Title
Cognitive Style and The Learning of a First Computer Language

Journal
Proceedings of the Annual Meeting of the Cognitive Science Society, 3(0)

Author
Coombs, M. J.

Publication Date
1981

Peer reviewed

Cognitive Style and the Learning of a First Computer Language

M.J. Coombs
Man-Computer Communication Research Group
Computer Laboratory
University of Liverpool
Liverpool, L69 3BX
U.K.

1. Individual Differences and Computing Skills

It is commonly asserted that the learning of computing skills is subject to marked individual differences (e.g. Weinberg, 1971; Gould, 1975; du Boulay and O'Shea, 1981). Such differences are of considerable importance for both practical and theoretical reasons. First, an analysis of successful and unsuccessful learning strategies should provide data on the nature of computing information and the skills required for its acquisition. This information would aid in the design of systems for guiding the rapidly increasing population of computer users who are not computer professionals. Secondly, the existence of clear individual differences should provide an ideal environment for exploring the relationships between learning strategy and cognitive performance.

Research was therefore undertaken with these two objectives in mind, but emphasising the first. Work was conducted in two stages:

> the identification of the main parameters of learning style related to the learning of a first computer language;
>
> the definition of interactions between style, strategy and knowledge representation.

2. Stage 1: Four Programming Studies

In order to characterize the relationship between learning style and the acquisition of programming skills and to provide early applicable results, research was conducted on science and engineering postgraduate students taking a standard introductory FORTRAN course (Coombs et al., 1981).

A research methodology was used in which subjects were presented with two tasks: a "target task" for the assessment of programming skill, and an "indicator task" for the assessment of learning style. The objective was to characterize performance on the target task, about which little was known, with reference to performance on the indicator task. It was thus important that the indicator task was well-founded and had some clearly defined relationship with the target task.

Two programming tests were devised for the target task, following the finding (Coombs, 1977) that recall of the units of procedural information may be independent of ability to recall the units in their correct order. The first test the Statement Test was devised to assess learning of the format and operation of individual FORTRAN structures and required the identification of appropriate errors in 3 short programs (21 in 56 lines of program). The second test the Logic Test was devised to assess ability to assemble individual structures into successful programs. Subjects were presented with 3 decks of cards, each comprising a complete program. The cards were given in random order and subjects were required to reconstruct the original program.

Selection of an indicator task proved difficult because of the poorly understood and unstable relationships between tests of cognitive style. A test was therefore used which had significant features in common with the skills assessed by the target tasks. A suitable classification of individual differences was found in the Operation Learning/Comprehension Learning distinction proposed by Pask (1975). The distinction was founded in a well-elaborated theory Conversation Theory and a test was available the Spy-Ring History Test.

Pask's categories are defined within two basic dimensions of human information processing: a) the management of data selected from the world (attention); b) the mental representation of that data (mental model building). The attention dimension draws the binary distinction between local and global features of a subject material. The representation dimension draws the distinction between representation of new information in terms of descriptions and in terms of procedures to be used for its generation from previously acquired information. Given interactions between the dimensions, Pask proposes two basic cognitive styles: operation learning, (procedure building using low-level, local information); comprehension learning (description building with attention to global features). Students biased towards operation learning are thus able to learn rules, methods and details but are unaware of how and why they fit together. Comprehension learners learn an overall picture of the subject matter but may not be able to perform the operations to use it. Comprehension of the information exists in the absence of rules or operational meaning and in ignorance of details required for the information to be used.

The procedural aspect of computing provides a conceptual link between cognitive style and the programming tests, it being our contention that learners with an operation bias would have a dual advantage of the Logic Test but no advantage on the Statement Test. With the Logic Test, operation learners would be expected to attend during the learning of a language structure to its internal logic. However, comprehension learners would only attempt to remember the global features of the structure, making no close distinction between the essential features and those arising from its illustrative context. A comprehension learner would be expected to accept the example as given and always work within the framework until he was presented with a different example. The operation learner would thus have an advantage when faced with the Logic Test, because he would have a clear idea of the essential features of the structure independent of context. Structures could therefore be readily assembled, without interference, to complete the problem program. The operation learner would also be expected to gain a second advantage by applying an operation strategy to the problem itself.

The Statement Test would not be expected to be so sensitive to learning style, all errors being contained within individual statements and not being related to the logic of the programs. They should therefore be identifiable from incidental learning of the surface features of the language. They would not 'look right' to anyone who had been exposed to FORTRAN, irrespective of his understanding of the workings of the language.

Four programming studies were conducted on different groups of students attending a standard FORTRAN course, the only addition being a request

3. Stage 2: Cognitive Style to Learner Strategy

The research reported above focussed on the role of cognitive style in learning. However, in order to apply the findings to designing educational and guidance systems for unsuccessful students, one must translate style into strategy. Protocol data therefore was collected during the first, third and fourth programming study on subjects' learning activities during the lectures and practicals. In the fourth study a computer based concept elicitation technique was used to record the changing understanding of concepts during the course. The protocol studies concentrated on subjects with an extreme operation learning (10) and extreme comprehension learning (8) bias.

The research has provided a description of two classes of strategic element: those concerned with mental processing and those concerned with learning activities. From these two classes it is possible to identify model strategies which are similar to competence models in linguistics. In the presence instance our model strategies also have sufficient empirical foundation to provide a basis both for planning further research and guiding instruction.

Learning to use any symbol system requires the student both to progressively identify acceptable structures and to apply them to some processing objective. Moreover, the competent use of a symbol system implies the ability to perceive structures at many different levels (Van Dijk, 1977). A symbol system, therefore, has both microstructure and macrostructure. Both levels should be born in mind when interpreting our findings. Green et al (1981) emphasises the importance of macrostructure for the writing and debugging of programs.

Operation learners appear to actively use both lecture and practical sessions. The former are used for recording facts, the learner ensuring that the lecturer's statements and examples are recorded accurately in full. Examples are also recorded carefully, the operation learner's priorities being a) to copy the example, b) to follow the lecturer's analysis of the logical relations between statements, c) to consider alternative methods of achieving the same programming objective. In b) and c) the operation learner works from "inside" the code, paying close attention to the logical relations between statements. By following these activities, the student leaves the lecture with an accurate and relatively uninterpreted record of the structures discussed and the examples given, and understanding the internal logic of most of the examples.

The major learning activity of operation learners takes place during practicals. Here they continue to build their knowledge of the functional relations between low-level language structures so leading to a knowledge of useful macrostructures. This again appears to be achieved from the "inside" by, for example, seeking various combinations of statements which will achieve the same processing objective. Operation learners thus build considerable knowledge of the computational possibilities of combinations of statements and rapidly develop a rich sense of macrostructure. The computer is itself the instructor, reference only being made to external information such as the principles behind the local implementation of FORTRAN or the design of the local operation system when prompted by a program failure which cannot otherwise be corrected. Through the practicals, the operation learner develops an understanding of the logic of program design and skills related to debugging and avoidance of errors.

Comprehension learners, however, adopt a very different distribution of activity. During lectures they attempt to achieve what the operation learner reserves for practical sessions. The comprehension learner performs significant "on line" processing and is rarely fully engaged in simply reporting facts and examples. He rarely takes verbatim notes, rather recording the abstracted or generalized results of his processing of the lecture material. Such processing is, of course, carried out without the aid of the computer. Significant structure is therefore determined by reference to external sources of information from related domains such as FORTRAN design principles or details concerning the local machine. Where relevant information is not available, the comprehension learner tends to create determining principles for himself rather than store the low-level information in unintegrated form. The comprehension learner thus leaves the lecture with represntations of language structures which have been determined to some extent by factors external to FORTRAN and which have never been validated by an actual computer run. Moreover, his knowledge of structure does not that students complete the Spy-Ring History Test early in the course and the programming tests at the end. A total sample of 42 subjects completed all tasks over a main study and four replications. The Spy-Ring Test yielded for each subject a score of operation, comprehension and incidental learning. The Logic Test was scored using a count of simple first-order transitional errors and the Statement Test was scored using the signal-detection theory measure of $P(\bar{A})$. Relationships were assessed using Pearson Correlation Coefficients.

The results for all these studies revealed one stable significant relationship (.05 level) between the exercise of operation learning and the Logic Test. Interpreting this result using Pask's (1975) theory and our own confirmatory factor-analysis of the Spy-Ring Test, it was concluded that the skill of assembling statements into programs was best acquired by subjects who:

- paid close attention to details;
- systematically abstracted the critical features of programming structures;
- represented structural relations in rule form.

Evidence of the relationship between cognitive style and the Statement Test was inconclusive, although contrary to our hypothesis there appeared to be a weak relationship with operation learning.

From results at this stage it was possible to make 4 assertions concerning the learning of FORTRAN:

1) One can define at least 2 different learning styles in a population of novice computer users.
2) Those exhibiting significant operation learning are more successful at assembling language structures into an effective algorithm.
3) The successful learning style is defined by close attention to detail and a preference for procedural representation.
4) Successful identification of individual language structures is independent of learning style.

develop beyond this point because he also fails to take advantage of practicals to conduct validation. The comprehension learner thus leaves a course with a descriptive knowledge of individual, low-level structures but no reliable, tested knowledge of macrostructure.

5. Some Final Remarks

The above accounts suggest that all students should be encouraged to use an operational strategy. However, Pask (1976) proposes that comprehension learners should be left with their natural style but with:

accurate conceptual and supporting information;
encouragement to solve problems in a structured environment.

It may be argued that the unique characteristics of FORTRAN and the specific nature of the Liverpool course make general proposals premature. Whilst accepting this possibility, we believe that the effects of operation and comprehension styles are sufficiently universal that similar differences will be found with other languages and computing situations. Although differences in detail may exist, we propose that whilst programming requires precise specification of code, and whilst languages fail to make macrostructure perceptually evident and are implemented on machines not easily modelled by the novice, similar effects will be found.

6. References

COOMBS, M.J. (1977). Modality and order recall in learning from television, with reference to teaching medical procedures. Ph.D. Thesis, University of Liverpool.

COOMBS, M.J., GIBSON, R. and ALTY, J.L. (1981). Acquiring a first computer language: a study of individual differences. In M.J. Coombs & J.L. Alty (eds), Computing Skills and the User Interface. London: Academic Press.

DU BOULAY, B. & O'SHEA, T. (1981). Teaching novices programming. In M.J. COOMBS & J.L. ALTY (eds), Computing Skills and the User Interface. London: Academic Press.

GOULD, J.D. (1975). Some psychological evidence on how people debug computer programs. International Journal of Man-Machine Studies, 7, 151-182.

GREEN, T.R.G., SIME, M.E. & FITTER, M.J. (1981). The art of notation. In M.J. COOMBS & J.L. ALTY (eds), Computing Skills and the User Interface. London: Academic Press.

PASK, G. (1975). Conversation, Cognition and Learning. Amsterdam: Elsevier.

PASK, G. (1976). Conversation Theory : Applications in Education and Epistemology. Amsterdam: Elsevier.

VAN DIJK, T.A. (1977). Semantic macrostructures and knowledge frames in discourse comprehension. In M.A. JUST & P.A. CARPENTER (eds), Cognitive Processes in Comprehension. New York: Wiley.

WEINBERG, G.M. (1971). The Psychology of Computer Programming. New York: Van Nostrand Reinhold.