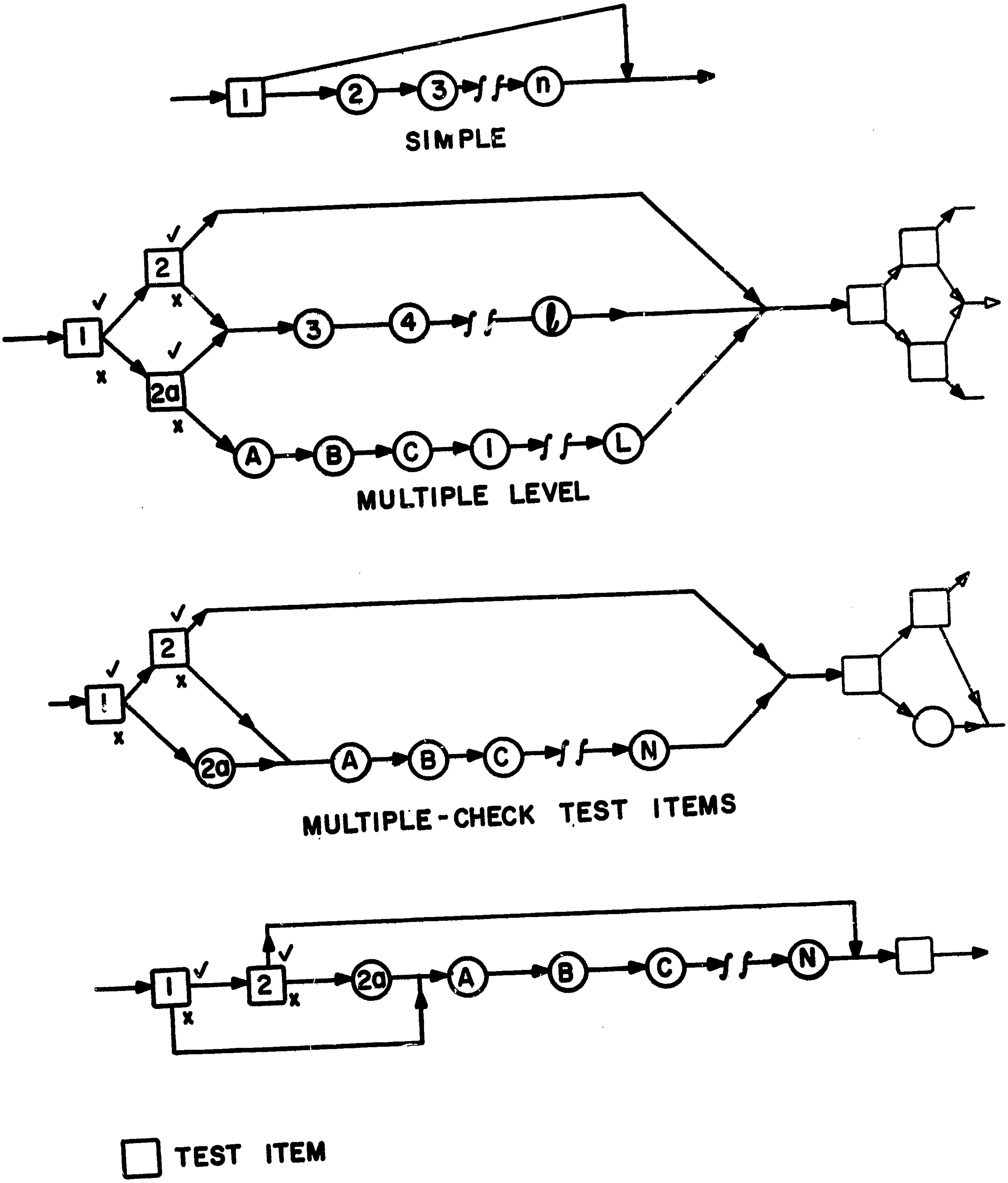
FAST-SLOW TRACKS

FIGURE 9

One might observe from the foregoing illustrations, and from Figure 11, that not only is it possible to string any sequence of branching program items into a linear format, but it is also possible to convolute a linear program into a branching format. Presumably there are no a priori reasons for including an item in the main line of a program rather than in a remedial loop, or vice versa. The final judgment on the number and arrangement of items can only come from testing the various possibilities on students.

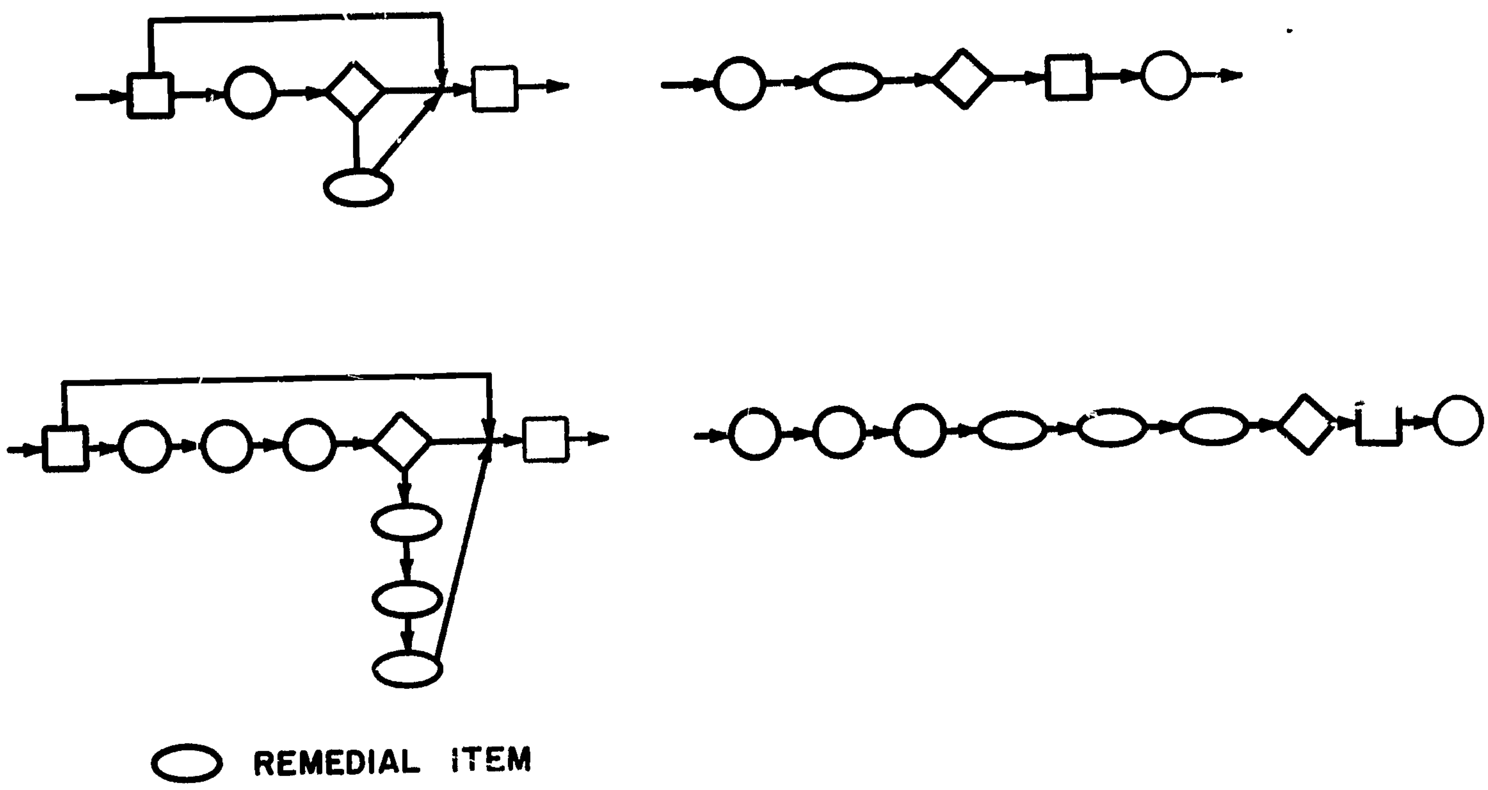
Using the concepts outlined above, it was possible to construct a very simple and inexpensive cardboard device* for evaluating student performance with various nominally different branching techniques.[6] (See Figure 12.) Since all the different branching techniques can be reduced to one format, the experimental conditions are identical for all the branching techniques. The device also permits quick and inexpensive changes in the program and provides a record of the student's response to each item, and a record of the individual path he followed through the program.

* Developed from suggestions by: the author, S. Pressey, J. Rigney, Y. Filby, R. Powell, F. Saunders, K. Vlachouli and R. Nazro.



FORWARD LOOPS

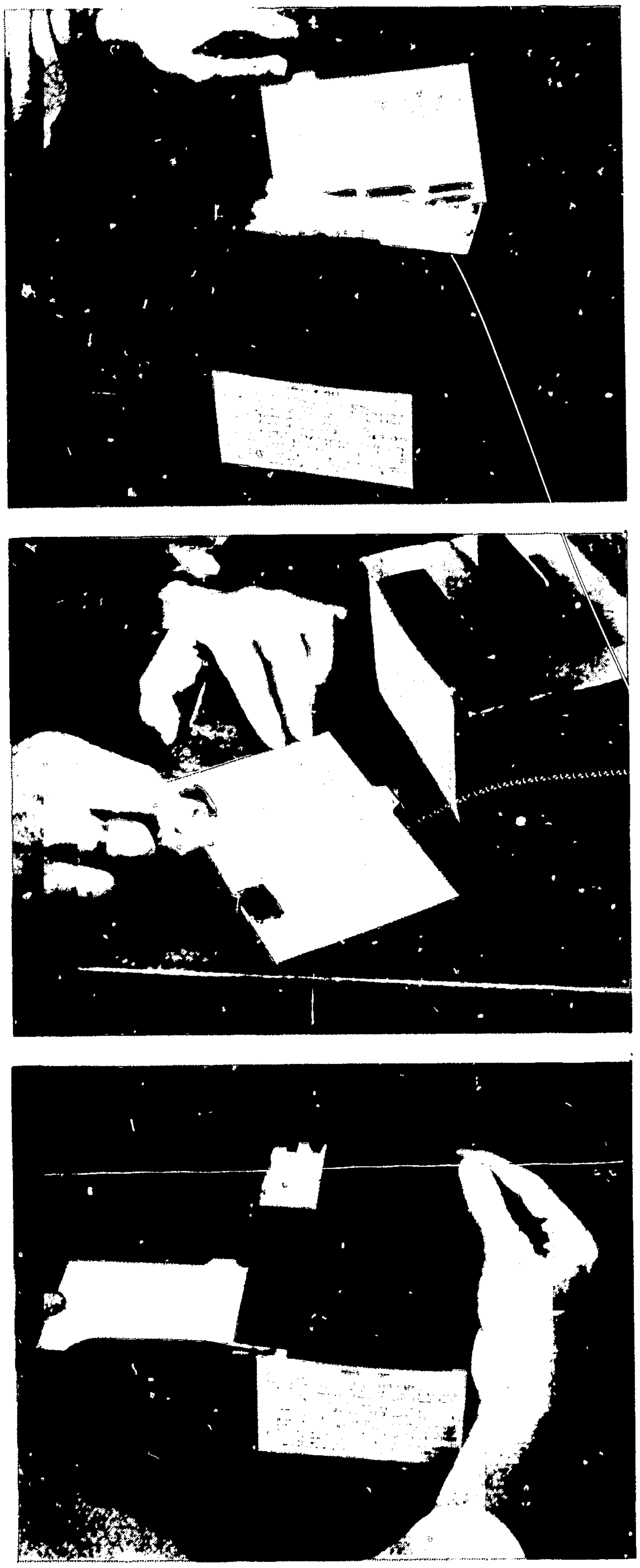
FIGURE 10



NUMBER AND ARRANGEMENT OF ITEMS

FIGURE 11

Slightly more sophisticated devices, using the linearized forward branching arrangement, can be constructed to automatically present the items to the student. The important thing to remember is that laying out and using a branching program need not be a very complicated or expensive procedure. The sooner this becomes evident to more researchers, then the sooner we may find teaching procedures and specific teaching programs which will shorten the time required by students for the completion of course material.



BRANCHING DEVICE. Learning items on numbered and tabbed 4 x 6-inch cards. The box is constructed and the cards are arranged to prevent the student from browsing and from recognizing the branching procedure.

FIGURE 12

REFERENCES

1. Roe, A., Bishop, J., Kirk, W., Massey, M., Moon, H., Tarter, M., and Weltman, G., A pilot study-automated learning research project. Los Angeles: University of California, Department of Engineering, Report No. 60-53, May, 1960.

2. Roe, A., Massey, M., Weltman, G., and Leeds, D., Automated teaching methods using linear programs. Los Angeles: University of California, Department of Engineering, Report No. 60-105, December, 1960.

3. Roe, K.V., Scrambled vs. ordered sequence in auto-instructional programs. Los Angeles: University of California, Department of Engineering, Report No. 61-48, August, 1961.

4. Roe, A., Research in programmed learning, presented at the ONR sponsored conference on Application of Digital Computers to Automated Instruction, Washington, D.C., October 10, 1961.

5. Roe, A. Automated teaching methods using branching programs, Los Angeles: University of California, Department of Engineering, Report No. 61-74, September, 1961.

6. Roe, A., A simple device for experimental use of branching programs, Los Angeles: University of California, Department of Engineering, Report No. 61-73, October, 1961.