A COMPARISON OF PROCEDURAL PROGRAMMING ACHIEVEMENT AND OBJECT-ORIENTED PROGRAMMING ACHIEVEMENT IN RELATION TO COGNITIVE SKILLS.

by
Terry Dwyer
BAppSc, BEd, GDipEd.

A thesis submitted in partial fulfilment
of the requirements of the degree of
Master of Education (Honours) of
the University of New England

June, 1997

Certificate of originality

I certify that the substance of this thesis has not already been submitted for any degree and is not being currently submitted for any other degrees.

I certify that to the best of my knowledge any help received in preparing this thesis, and all sources used, have been acknowledged in this thesis.

[redacted]

Signature

Acknowledgments

The completion of this effort would not have been possible without the help of a select number of people.

The Year 12 Information Processing and Technology students of Stanthorpe High School through 1994 and 1995 are thanked for being willing participants in this study. The Principals of the school, in 1994 and 1995, are also thanked not only for giving their permission to carry out this study but also for their strong support and interest in the study.

The guidance in design and statistical analysis provided by Dr Ken Vine contributed to the work.

I want to thank my wife, Julie, and son Mark who endured the long hours spent on this work. The support from Julie, not only morally, but also in correcting punctuation, sentence structure and grammatical errors was of immense and indispensable help.

To Dr Maurice Anker who inspired, encouraged, and challenged me, I owe a special debt of gratitude. I am especially indebted to Dr Anker and Dr Vine for their willingness to carefully examine the numerous chapter drafts and redrafts.

Finally I offer my thanks to the many people who inquired about the study and who quietly encouraged me to persevere with the study.

Abstract

The recent advent of programming languages incorporating both procedural programming and object-oriented capabilities at a reasonable price for schools raises the question of whether an exposure of Year twelve information technology students to object-oriented programming is an appropriate pedagogical objective. The potential of object-oriented programming to inherit user interfaces formed the contextural justification for also considering the possibility that the addition of sophisticated user interfaces would enthuse students and provide more stimulating and exciting developmental work.

An information processing model based on cognitive skills and motivational factors provided the framework within which student achievement in traditional procedural programming was compared with student achievement in object-oriented programming. Forty-eight Year twelve students, randomly assigned to procedural programming and object-oriented programming classes, were administered the Learning Style Profile. Achievement was measured in three areas of programming competence: knowledge of syntax, program modification, and program composition. Attitude towards programming was measured by the use of a liking of programming subscale, a programming difficulty subscale, and a programming usefulness subscale.

The findings indicate that while there is no significant difference in student achievement between each instructional treatment, the cognitive demands of each programming environment do differ. Object-oriented programming achievement in the areas of program modification and program composition is significantly related to cognitive skill factors of simultaneous processing, persistence, and memory. The object-oriented feature of encapsulation will require instructional strategies which develop simultaneous processing skills. There is evidence to suggest that student skill in sensing an overall pattern from the relationships among components is amenable to change.

Despite the additional cognitive demands of object-oriented programming, it appears that students have no inherent difficulties in learning object-oriented programming. The study suggests that students would not feel that object-oriented programming is more difficult than procedural programming.

The majority of Queensland schools support the algorithms and programming topic, within the Year 11 and 12 Information Processing and Technology subject, with procedural programming (Clarke 1992, 4; King, Feltham and Nucifora 1994, 21). Within this context, a sequence in which students experience procedural

programming and then later use a sophisticated user interface within an object-oriented programming environment may be an acceptable transitional curriculum evolution. This intermediary position in which students study both procedural programming and object-oriented programming has some justification in that two programming languages enhance the problem solving approach and students' view of the use of computer systems for problem solving (Lawson 1985a, 541; Lawson 1985b, 542; McGrath 1988, 467-484).

The exploration of the relationships between cognitive skills and object-oriented programming achievement does provide some guideline for the design of instructional strategies and learning experiences. It also promises to improve student success because learning problems are more frequently related to the type and level of the cognitive processes required to learn the material rather than to the difficulty of the subject matter (Letteri 1988, 22).

Contents

List of Tables

List of Figures