Comprehension as affected by structure of problem representation

RICHARD E. MAYER
University of California, Santa Barbara, Santa Barbara, California 93106

A problem-like branching system describing what prizes (A through F) were awarded for particular outcomes of a tournament of games among three teams was presented to 200 subjects as either a verbal list with "go to" structure (Jump), a shortened verbal list (Short-Jump), nested verbal paragraphs with "if ... then ... else" structure (Nest), a matrix table (Example), or as diagrammatic representations of each of these. In tests of comprehension, the overall performance increased from lowest to highest as follows: Jump < Short-Jump ≃ Nest < Example, and this order was particularly strong for performance on complex questions relative to less complex questions. Jump and Short-Jump performance was relatively higher with diagrams and Example was lower with diagrams. Implications for a theory of problem representation and for development of computer programming languages were discussed.

Much research seems to indicate that how a problem is organized and represented influences the difficulty of problem solving. In studies of traditional problem tasks, the performance of subjects was affected by presenting a problem in different contexts, e.g., the Missionaries and Cannibals problem as a problem about "jealous husbands" (Reed, Ernst, & Banerji, 1974) or the Tower of Hanoi problem as a problem about "space monsters" holding various sizes of globes (Hayes, 1975). Similarly, Schwartz (Polich & Schwartz, 1975; Schwartz, 1971) found overall differences in time and accuracy of solution on a complex deduction task depending on whether the information was represented by the subject as a list of facts, a matrix, a directed graph, a diagram, etc. Based on evidence from syllogistic reasoning, Revlis (Note 4) has argued that the stage of encoding the problem information can be influenced by several factors, including the concreteness/abstractness of the premises; in addition, "errors" in reasoning may often arise from correct inferences based on different internal representations of the problem.

Complementary attempts by educational psychologists to determine the important structural variables in representation of algebra or arithmetic story problems which influence problem solving performance have produced a list of factors, including number of operations, organization, whether a picture is included (Loftus & Suppes, 1972; Paige &

Simon, 1966; Steffe, Note 6). Another promising area that also has important practical implications is the study of computer programming languages, and especially how the organizational conventions of language affect the problem solving efficiencies of novice programmers. Preliminary studies now appearing in the literature have been directed at testing whether one method of representation is better overall than another. For example, Sime, Green, and Guest (1973) taught nonprogrammers a micro-language using a listable "branch to label" conditional structure (JUMP) or a nestable "if ... then ... else" conditional structure (NEST) and found that the latter was easier to use in writing "programs." Similarly, Reisner, Boyce, and Chamberlin (Note 3) found that nonprogrammers could more easily learn a data management language based on a fixed, almost diagrammatic format (SEQUEL) than a more traditional system (SQUARE).

Such findings are encouraging for the idea that different ways of structuring and representing problems influence the problem solving performance in an overall or quantitative manner. The present experiments attempted to extend these findings to an area that also has implications for the development of new programming languages (as suggested by Miller, Note 2), and which provides new information on what aspects of interpretation (rather than composition) performance are influenced by various ways of structuring the statement of a problem.

Greeno (1973) has suggested that problem solving involves accessing and transferring information from short-term memory and long-term memory to "working memory" and combining or restructuring that information in "working memory" in order to construct the solution. If better structured presentation provides a general aid, such as faster accessing of information, to working memory, then

well-structured problem representations should result in better performance on all types of questions; if, instead, better structured problem representations allow a subject to "think about" more information in his limited working memory by allowing information to be integrated in effectively larger "chunks," then superior performance would be expected only on lengthy or complicated problems requiring a lot of information in working memory.

EXPERIMENT I

Method

Subjects and Design. The subjects were 96 Indiana University students who participated in the experiment in order to fulfill a requirement for their introductory psychology course. Only subjects who had had no prior experience with computer programming and who could successfully solve three or more of five algebra substitution problems were included. Twelve subjects were assigned to each cell of a 4 by 2 factorial design, with the first factor being the organization of the problem representation card (Jump, Nest, Short-Jump, Example) and the second factor being format of the problem representation card (Verbal, Flow Chart). Since each subject solved eight sets of questions, comparisons by type of question are within-subject comparisons.

Materials. Eight problem representation cards were constructed on 5 x 8 in. cards, each representing the prizes obtained (Prize A, B, C, D, E, or F) for various outcomes of a tournament among three teams (Michigan, Indiana, and Ohio). The representations varied organization: Jump consisted of a list of propositions, with conditionals branched to labels similar to the "if ... go to..." structure in programming; Short-Jump modified the list to shorten the jumps; Nest consisted of a paragraph with nested, more compact, "if ... then ... else" structure; Example consisted of a contingency table with an outcome on the same line as all necessary conditions. The four representations were presented in the format of words (Verbal) or as diagrams (Flow). All eight cards assigned the same prizes to the same conditions, but differed in organization and format of the representation as shown in the Appendix.

Eight sets of question sheets were generated based on the following forms: (a) If team 1 defeats team 2, then could you possibly win prize X? (b) If team 1 defeats team 2, which prizes could you possibly win? (c) If team 1 is the best (or worst) team, which prizes could you possibly win? (d) If team 1 is not the best (or worst) team, which prizes could you possibly win? (e) If you won prize X, then is it possible that team 1 defeated team 2? (f) If you won prize X, what is the maximum number of games that team 1 could have won? (g) If you won prize X, which is the best (or worst) team? (h) If you won prize X or prize Y, which team cannot possibly be best (or worst)? There were six questions on the sheets for types a and e, and four questions for all other types. Questions a, b, c, and d were termed "forward" questions because they gave information about games and asked about prizes, and e, f, g, and h were called "backward" because they gave information about prizes and asked about games. In addition, the questions were designed to involve increasing amounts of complexity from a to d and from e to h.

Additional materials included a subject questionnaire containing questions about the subject's mathematics and computer science background, and a pretest consisting of five algebra substitution problems.

Procedure. Subjects were tested in small groups of two to four per session, with subjects randomly assigned to treatments within sessions. First, instructions were read, then the subjects were each given one of the eight problem representation cards and the first sheet of questions. When the subject handed in sheet 1, he was given sheet 2 and so forth for all eight question sheets. Subjects could work at their own rates but could not go back to work on previous sheets. Subjects retained the same problem representation card for all question sheets. The order of presentation of sheets was counterbalanced, determined by a Latin square. The experimenter recorded the time spent on each question sheet.

Results

Since error rates were high, the analysis focused on proportion correct. If a question had several parts, such as "List all the prizes possible," all parts had to be correctly answered (i.e., all prizes listed with no false positives) in order to be marked correct. Table 1 gives the overall proportion correct response for the eight groups, with the difference between proportion correct on questions (e + f + g + h) minus (a + b + c + d) shown in parentheses.

One question concerns the role of verbal vs. flow chart problem representation. Flow charts have been used as a pedagogic aid, although there is an increasing literature indicating that flow charts do not always aid in the comprehension, debugging, modification, or generation of programs (Mayer, Note 1). In the present experiment, the flow chart format resulted in better performance for the Jump and Nest groups, but the verbal format resulted in better performance for the Example group [Organization by Format interaction, $F(3,88) = 4.07$, $p < .025$]. The Jump and Nest organizations were the most complex—with subjects probably not able to immediately grasp, in one "chunk," the overall structure of the situation—and the Short-Jump and Example organizations were more compact, allowing for subjects to more easily determine the situation structure. Apparently, presenting a problem in terms of a flow diagram allowed the former two groups to more readily integrate the complex verbal list into an understandable structure, while the latter two organizations were already comprehensible in verbal form. These results are consistent with the previous finding that diagrammatic problem representation results in superior problem solving performance over verbal representation (Schwartz, 1971); however, the present experiments extend this finding to a new area (i.e., computer programming-like problem specifications) and indicate some limits of this finding (i.e.,

Table 1
Overall Proportion Correct Response for Eight Treatment Groups

Format of Problem Specification	Organization of Problem Specification			
	Jump	Nest	Short Jump	Example
Verbal	.53 (−.04)	.51 (+.10)	.61 (+.03)	.82 (+.02)
Flow	.59 (+.06)	.69 (+.05)	.62 (+.02)	.73 (+.12)

Note—Number in parentheses is difference between proportion correct on backward questions (e, f, g, h) minus proportion correct on forward questions (a, b, c, d). Approximate standard error for each cell is .05.

Table 2
Proportion Correct Response for Four Presentation Organizations by Type of Question

Organization of Problem Specification	Forward Question				Backward Question			
	a	b	c	d	e	f	g	h
Jump	.83	.56	.24	.44	.82	.25	.67	.42
Nest	.81	.55	.31	.46	.79	.35	.76	.59
Short Jump	.85	.56	.43	.45	.88	.32	.75	.46
Example	.96	.83	.52	.61	.94	.83	.93	.43

Note—Approximate standard error for each cell is .08.

diagrams do not aid "understandable" verbal representations).

There was also a pattern in which the Verbal/Jump group excelled on forward relative to backward questions, while all other groups showed the reverse trend. Since Jump organization provided the most complex or broken "trail" between outcome and conditions, it seems that expression as a flow chart might aid mainly on complex "backward" questions that require tracing from outcome to conditions. However, since the Organization by Format by Direction interaction did not reach statistical reliability [$F(3,88) = 1.81$, $p < .20$], there is strong evidence only that flow charts affect overall performance, such as general accessing of information.

A second question is whether the particular way of organizing how a problem situation is presented will influence problem solving performance, and Table 2 presents the performance of the four organization groups by type of question. There were overall differences among the groups [$F(1,88) = 13.74$, $p < .001$], reflecting the overall poor performance for the Jump subjects and the overall superior performance for the Example subjects. In addition, the four groups produced reliably different patterns on problems of different complexity, a + e, b + f, c + g, and d + h [$F(9,264) = 6.44$, $p < .001$] and by all eight question types [$F(9,264) = 3.16$, $p < .005$], suggesting the conclusion that the advantage of getting better organized presentation shows up mainly in complex problems. One hypothesis is that better integrated ways of presenting the same information aid subjects in integrating or "chunking" masses of information in "working memory" rather than providing a general aid in locating and transferring information to "working memory." For example, the Jump group performed well—better or about the same as the Nest group—on the easier questions, such as a, b, and e, but performed much worse on the very difficult questions requiring interpretation, such as f, g, and h. This finding provides an extension of the Sime, Green, and Guest (1973) report, which found an overall superiority of a language similar to our Nest over one like our Jump on the task of writing (rather than interpreting) problem representations. Apparently,

Nest, by virtue of its more integrated structure, allows a more complete integration of information required in answering complex questions. In addition, the Short-Jump organization showed a general pattern similar to Nest—improved performance on the more complex questions as compared with Jump—and, perhaps for the same reason, a more integrated structure. Finally, the Example organization seemed to result in superior performance on all questions, and hence serves as a sort of control group.

EXPERIMENT II

Experiment II attempted to extend the results of Experiment I by asking "yes" or "no" questions concerning every branch in the problem space.

Method

Subjects and Design. The subjects were 104 Indiana University students recruited from the same pool as in Experiment I. The same 4 by 2 between-subjects design was used as in Experiment I, with 13 subjects per cell. Each subject made yes-no judgments on 144 statements, so comparisons across type of question are within-subject comparisons.

Materials. The same eight problem cards were used as in Experiment I. A set of 72 questions was generated using a 2 by 6 by 6 design with the factors as follows: (a) Direction: Forward questions were of the form "If team 1 defeated team 2 could you win prize X?"; backward questions were of the form "If you won prize X, is it possible that team 1 defeated team 2?" (b) Game: Using each of the six permutations of "team _____ defeats team _____." (c) Prize: Using each of the six possible prizes.

Procedure. Subjects were randomly assigned to treatments and were tested in groups of up to eight per session. Subjects were seated in individual booths containing a CRT screen and a two-button response box controlled by an IBM 1800 computer. First instructions were read, and subjects read an eight-frame sequence presented via the CRT screen explaining how to read the problem card and how to answer questions. When subjects finished their instructions, the problem card was taped to the response box in an easily seen position, and the first question appeared. When the subject pressed a button (i.e., "yes" or "no"), the next question appeared and so on. The 72 questions were randomly presented, and then randomly presented again. After each block of 36 questions, the subject was given feedback and a break of the form, "You were _____ correct on this set of questions. When you are ready for more questions just press either button below."

Results

The computer recorded response latencies and whether the answer was correct for each subject's response to each question. Since error rates were relatively high, the present analysis focused mainly on proportion correct; and since there might be some general learning required, the analyses focused on the performance on the second block of the questions.

Experiment II provided additional information concerning the role of diagrammatic representation. Table 3 is comparable to Table 1, as it shows the overall proportion correct for the eight treatment groups and, in parentheses, the difference between proportion correct on backward minus forward questions. As in Experiment I, there is reliable

Table 3
Overall Proportion Correct Response for
Eight Treatment Groups

Format of Problem Specification	Organization of Problem Specification			
	Jump	Nest	Short Jump	Example
Verbal	.70 (−.05)	.82 (−.03)	.81 (−.10)	.96 (−.00)
Flow	.82 (−.07)	.83 (−.02)	.83 (−.09)	.83 (+.01)

Note—Number in parentheses is difference between proportion correct on backward questions (e, f, g, h) minus proportion correct on forward questions (a, b, c, d). Approximate standard error for each cell is .02.

Table 4
Proportion Correct Response for Four Problem Structures by Direction and Prize Cited in Question

Structure of Problem Representation	Direction and Prize Cited in Question										
	Forward					Backward					
	F	E	D	C	B	A	F	E	D	C	B	A
Jump	.88	.86	.72	.68	.81	.79	.90	.84	.47	.50	.83	.80
Nest	.93	.91	.78	.77	.81	.83	.96	.87	.60	.63	.90	.91
Short Jump	.89	.91	.81	.79	.87	.92	.92	.88	.57	.53	.86	.86
Example	.91	.94	.84	.87	.89	.89	.95	.90	.85	.88	.90	.90

Note—Approximate standard error for each cell is .06.

Format by Organization interaction [F(3,96) = 4.08, p < .01] in which the Flow Chart format resulted in better performance for the Jump and Nest groups and in poorer performance for the Example groups. However, unlike Experiment I, there was no evidence that the format interacted with type of question, e.g., the Format by Organization by Direction of Question interaction that was hinted at in Experiment I was not present in Experiment II (F < 1). Apparently, representing problems diagrammatically aids overall in locating relevant information—if it was hard to comprehend in its verbal form—but there is little support for the idea that format influences the quality or structure of a subject's reasoning process in working memory.

Experiment II also provides interesting new information concerning the role of presentation organization on the overall efficiency and the

qualitative structure of reasoning performance. Tables 4 and 5 show the proportion correct response for the four ways of organizing the presented problem by types of question. As noted in Experiment I, there were reliable overall differences [F(3,96) = 4.92, p < .005] among the four organizations in a way that is consistent with the results of Sime, Green, and Guest (1973)—with Nest and Short-Jump better than Jump, and the Example group best of all. In addition, there is, as in Experiment I, an interesting pattern of Organization by Direction of Question interaction [F(3,96) = 8.00, p < .001], in which the Jump and Short-Jump groups performed much better on forward than on backward questions, while the difference was much less or zero for the Nest and Example organizations. Apparently, the list structure of Jump and Short-Jump with its broken paths (i.e., "go to" structure) allowed relatively good performance in one direction only (i.e., down) but the continuous paths provided in Nest and Example allowed efficient "two-way" thinking.

In addition, Table 4 displays reliable Organization by Prize of Question interaction [F(15,480) = 4.31, p < .001] and Organization by Direction by Prize interaction [F(15,480) = 3.67, p < .001]. These interactions may be summarized by stating that the Jump organization resulted in particular trouble for the questions about prizes C and D, while the Short-Jump resulted in trouble on these questions only if working on the complex backward questions. Prizes C and D are determined by only two games (thus either team could win the third game), while the other prizes are determined by specific outcomes of all three games. The added confusion of questions with prizes C or D had an effect for all groups, but a much stronger effect for the list organizations (Jump and Short-Jump); thus, these findings are comparable with those in Experiment I in that complex problems were disproportionately more difficult for the Jump and Short-Jump organization.

The Organization by Game Cited in Question interaction [F(15,480) = 6.01, p < .001] and Organization by Direction by Game interaction [F(15,480) = 3.67, p < .001] shown in Table 5 are reliable. All groups performed well on questions involving the first game mentioned—Indiana vs.

Table 5
Proportion Correct Response for Four Problem Structures by Direction and Game Cited in Question

Structure of Problem Representation	Direction and Game Cited in Question											
	Forward						Backward					
	I > M	M > I	O > M	M > O	I > O	O > I	I > M	M > I	O > M	M > O	I > O	O > I
Jump	.83	.87	.72	.86	.80	.66	.87	.85	.61	.76	.62	.63
Nest	.85	.85	.83	.86	.88	.74	.88	.88	.72	.85	.75	.79
Short Jump	.91	.89	.87	.92	.88	.71	.86	.88	.67	.88	.64	.69
Example	.88	.88	.88	.89	.88	.92	.90	.92	.90	.87	.89	.92

Note—The symbols used are I = Indiana, M = Michigan, O = Ohio, and > = defeats. Approximate standard error for each cell is .05.

Michigan—but performance fell for the Jump group on questions involving the second and third games, especially on backward questions; in addition, the Short-Jump group performed particularly poorly on the third game, especially on backward questions concerning the third game. Since the first game (Michigan vs. Indiana) occurs in only one place in the problem tree, the second game (Ohio vs. Michigan) occurs on two branches, and the third game (Ohio vs. Indiana) occurs on four branches, it is reasonable to assume that questions involving the third game may require more processing than questions involving the second, and the first game should be easiest. The fact that the Example group performed at similarly high rates for all questions suggests that there are ways of representing the decision tree which allow simultaneous processing of all three games. However, the comparison between Jump and Nest groups is consistent with the idea that the branches of the tree are processed serially for these groups, and that the broken paths of the Jump organization are particularly detrimental to answering complex problems that require integrating of many branches. The Short-Jump organization may provide a somewhat more well-structured representation, but on very complex problems involving the third game performance is considerably poorer than Nest.

GENERAL CONCLUSION

Diagrams

There has been much discussion concerning the supposed benefits of "flow charts" in programming, and of diagrammatic representation as an aid in general problem solving (Paige & Simon, 1966). The present results indicate that (1) diagrams produce better performance if they replace particularly complicated verbal representation presumably due to faster accessing of information, and (2) diagrams result in poor performance if they are substituted for well-structured verbal representation, presumably due to distraction. There was, however, no consistent evidence that diagrams resulted in a different *kind* of internal processing style in working memory. This failure to find differences in reasoning style due to diagrams may be due to the particular kind of diagrams used—ones closely shadowing the verbal representation—and further research is needed to confirm the generality of our conclusion. In addition, there could be individual differences in the effects of diagram vs. verbal representation, and more research is needed here.

Organization

The present results extend the findings of Sime, Green, and Guest (1973) that a well-integrated "Nest" organization results in better performance overall than a list with "go to"-s such as "Jump," and

that even more compact organization such as "Example" is best overall. The main new finding is that different ways of organizing the problem representation influence the amount of information that can be processed—or the quality of integration in working memory—during problem solving. It is tempting to claim that the four organizations—Jump, Short-Jump, Nest, and Example—are points on a continuum of structural integration; our results are consistent with the idea that the four, respectively, allow subjects to process increasing amounts of complexity. In preliminary work on how novices learn to generate programs, Norman, Gentner, and Stevens (1975) suggest that subjects develop "schema" for various control statements (such as conditional "jumps") which may not be sufficiently broad. An important question is how subjects learn to modify their representations, and especially whether subjects can learn to "translate" the apparently complex "Jump" structure to a more understandable form.

In our experiments, the effects of presentation organization were still strong even on the second block of 72 questions (Experiment II). There is some evidence, however, that while the type of conditional structure influences novices, it has no effect on professional programmers, presumably due to their ability to "translate" to a deeper internal code (Schneiderman & Ho, Note 5). Apparently, the type of problem representation, especially the structures used to represent conditionals, have their strongest effect on unpracticed novices; hence, a careful consideration of such structures is particularly important for the development of new microlanguages to be used by the general public.

APPENDIX

Eight Problem Representations

Jump-Verbal

(1) If Indiana defeats Michigan go on to next step, otherwise go to step 7.

(2) If Ohio defeats Michigan go on to next step, otherwise go to step 6.

(3) If Indiana defeats Ohio go on to next step, otherwise go to step 5.

(4) You win prize F.

(5) You win prize E.

(6) You win prize D.

(7) If Ohio defeats Michigan go on to next step, otherwise go to step 9.

(8) You win prize C.

(9) If Indiana defeats Ohio go on to next step, otherwise go to step 11.

(10) You win prize B.

(11) You win prize A.

Nest-Verbal

  If Indiana defeats Michigan then

    If ohio defeats Michigan then

      If Indiana defeats Ohio then you win prize F.

      Otherwise you win prize E.

    Otherwise you win prize D.

Otherwise
    If Ohio defeats Michigan then you win prize C.
    Otherwise
        If Indiana defeats Ohio then you win prize B.
        Otherwise you win prize A.

Short-Jump Verbal

(1) If Indiana defeats Michigan go to step 2, otherwise go to step 4.

(2) If Ohio defeats Michigan go to step 3, otherwise you win prize D.

(3) If Indiana defeats Ohio you win prize F, otherwise you win prize E.

(4) If Ohio defeats Michigan you win prize D, otherwise go to step 5.

(5) If Indiana defeats Ohio you win prize B, otherwise you win prize A.

Example-Verbal

There are three games: (1) Indiana vs. Michigan, (2) Ohio vs. Michigan, (3) Indiana vs. Ohio.

If the winners are (1) Indiana, (2) Ohio, (3) Indiana, then you win prize F.

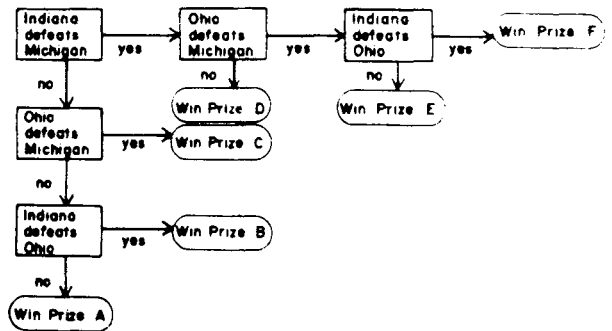If the winners are (1) Indiana, (2) Ohio, (3) Ohio, then you win prize E.

If the winners are (1) Indiana, (2) Michigan, (3) either team, then you win prize D.

If the winners are (1) Michigan, (2) Ohio, (3) either team, then you win prize C.
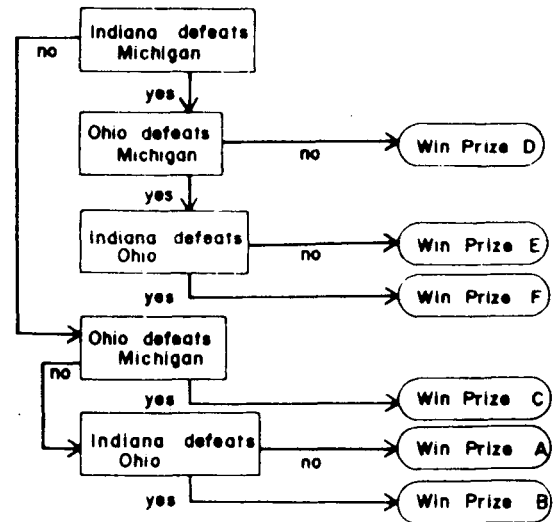
If the winners are (1) Michigan, (2) Michigan, (3) Indiana, then you win prize B.

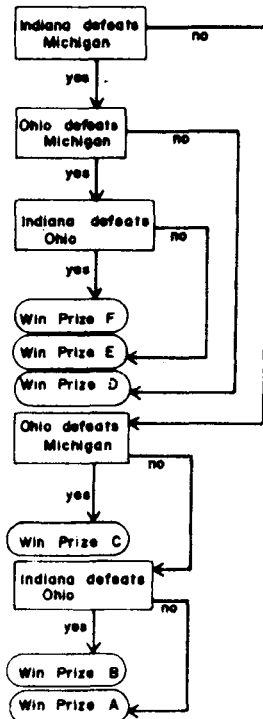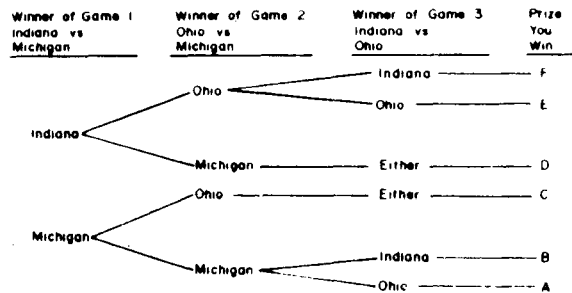If the winners are (1) Michigan, (2) Michigan, (3) Ohio, then you win prize A.

NEST—FLOW

SHORT—JUMP—FLOW

JUMP—FLOW

EXAMPLE—FLOW

REFERENCE NOTES

1. Mayer, R. E. *Instructional variables in meaningful learning of computer programming.* Indiana University: Mathematical Psychology Technical Report No. 75-1, 1975.

2. Miller, L. *Programming by nonprogrammers.* Yorktown Heights, N.Y: IBM Technical Report No. RC-4280, 1973.

3. Reisner, P., Boyce, R., & Chamberlin, D. D. *Human factors evaluation of two data base query languages.* San Jose, Calif: IBM Research Report, 1974.

4. Revlis, R. *Conversion process in syllogistic reasoning*. Paper read at meeting of the Psychonomic Society, November 1972.

5. Schneiderman, B., & Ho, M. *Two experiments in programming behavior*. Indiana University: Computer Science Department Technical Report No. 17, 1974.

6. Steffe, L. P. *The effects of two variables on the problem-solving abilities of first-grade children*. University of Wisconsin: Center for Cognitive Learning Technical Report No. 21, 1967.

REFERENCES

GREENO, J. G. The structure of memory and the process of solving problems. In R. Solso (Ed.), *Contemporary issues in cognitive psychology: The Loyola Symposium*. Washington: Winston, 1973.

HAYES, J. R. Psychological differences among problem isomorphs. In N. J. Castellan, Jr., & D. B. Pisoni (Eds.), *Cognitive theory* (Vol. 2). Hillsdale, N.J: Lawrence Erlbaum Associates, 1975.

LOFTUS, E. F., & SUPPES, P. Structural variables that determine problem-solving difficulty in computer-assisted instruction. *Journal of Educational Psychology*, 1972, **63**, 531-542.

NORMAN, D. A., GENTNER, D. R., & STEVENS, A. L. Comments on learning: Schemata and memory representation. In D. Klahr (Ed.), *Cognitive approaches to education: Proceedings of the 10th Carnegie-Mellon University Symposium on Cognition*. Hillsdale, N.J: Lawrence Erlbaum Associates, 1975.

PAIGE, J. M., & SIMON, H. A. Cognitive processes in algebra word problems. In B. Kleinmuntz (Ed.), *Problem solving*. New York: Wiley, 1966.

POLICH, J. M., & SCHWARTZ, S. H. The effect of problem size on representation in deductive reasoning. *Memory & Cognition*, 1974, **2**, 683-686.

REED, S. K., ERNST, G. W., & BANERJI, R. The role of analogy in transfer between similar problem states. *Cognitive Psychology*, 1974, **6**, 436-450.

SCHWARTZ, S. H. Modes of representation and problem solving: Well evolved is half solved. *Journal of Experimental Psychology*. 1971, **91**, 347-350.

SIME, M. E., GREEN, T. R. G., & GUEST, D. J. Psychological evaluation of two conditional constructions used in computer languages. *International Journal of Man-Machine Studies*, 1973, **5**, 105-113.