ISSN 0254-2757

QI QUÆSTIONES INFORMATICÆ

Volume 5 • Number 1 April 1987

An official publication of the Computer Society of South Africa and of the South African Institute of Computer Scientists

'n Amptelike tydskrif van die Rekenaarvereeneging van Suid-Afrika en van die Suid-Afrikaanse Instituut van Rekenaarwetenskaplikes

QUÆSTIONES INFORMATICÆ

An official publication of the Computer Society of South Africa and of the South African Institute of Computer Scientists

'n Amptelike tydskrif van die Rekenaarvereniging van Suid-Africa en van die Suid-Afrikaanse Instituut van Rekenaarwetenskaplikes

Subscriptions

Annual subscription are as follows:

	SA	US	UK
Individuals	R10	$ 7	£ 5
Institutions	R15	$14	£10

Quæstiones Informaticæ is prepared by the Computer Science Department of the University of the Witwatersrand and printed by Printed Matter, for the Computer Society of South Africa and the South African Institute of Computer Scientists.

ON LEARNING STYLES AND NOVICE COMPUTER USE

G. R. Finnie
Department of Computer Science,
University of Natal,
Pietermaritzburg, 3200

ABSTRACT

Learning to use a decision support system involves both formal and informal teaching as well as self-teaching. Many cases have been documented of unsuccesful implementation of DSS or MIS in organisations and identifying successful learning styles for novice DSS users would be of value for systems design and the development of training programs, training aids, documentation, and computer-based "help" systems.

This paper describes an experiment performed to relate specific learning style profiles (as assessed by Kolb's Learning Styles Inventory) to aspects of DSS use by novices. Analysis indicated a possible relationship between successful use and one learning style dimension while stronger relationships were found between learning styles and both use of the "help" facility and the frequency of errors.

ACM Categories: H.1.2, H.4.2, K.3.0.

1. INTRODUCTION

Computer based decision support systems (DSS) and management information systems (MIS) can play an important role in the management of business operations. Incorporating (supposedly) "user-friendly" features such as nonprocedural structure, menu-driven interfaces and extensive "help" subsystems, these packages bring in a new class of novice or occasional computer user who requires the computer as a tool to support their expertise in other areas and who are not intent on, or interested in, becoming professional programmers. Aquiring the ability to use the more sophisticated packages is a non-trivial exercise involving both formal and informal teaching as well as self-teaching. Many potential users of DSS or MIS do not utilise these systems, even if training has been provided, and numerous cases of failure to successfully implement MIS/DSS have been documented (e.g. [16],[18].)

Individuals show considerable differences in their ability to make use of "conventional" computing facilities and programming languages [12] and it is likely that similar differences exist in the MIS/DSS area. Identifying successful learning styles for novice DSS users would be of value in such areas as systems design and the development of training programs and training aids. Coombs, Gibson and Alty [5:292], in their study of successful learning styles for FORTRAN, observe that

> *"... the study of such individual differences in the learning of computer skills would provide valuable insights into ways of effectively supporting computer users. First, an analysis of the contrasting learning strategies used by successful and unsuccessful students should provide data on the nature of computing information itself and the cognitive skills required for its aquisition. This would aid in the design of documentation, machine-based "help" facilities and training courses. Secondly it would be helpful to be able to identify in advance those individuals needing special attention so that they can either be given an individual training program or be instructed at an early stage in the relevant basic learning skills."*

These factors are as valid in the area of decision support systems as in that of "conventional" programming.

This study deals specifically with the effect of learning style on the ability of novices to successfully use a DSS package (a simple nonprocedural financial modelling system) as well as investigating the role of learning style in the use of "help" facilities and on error generation during model development. The learning styles considered are those proposed by Kolb [14].

2. STUDYING NOVICE COMPUTER USERS

A considerable amount of research effort has been applied to the study of programmer behaviour and other aspects of human-computer interaction. An intensive review of the psychological study of novices learning programming in "conventional" languages such as BASIC or PASCAL has been provided by Mayer [17]. Du Boulay and O'Shea [7] provide another review of the field, covering various areas of difficulty in teaching, language design and the unanswered (and probably unanswerable) question of "which first language" to teach. A number of authors have discussed the problems of experimental reasearch in the programming area (e.g. [5], [13], [21]) with a review being provided by Brooks [4]. Little appears to have as yet been done directly in the DSS area, although much of the work in fields such as database query languages (e.g. [23]) and the design of "menus" and "help" subsystems (e.g. [22]) is of course applicable.

The study of learning styles of novice computer users or novice programmers has not received extensive coverage, an exception being the work of Coombs et al [5], [6] on the learning of FORTRAN. These authors selected a learning style classification based on the model by Pask [19] which identifies two major learning modes, i.e. operation learning and comprehension learning. Initial results in this research tended to suggest that operation learning was a more successful style in procedural languages.

3. THE KOLB LEARNING STYLES INVENTORY

Kolb's Learning Styles Inventory (LSI) is based on a model which suggests that learning style is a result of heredity, experience and present environment and is intended to assess aspects of both learning and problem solving techniques. To quote Kolb [15:37]

> *"By combining these characteristics of learning and problem solving and conceiving of them as a single process, we can come closer to understanding how it is that people generate from their experience concepts, rules, and principles to guide their behaviour in new situations, and how they modify these concepts in order to improve their effectiveness. This process is both active and passive, concrete and abstract. It can be conceived of as a four-stage cycle:*
>
> *(1) concrete experience is followed by*
> *(2) observation and reflection, which leads to*
> *(3) the formation of abstract concepts and generalizations, which lead to*
> *(4) hypotheses to be tested in future action, which in turn leads to new experiences."*

Each phase in the learning cycle defines a specific learning mode and to measure the relative emphasis placed by an individual on a learning style, the Kolb LSI requires each subject to rank order nine sets of four words, each word within a set being supposedly indicative of a specific learning mode.

The learning modes may be briefly characterised as follows [15:39-40]

(a) Concrete experience (CE): receptive experience-based approach to learning which relies heavily on feeling-based judgements, empathetic and people-oriented; find theoretical approaches unhelpful; learn best from specific examples; tend to be oriented more toward peers and less toward authority.

(b) Abstract conceptualisation (AC): Analytical, conceptual approach to learning that relies heavily on logical thinking and rational evaluation; oriented more towards things and symbols than people; learn best in authority-directed, impersonal learning situations that emphasize theory and systematic analysis; benefit little from unstructured "discovery" learning approaches.

(c) Active experimentation (AE): active "doing" orientation to learning; learn best in projects, homework or small group discussion; dislike passive learning; tend to be extrovert.

(d) Reflective observation (RO): tentative, impartial and reflective approach to learning; rely heavily on careful observation in making judgements; prefer learning situations such as lectures; tend to be introvert.

Factor analysis of learning modes indicates that these can be combined into two biploar dimensions, i.e. CE vs AC and AE vs RO. To establish an individual's learning style, these dimensions can be laid out in grid form (Figure 1), each quadrant of the grid being indicative of a specific learning style profile in which two of the learning modes dominate their counterparts. Kolb characterised these types as accommodators, divergers, assimilators and convergers. Some relevant characteristics of each type are as follows [15:40-42]:

(a) Convergers: do best where there is a single correct answer; focus on specific problems; relatively unemotional; prefer to deal with things rather than people; narrow technical interests; characteristic of engineers.
(b) Divergers: greatest strength in imaginative ability; ability to view concrete situations from many perspectives; interested in people; emotional; tend to specialize in the arts; characteristic of counsellors, organisation development and personnel managers.
(c) Assimilators: ability to create theoretical models; excel in inductive reasoning and assimilating disparate observations into integrated explanations; more concerned with abstract concepts than people; characteristic of research and development departments.
(d) Accomodators: greatest strength in carrying out plans and experiments and involving oneself in new experiences; more of a risk-taker; adapt to circumstances; solve problems in an intuitive trial-and-error manner, relying on other people for information; characteristic of marketing or sales.

The measurement of learning styles using this instrument has been challenged on psychometric grounds by Freedman and Stumpf [11]. Although a two factor solution of their sample supported Kolb's theory of two bipolar dimensions, the total variance accounted for was only 20.6% while a four factor solution accounted for 30%. A four factor analysis by Ferrell [8] similarly accounted for only 31.6% of total variance, although again providing some support for construct validity. The large amount of unexplained variance raises questions about the reliability of the instrument, with Freedman and Stimpf stating that

> *"The LSI appears to be an example of a worthwhile idea which has some theoretic value but has been operationalised too soon."*

Beutell and Kressel [3] have questioned the use of a forced-choice format to control for social desirability and suggest that normative formats should be considered. West [24], using standard personality profile tests, was unable to relate learning styles to the personality types proposed by Kolb.

Despite the weaknesses suggested by the above research, the LSI has been fairly widely used, particularly in medical education. The work by Ferrell on four learning-style instruments provided some validation only for the Kolb instrument, suggesting that the others are even more suspect. Although the robustness of the Kolb metric can be attacked, it was selected for this study because the learning-modes have certain characteristics which might be expected to influence the way in which novices acquire knowledge of computer use, e.g. one might expect subjects emphasizing abstract conceptualisation over concrete experience to be more successful at acquiring computer skills given that this mode relies more on logical thinking and rational evaluation than feeling-based judgements. Similarly, the problem solving approach adopted for those emphasizing active experimentation might be expected to differ from those who stressed reflective observation as a major learning mode.

4. RESEARCH AIMS

This study was designed to investigate the following questions concerning the interaction of Kolb's learning styles and computer use:

(a) Does a particular learning style influence the successful use of DSS by novices?
(b) Do individuals adopting specific learning styles experience more difficulty (in terms of error rates) than those utilising other styles?
(c) Do particular learning styles influence the use of computer systems as a learning aid?

These questions are of some commercial significance as regards the introduction and use of DSS in organisations. Unsuccessful initial use of a DSS by novices has been shown to significantly create or enhance negative attitudes towards computers [9] and attitude plays an important role in the use of MIS or DSS in organisations [9], [16], [20], [25]. It is probable, although not proven, that difficulties (e.g. high error rates) could similarly influence attitudes. A high error rate might also be indicative of a particular debugging or problem solving style. As noted by Coombs et al [5], [6], knowledge of learning style differences in these areas could aid in identifying individuals who might need special attention as well assisting in the design of training courses and methods.

Most decision support systems provide fairly extensive "help" facilities and several authors have stressed the value of machine-based interactive teaching and assistance for novices (e.g. [22]). If, however, certain learning styles inhibit the use of these facilities , alternative training aids should be considered. Although consideration could be given to instruction in the relevant learning skills, Pask [19] found that, within his framework, it was very difficult for students to change their learning style and that they functioned considerably less effectively under different styles.

As the research was essentially investigative, no specific hypotheses were formulated regarding these questions. Tests were applied to determine whether significant differences existed for the Kolb LSI variables and bipolar dimensions between successful and unsuccessful users, between those with high and low error rates and between those who utilised the "help" facility as opposed to those who did not.

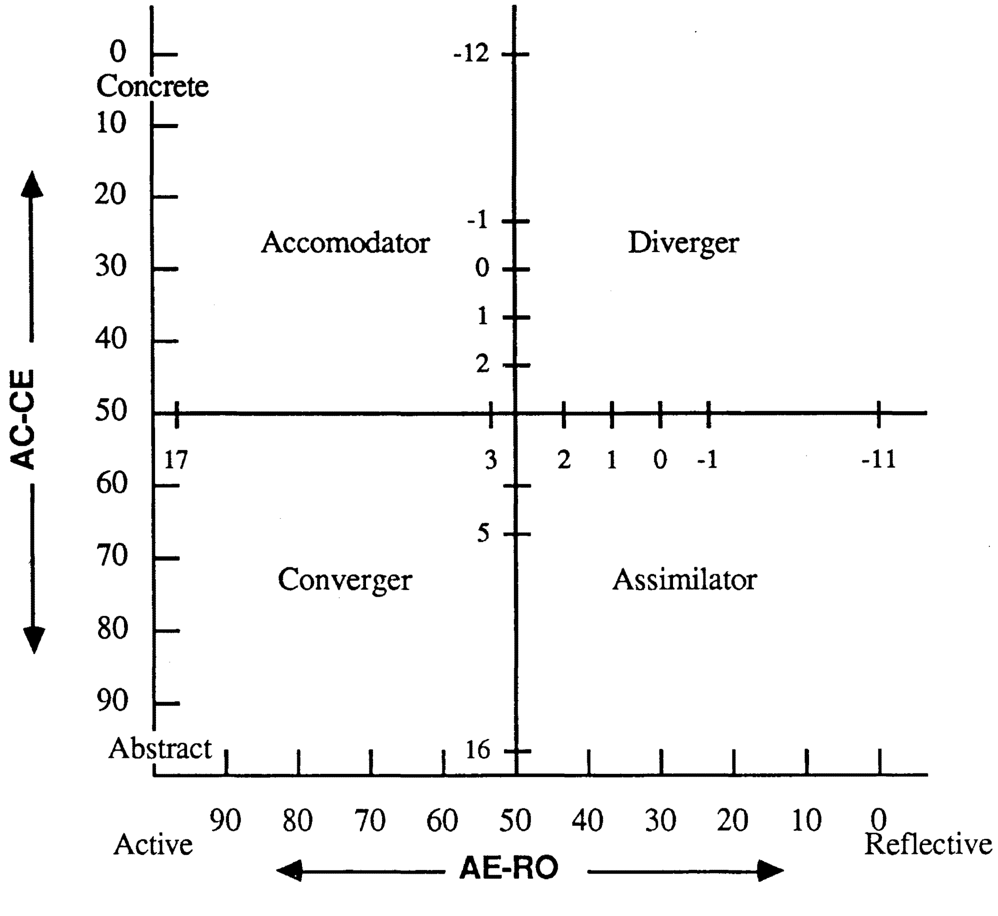

figure 1

Learning Style Type Grid (Copyright 1976 by David A. Kolb)

5. EXPERIMENTAL METHOD

The subjects in the experiment were 142 undergraduate business administration students beginning their first course dealing with commercial aspects of computing. 17 of these had some prior experience of computing and were removed from the analysis and a further 19 were removed for noncompletion of the project or failure to complete the Kolb LSI satisfactorily. The remaining 106 subjects were true novices in the use of DSS and computers in general. The subjects were required to complete a case study which required the development of a simple financial model to answer a number of questions concerning the case. Participation was considered part of the course curriculum.

The financial models were developed using a package called MODELLER. This is a simple nonprocedural financial modelling system designed to collect a variety of statistics on novice interaction with decision support systems as well as acting as a control mechanism for student experiments [10]. The system is self-contained, consisting of a compiler, interpreter, editor and simple filing system. The language for model definition is similar to a subset of IFPS (Execucom, 1984), although it obviously lacks much of the sophistication of this package. The restriction of the MODELLER system to a relatively small command and model set has resulted in a system which can be used to teach the principles of financial model design rapidly without subjects being distracted by the complexity of a complete commercial system. Although limited for general application, the system is well suited to an undergraduate teaching environment where case studies can be defined for the capabilities available and where the more sophisticated financial modelling facilities are not required. An additional teaching advantage over commercial packages has been that it does not consume machine resources to the same extent. At its present installation, all student terminals may be in action simultaneously without severe response degradation.

MODELLER traces model development and collects statistics by creating a separate trace file for each user. The file is locked to prevent illegal access and is invaluable as a control mechanism in a student user environment. A variety of data is collected for each user, including date and time stamps for each session, copies of models at various stages of development, tracing of editing operations performed, compilation error distributions and a record of model manipulation. MODELLER also collects data on the way in which the nonprocedural features of the language are used in practice.

The definition of successful use of DSS or MIS has received considerable research attention, with measures based on actual usage, user satisfaction and user performance being applied (see e.g. [1], [20]). The use of error rates to assess successful "conventional" programming has also been considered [2]. Given the conditions of forced participaton in this study, subjects were classified as successful or unsuccessful by a subjective evaluation of the quality of the result and an assessment of the extent to which the model was the student's own work. Careful analysis of the trace files generated by MODELLER, particularly noting the date and time at which the project was started as well as the amount of correction required to a model, enabled most cases of copying to be detected. 23 subjects were removed from the classification as being doubtful in terms of the amount of work actually done by themselves while a further 9 had midrange results and were not considered for the successful/unsuccessful categories. 31 subjects were classified as unsuccessful, mostly in terms of totally inadequate or incorrect results although a number of subjects who had copied partially or totally incorrect results were included here. 43 subjects were classified as successful for having developed adequate solutions and having done the bulk of the work themselves.

6. ANALYSIS AND RESULTS

6.1 Successful DSS Use

A t-test was applied to the four learning mode variables and the two difference dimensions (AC-CE and AE-RO) to determine whether differences were present in learning styles between successful and unsuccessful novice DSS users (Table 1). Although not particularly strong, significant differences were present on the reflective observation variable (P<0,05) and on the dimension AE-RO (P<0,025). Successful users rated higher on reflective observation (X= 14,26) than did unsuccessful (X = 12,93). Successful users generally lie within the diverger/assimilator plane (X = 0,51) while unsuccessful users lie more towards the midpoint of this scale (X = 2,94). To assess whether the significant differences extend to the interaction of the learning styles (i.e. the quadrants), a 4 x 2 contingency table was used. These differences are not significant ($\chi^2 = 2,83$).

Given the characteristics of the learning modes outlined in section 3, the differences are possibly surprising as it could have been hypothesised that differences would be more likely to exist in the AC-CE dimension or, alternatively, that active experimentation would be a more successful learning mode for computer package use than reflective observation. However, given the number of variables involved and the fairly weak significance levels, the validity of these findings should be treated with caution.

Variable	U/S	X	t
CE	U	14,48	0,60
	S	14,91	
RO	U	12,93	2,18 *
	S	14,26	
AC	U	18,64	0,91
	S	17,89	
AE	U	15,87	1,80
	S	14,77	
AC-CE	U	4,16	0,89
	S	2,98	
AE-RO	U	2,94	2,37**
	S	0,51	

*	P < 0,05
**	P < 0,025

U = Unsuccessful	(n = 31)
S = Successful	(n = 43)

table 1

Learning Styles and Successful DSS Use

6.2 Use of "HELP" Facilities

In assessing the relationship between learning styles and the use of computer-based aids, users were partitioned into those who made no requests to the "help" facilities (61 subjects) and those who made at least one request (22 subjects). (The nine midrange subjects excluded from the successful/unsuccessful analysis were added here). A t-test for group differences in the learning style variables and dimensions (Table 2) indicated significant differences in the active experimentation and reflective observation learning modes as well as in the AE-RO difference dimension. Interestingly, these differences are the reverse of those observed as possibly significant in the successful/unsuccessful analysis. Users who rate highly on the active experimentation mode tend to use the "help" facilities while the reverse holds for those rating higher on reflective observation. The finding would be in line with Kolb's theory with the AE group having an "active doing orientation to learning" and the RO group adopting a reflective approach. A 4 x 2 contingency table was used to assess whether the significant differences extended to the interaction of the learning styles but again the difference was not significant (χ^2 =

4,08) i.e. the differences lie in one plane and not within the quadrants.

Given that the variables significant in the use of "help" facilities appear the reverse of those possibly significant in successful use, it might be assumed that successful users felt no need for the "help" facility while unsuccessful users did. This does not, however, appear to be the case. Of the "help" facility users, 45% (10) were successful and the difference in a 2 x 2 contingency table relating success/failure with the use of the "help" facility is not significant ($\chi^2 = 1,88$).

Variable	N/H	X	t
CE	N	14,75	0,61
	H	14,32	
RO	N	14,21	2,33 *
	H	12,72	
AC	N	17,98	1,30
	H	19,09	
AE	N	14,72	2,01 *
	H	16,00	
AC-CE	N	3,23	1,14
	H	4,77	
AE-RO	N	0,51	2,37**
	H	3,27	

* P < 0,05
** P < 0,01

N = No use of HELP
H = Used HELP

table 2

Learning Styles and HELP Facility Use

6.3 Learning Styles and Error Rates

Both the absolute and relative compilation error rates were used to assess the difficulties experienced by novices in using the system. The relative error rate is simply the absolute error count divided by the total time taken to develop the model, both statistics being collected by MODELLER. Although the inclusion of logical errors would have given a more rounded picture of novice problems, the extraction of these from the trace files is a time-consuming and error-prone process. Analysis of variance was used to test whether any significant differences exist in error rates for either difference metric (AC-CE and AE-RO) or for their interaction. The subjects were classified into two groups for each difference dimension by assigning those with a score of less than three to one group and those with a score exceeding three to another. Subjects with a score of three on either dimension were removed. The results are given in table 3.

A significant difference is present for both the absolute and relative error rates on the AC-CE plane. The interaction effect is negligible indicating that the learning style profile (i.e. specific quadrant) is not significant in terms of error generation. Subjects emphasising the concrete experience (CE) learning mode over the abstract conceptualisation (AC) mode had a higher error rate (X= 143,4) than those placing more stress on abstract conceptualisation (X= 92,3). The absolute and relative compilation error rates were also correlated with the learning mode variables and the difference metrics (Table 4). Both the CE and AC variables correlate significantly with the error rate (as does AC-CE). The error rate is positively correlated with the CE mode and negatively correlated with AC, i.e. a higher error rate is associated with the concrete experience approach and the reverse for abstract conceptualisation. This finding appears to tie in with Kolb's classification as individuals applying a concrete experience learning mode (feeling-based judgements, people-oriented, dislike theoretical approaches) might be expected to experience more difficulty in using the computer systems than those emphasizing abstract conceptualisation (analytical, logical thinking, rational evaluation, impersonal learning situations).

The question of course arises as to whether a higher error rate is associated with the success or failure of development. A t-test on relative error rate between successful users (X = 0,55) and unsuccessful users (X = 0,85) was significant (t= 2,08; P<0,05), i.e.unsuccessful users make more errors per unit time.

7. DISCUSSION

Although Kolb's learning styles instrument can be attacked for its psychometric properties, it is apparent that it has some ability to discriminate between certain aspects of computer use by novices. Irrespective of the labels attached to each learning mode (i.e. what is in fact being measured), this study indicated that learning styles have some potential to distinguish successful users from unsuccessful and to identify those likely to experience difficulty with systems like DSS (in terms of a high error rate). Learning styles also appear to differ between those who do not use computer-based "help" facilities and those who do (and who might be amenable to computer-aided self-teaching).

As discussed earlier, successful initial use of a DSS has been shown to have significant effect on attitude towards computers, while attitude is a well known factor in the successful implementation of MIS or DSS in organisations. An instrument based on the constructs underlying Kolb's learning profiles of active experimentation and reflective observation has the potential to identify novice users who might be unsuccessful as well as those unlikely to react well to a computer-aided self-teaching approach. Similarly, the learning profiles of abstract conceptualisation and concrete experience could assist in identifying those likely to have some difficulty in using DSS. Learning mode differences could also provide some guidance for documentation and the design of training courses. However, Kolb's LSI is a general learning styles instrument and further research would be necessary to develop a specialised form for use in assessing the learning of computer skills.

	Absolute Error Rate		Relative Error Rate	
	F	Significance	F	Significance
Main Effect	4,11	0,021	4,27	0,018
AC-CE	8,23	0,006	8,36	0,005
AE-RO	0,07	0,790	0,59	0,444
Interaction	0,69	0,408	0,75	0,389

table 3

ANOVA of Learning Styles and Error Rates

	Absolute Errors	Relative Errors
Variable	r	r
CE	,26*	,29**
RO	,14	,03
AC	-,32**	-,26*
AE	,06	,03
AC-CE	-,34***	-,32**
AE-RO	-,05	,00

* : P < 0,01
** : P < 0,005
*** : P < 0,001

table 4

Regression of Learning Style Variables with Error Rates

REFERENCES

1. Bailey, J.E. and Pearson, S.W., [1983], Development of a Tool for Measuring and Analysing Computer User Satisfaction., *Management Science,* **29(5)**, pp 530-545.
2. Bell, D., [1976], Programmer Selection and Programming Errors., *Computer Journal,* **19**, pp 202-206.
3. Beutell, N.J. and Kressell, S.S., [1984], An Investigation of Kolb's Learning Styles Inventory : Social Desirabilty and Normative Scoring., *Psychological Reports,* **55**, pp 89-90.
4. Brooks, R.E., [1980], Studying Programmer Behaviour Experimentally : The Problems of Proper Methodology., *Comm ACM,* **23**, pp 207-213.
5. Coombs, M.J., Gibson, R. and Alty, J.L., [1981], Acquiring a First Computer Language : A Study of Individual Differences., in M.J. Coombs and J.L. Alty (Eds.), *Computing Skills and the User Interface.,* Academic Press, London, pp 289-314.
6. Coombs, M.J., Gibson, R. and Alty, J.L., [1982], Learning a First Computer Language : Strategies for Making Sense., *Int. Jnl. Man-Machine Studies,* **16**, pp 449-486.
7. Du Boulay, J.B. and O'Shea, T., [1981], Teaching Novices Programming., in M.J. Coombs and J.L. Alty (Eds.), *Computing Skills and the User Interface.,* Academic Press, London.
8. Ferrell , B.G., [1983], A Factor Analytic Comparison of Four Learning-styles Instruments., *Jnl. Educ. Psycho.,* **75**, pp 33-39.
9. Finnie, G.R., [1985], *Using Management Decision Support Systems : an Experimental Investigation of the Role of Attitude, Locus of Control and Nonprocedural Design.,* Unpublished DBL thesis, University of South Africa.
10. Finnie, G.R., [1986], MODELLER : A DSS for Experimental Study of Human-computer Interaction., *Quaestiones Informaticae* , **4**(2), pp 1-6.
11. Freedman, R.D. and Stumpf, S.A., [1978], What can one Learn from the Learning Style Inventory?, *Academy of Management Journal,* **21**, pp 275-282.
12. Gould, J.D., [1975], Some Psychological Evidence on how People Debug Computer Programs., *Int Jnl. Man-Machine Studies,* **7**, pp 151-182.
13. Hoc, J.M., [1983], Analysis of Beginners Problem-solving Strategies in Programming., in T.R. Green, S.J. Payne and G.C. van der Veer (eds.), *The Psychology of Computer Use.* Academic Press, London.
14. Kolb, D.A., [1976], *The Learning Style Inventory.* McBer and Co., Boston, Mass.

15. Kolb, D.A., Rubin, I.M. and McIntyre, J.M., [1979], *Organisational Psychology : an Experimental Approach*, Prentice-Hall, Englewood Cliffs.
16. Lucas, H.C., [1975], *Why Information Systems Fail.* Columbia University Press, New York.
17. Mayer, R.E., [1981], The Psychology of how Novices Learn Computer Programming. *Comp. Surveys,* **13**, pp 121-141.
18. McCosh, A.M., [1984], Factors common to the successful Implementation of twelve Decision Support Systems and how they differ from three failures., *Systems Objectives, Solutions,* ,**4**, pp 17-28.
19. Pask, G., [1976], Styles and Strategies of Learning., *Br. Jnl. Educ. Psychol.*, **46**, pp 12-25.
20. Robey, D., [1979], User Attitudes and Management Information System Use., *Academy of Management Journal*, **22**, pp 527-538.
21. Sheil, B.A., [1981], The Psychological Study of Programming. *Comp. Surveys,* **13**, pp 101-119.
22. Shneiderman, B., [1980], *Software Psychology : Human Factors in Computer and Information Systems.*, Winthorp, Cambridge, Mass.
23. Welty, C. and Stemple, D.W., [1981], Human Factors Comparison of a Procedural and a Nonprocedural Query Language., *ACM Trans. Database Systems*, **6**, pp 626-649.
24. West, R.F., [1982], A Construct Validity Study of Kolb's Learning Style types in Medical Education., *Jnl. Med. Educ.*, **57**, pp 794-796.
25. Zmud, R.W., [1979], Individual Differences and MIS Success : A review of the Empirical Literature., *Management Science*, **25**, 966-979.

NOTES FOR CONTRIBUTORS

The purpose of the journal will be to publish original papers in any field of computing. Papers submitted may be research articles, review artilces and exploratory articles of general interest to readers of the journal. The preferred languages of the journal will be the congress languages of IFIP although papers in other languages will not be precluded.

Manuscripts should be submitted in triplicate to:

Prof. G. Wiechers
INFOPLAN
Private Bag 3002
Monument Park 0105
South Africa

Form of manuscript

Manuscripts should be in double-space typing on one side only of sheets of A4 size with wide margins. Manuscripts produced using the Apple Macintosh will be welcomed. Authors should write concisely.

The first page should include the article title (which should be brief), the author's name and affiliation and address. Each paper must be accompanied by an abstract less than 200 words which will be printed at the beginning of the paper, together with an appropriate key word list and a list of relevant Computing Review categories.

Tables and figures

Tables and figures should not be included in the text, although tables and figures should be referred to in the printed text. Tables should be typed on separate sheets and should be numbered consecutively and titled.

Figures should also be supplied on separate sheets, and each should be clearly identified on the back in pencil and the authors name and figure number. Original line drawings (not photocopies) should be submitted and should include all the relevant details. Drawings etc., should be submitted and should include all relevant details. Photographs as illustrations should be avoided if possible. If this cannot be avoided, glossy bromide prints are required.

Symbols

Mathematical and other symbols may be either handwritten or typewritten. Greek letters and unusual symbols should be identified in the margin. Distinction should be made between capital and lower case letters; between the letter O and zero; between the letter I, the number one and prime; between K and kappa.

References

References should be listed at the end of the manuscript in alphabetic order of the author's name, and cited in the text in square brackets. Journal references should be arranged thus:

1. Ashcroft E. and Manna Z., The Translation of ‘GOTO’ Programs to ‘WHILE’ programs., *Proceedings of IFIP Congress 71,* North-Holland, Amsterdam, 250-255, 1972.
2. Bohm C. and Jacopini G., Flow Diagrams, Turing Machines and Languages with only Two Formation Rules., *Comm. ACM,* **9**, 366-371, 1966.
3. Ginsburg S., Mathematical Theory of Context-free Languages, McGraw Hill, NewYork, 1966.

Proofs

Proofs will be sent to the author to ensure that the papers have been correctly typeset and *not* for the addition of new material or major amendment to the texts. Excessive alterations may be disallowed. Corrected proofs must be returned to the production manager within three days to minimize the risk of the author's contribution having to be held over to a later issue.

Only orginal papers will be accepted, and copyright in published papers will be vested in the publisher.

Letters

A section of “Letters to the Editor” (each limited to about 500 words) will provide a forum for discussion of recent problems.