

COMPUTER EDUCATION IN CIVIL
ENGINEERING-IS IT WORKING?

I. C. GOULTER,
DEPARTMENT OF CIVIL ENGINEERING,
UNIVERSITY OF MANITOBA,
WINNIPEG, MANITOBA, R3T 2N2.

INTRODUCTION

The tremendous advancement in computer technology over the last decade has had a direct impact on all fields of engineering. These advancements have affected not only the manner in which problems are solved but also what problems can now be solved. These changes, however, have not affected all engineering disciplines to the same degree or in the same way. Of all the engineering fields, Civil Engineering has perhaps been the least affected. There have, nevertheless, been significant developments in the use of computers in Civil Engineering applications. Unfortunately, over this period of rapid development of computer use, a divergence between the engineering applications of the computer and engineering education in the use of computers has occurred. Developments in civil engineering education in computers appear to have lagged significantly behind the developments in computer use. It is recognised that computer use has exploded and consequently it is unreasonable to expect education to keep abreast of all developments. Nevertheless, the gap between use and education should not have grown to the extent to which it has now reached. In the following sections this divergence is discussed and ways to improve the current situation are suggested.

PROBLEM DEFINITION

When questioned about use of computers, graduating students from civil engineering departments often express a strong sense of alienation from computers. This sense of alienation also occurs in the earlier years of the civil engineering student's academic career but since the university environment is probably the last place where these students can achieve some confidence in the use of the computers it is more troubling when expressed by the graduating students.

The question that should be addressed is why do these students develop such a strong sense of alienation from what is basically a very logical and straight forward subject. The question is made more puzzling when it is observed that many students own their own programmable calculators and experience no difficulty in their use.

Before the question of how this alienation has developed is addressed directly, it is first necessary to review the current procedures by which civil engineering undergraduates are introduced to computers. All civil engineering undergraduate curriculums have one or possibly two core courses which are directed towards the learning of at least one computer language. The writing and running of some simple programs is associated with the course(s). The formal computer education normally finishes at this point and any further development in computer skills depends on the personal interest

of individual students in the taking of other advanced computer course electives or the continued exposure to programming in other courses taken during the student's academic career. At this point it should be noted that the two possible alternatives for continued education to a large extent depend upon student choice.

The student who wishes to enhance his computer expertise has ample opportunity to do so. The problem lies with the student who does not choose to advance his own computer education. This individual still has a good possibility of being called upon to use computers in some form when he graduates. It is extremely optimistic to expect that the student who is graduating with only one or maybe two courses in computer programming and the possibility of some peripheral exposure in computer usage in other courses to have even basic competence. This lack of competence in computer use can actually show up during the student's academic career. In a survey of 4th year Civil Engineering students at the University of Manitoba, 74% of those enrolled in one course, expressed severe misgivings about their own ability to develop and program basic computer programs associated with the course. In fact, it appeared that a large number of students may in fact be deterred from taking a course that requires computer application. This problem besets even the brighter students.

The problem therefore can be stated quite succinctly as how can a civil engineering curriculum be designed to ensure that the graduates have a basic competency and confidence in computer use?

STUDENT PERCEPTION OF COMPUTER EDUCATION

Part of the answer to the question outlined above can be obtained from the students themselves. One of the first things that becomes evident in discussions with the students is that the students tend not only to recognize their own lack of basic competence but also to fear using the computer. The students have no confidence in their own abilities to obtain the answer from the computer. In order to obtain a more precise view of the students' concerns about their undergraduate computer education a survey was conducted among fourth year civil engineering students who had participated in two elective courses in which significant computer use was required. The results of this survey are summarized in Table 1.

The students who replied to the survey were drawn from two separate courses. Some students had enrolled in both courses. In one course they were required to use a prewritten or "canned" program and then interpret the results. In the

TABLE 1
SURVEY RESULTS

		Students Checking This Point*
Program Preference	Canned	11%(4)
	Write-Your-Own	37%(13)
Computer Use Preparation	Sufficient Coverage in Introductory Course	54%(19)
	Insufficient Coverage in Introductory Course	46%(16)
Suggestions For Improvements In Curriculums	Changes in Introductory Material	37%(18)
	More Exposure To Programming	86%(30)
	More Exposure To "Canned" Programs	51%(18)

* () Number of students who checked this point.

other course the students were required to develop their own programs. In the latter course the development of the program included the rather time consuming and at times extremely frustrating debugging process.

Table 1, however, shows quite clearly that the students are far more comfortable with the "write-your-own" approach when it is compared to the "canned" program approach. This preference occurs in spite of the fact that the "write-your-own" programs are generally more difficult and time consuming for the students than the "canned" program. This observation is also important when it is recognized that there is a tendency in the advanced undergraduate courses to use packaged programs in preference to "write-your-own" programs. To some degree, this tendency is based upon the complexity of the problems to be solved and the lack of time available for the students. Even though "write-your-own" programs may be less realistic with respect to actual engineering and more difficult in terms of student time to develop, the students, nevertheless, seem to prefer them.

Some of the reasons for the preference of the students for "write-your-own" programs became evident in the survey. The survey participants commented that as they knew exactly what the program was doing, they were more capable of understanding the results. Further discussions with the students regarding the computer assignments also indicated that they recognized that the development of the computer program required them to understand explicitly the assignment and its method of solution. This in itself is an educational aid as it develops a more complete understanding of the course material.

A large portion of the students also indicated a great deal of satisfaction in completing "write-your-own" computer assignments. It seems that the more extensive challenge provided by the "write-your-own" assignments provides a greater amount of satisfaction than that provided by the "canned" programs.

The above discussion has focused upon the more positive aspects indicated by the survey. It does seem to indicate that computer education is working to the degree that students recognize the educational component of some computer-based problems. There are, however, a number of areas of concern that were brought out by the survey. These areas have been introduced in previous sections but will now be discussed in detail.

AREAS OF CONCERN

Examination of the survey results in Table 1 shows two

major areas of concern. The first concern arises in the first exposure to computer programming in the university environment. Many students (46%) consider that the introductory computer course to be inadequate preparation for computer usage especially when it is taken in isolation. These perceived inadequacies are compounded if the student is not exposed to computer use in succeeding courses. At the University of Manitoba, the introductory course in Civil Engineering is given in the second year. There is very little subsequent exposure to computer use in other second year courses or in third year courses. Consequently, any material that is learned in the second year course has a very good chance of being forgotten by the fourth year through lack of application. This concern also holds true for the brighter students who tend to be better prepared than the poorer students by the initial computer course. This is exactly the situation that was observed in the computer assignments for the fourth year courses.

The lack of computer applications in the intermediate years of the engineering program not only provides an opportunity for the material to be forgotten but also prevents any further development of programming skills. It is the issue of the development of programming skills that leads to the next area of concern. Knowledge of a computer language is not the only prerequisite to being a successful computer user in the engineering context. The ability to either develop new programs or improve or even use existing programs is dependent not only on knowledge of computer languages but also on the philosophy of debugging and checking the programs.

The type of program the students normally encounter in the introductory computer course is rather a trivial problem, in which errors are easily located and solved. This type of approach is of course necessary as the course generally emphasises the learning of a computer language. The examples must therefore be reasonably simple so that the language itself can be practised. One of the major problems in computer use however, lies in the debugging aspect of program development or "canned" program use. There is a certain philosophy attached to the debugging of programs. If the students do not learn how to debug programs, then it may be asserted that computer education is in fact not working. It is precisely this situation that is one area for concern outlined in this paper. One example that illustrates this problem is an inability or unwillingness to accept the evidence of errors in the input data. These input data problems are generally very simple to solve and yet the students are often at a complete loss as to how to approach the solution.

The question, therefore, becomes; how can it be ensured

that civil engineering students become competent in computer program development and use. At this stage, two methods present themselves. The first method is really a process of learning through continued exposure to "write-your-own" programs at all levels of the undergraduate program. This method requires a certain commitment to this process by all staff members teaching in the undergraduate program. Due to the varying levels of computer expertise among the staff members themselves, this method is not very certain of fulfilling the objective of the process.

The second method requires the introduction of an additional computer course. This course would be oriented towards allowing the student to develop more complex programs and thus be exposed to the philosophy of program development. Such a course is in existence at the University of Illinois at Urbana-Champaign and is recognized by the students as being particularly helpful in developing the skills required for efficient computer use. However, the course is an elective and a significant proportion of students are frightened away from it. There is, therefore, no guarantee that a majority of the students will learn the competence in programming that may be asserted as so necessary for a complete computer education. The course, therefore, must be mandatory or placed into the curriculum in such a fashion that a majority of the students take the course.

It is the personal belief of the author that the second method is preferable to the first method. One of the main reasons for this preference is that the continued exposure to programming essential to the first method should also be maintained. This exposure will complement the new course and as a result provide the students with an even better opportunity to develop a better all-round understanding of computer use.

PROGRAMMABLE CALCULATORS AND COMPUTERS

Part of the computer explosion has been a proliferation of programmable calculators and mini-computers. One of the issues that makes the concern regarding computer education in the undergraduate programs difficult to analyze is the sometimes surprising willingness and ability of the students to program their own calculators. This programming is not significantly different from the programming encountered on the larger machines but it seems to strike a more receptive chord in the student's minds. It is this paradox that seems to indicate that formalized computer education in the civil engineering undergraduate programs is not working very well at this point in time.

Some explanation for this paradox may lie in the comments arising from the survey on which this paper is based. As

indicated earlier, a large majority of the students indicated a preference for "write-your-own" programs which are exactly the type of programs developed on the programmable calculators. The recommendation for more use of "write-your-own" programs in the intermediate years at the undergraduate curriculum draws additional support from this observation. It may be asserted therefore that the objective of undergraduate computer training should perhaps be that the students develop the same competence with the larger computing machinery as they have with the programmable calculators.

STAFF RESPONSIBILITY

Up to this point, the problems have been discussed with respect to methods of changing courses and curriculums to ensure a more complete education. The responsibility of the individual staff members has not been addressed and it is this area that deserves some further attention.

One of the comments that arose in the survey suggested that faculty members should ensure that they are personally familiar with computer use. This comment is particularly relevant when the methods of how best to improve student competence as discussed in the previous section are examined more closely. Exposure to computer applications at all levels of the undergraduate program is an important component in the full development of computer use. If courses at the intermediate levels are to contain computer applications and in particular "write-your-own" program applications, it is obviously necessary for the staff members to develop or maintain their own competence. This does not mean that staff member is necessarily an expert but he must at least be sufficiently competent so that reasonable and yet challenging assignments can be developed.

One reason for discussing this concern lies in a tendency for staff members to allow the graduate students working with them to be the computer users. If the staff member has a problem he goes to his graduate students. This approach tends to cause the staff member to lose his skills in the same fashion as the undergraduate student who is not provided with an opportunity to polish his computer skills at all levels of the undergraduate program. One result of a situation in which the staff member does not have competence in computer use is the same attitude that afflicts many of the engineering graduates, viz, lack of confidence in the ability to use the computer and as a consequence a somewhat fearful approach to situations which require computer use. It is not unreasonable, therefore, to expect that staff members maintain their competence in computer use not only to improve the quality of the undergraduate education but also to develop their own competence.

EDUCATION FOR COMPUTER USE IN INDUSTRY

At this stage it is wise to recognize that part of computer training should be the development of a realization that the computer is only a tool for use in the solution of engineering problems. The answers provided by programs, especially the packaged programs often used in design, must always be examined to see if they are in fact realistic. The problem of new engineers considering computer results as absolute, not requiring additional information and analysis, is often voiced by practising engineers. A significant amount of the computer education should therefore be directed towards this end. To some degree, this objective may be best fulfilled by having the students use some "canned" programs and then analyze the results. On this basis, some combination of "canned" and "write-your-own" programs appears to be the best direction for improvement in computer education.

This observation is supported by two articles by Hayes (1981a,1981b) in which he stresses the importance of design experience during engineering education. The inclusion of computer programs used in design will facilitate this objective. These programs by necessity must be "canned" programs. Consequently some "canned" program use must be maintained in the undergraduate curriculum. The change in emphasis between "write-your-own" programs and "canned" through the various years of the undergraduate curriculum is suggested in Figure 1. This change in emphasis reflects the actual change in emphasis in the coursework itself, and as such complements the coursework material.

SUMMARY AND CONCLUSIONS

Computer education in civil engineering curriculums is experiencing some difficulty in providing an instructional system in which the students can learn both the technical aspects of computer programming plus the philosophy behind the development and debugging of new programs. Most introductory computer courses are aimed at the learning of a particular computer language. In order for the students to develop and understand the philosophy of computer program development, it is suggested that an additional computer course requiring the development of more complex programs be introduced into the curriculum in such a way that a majority of students end up taking it.

Additional exposure to both "write-your-own" and "canned" program in other courses in the curriculum will further develop computer expertise. The "write-your-own" programs are preferred to "canned" programs for clarifying course material, while "canned" programs are to be preferred when design experience is the course objective. A balanced combination of both will

Basic Principles
Write-Your-Own Programs

Design Applications
Canned Programs

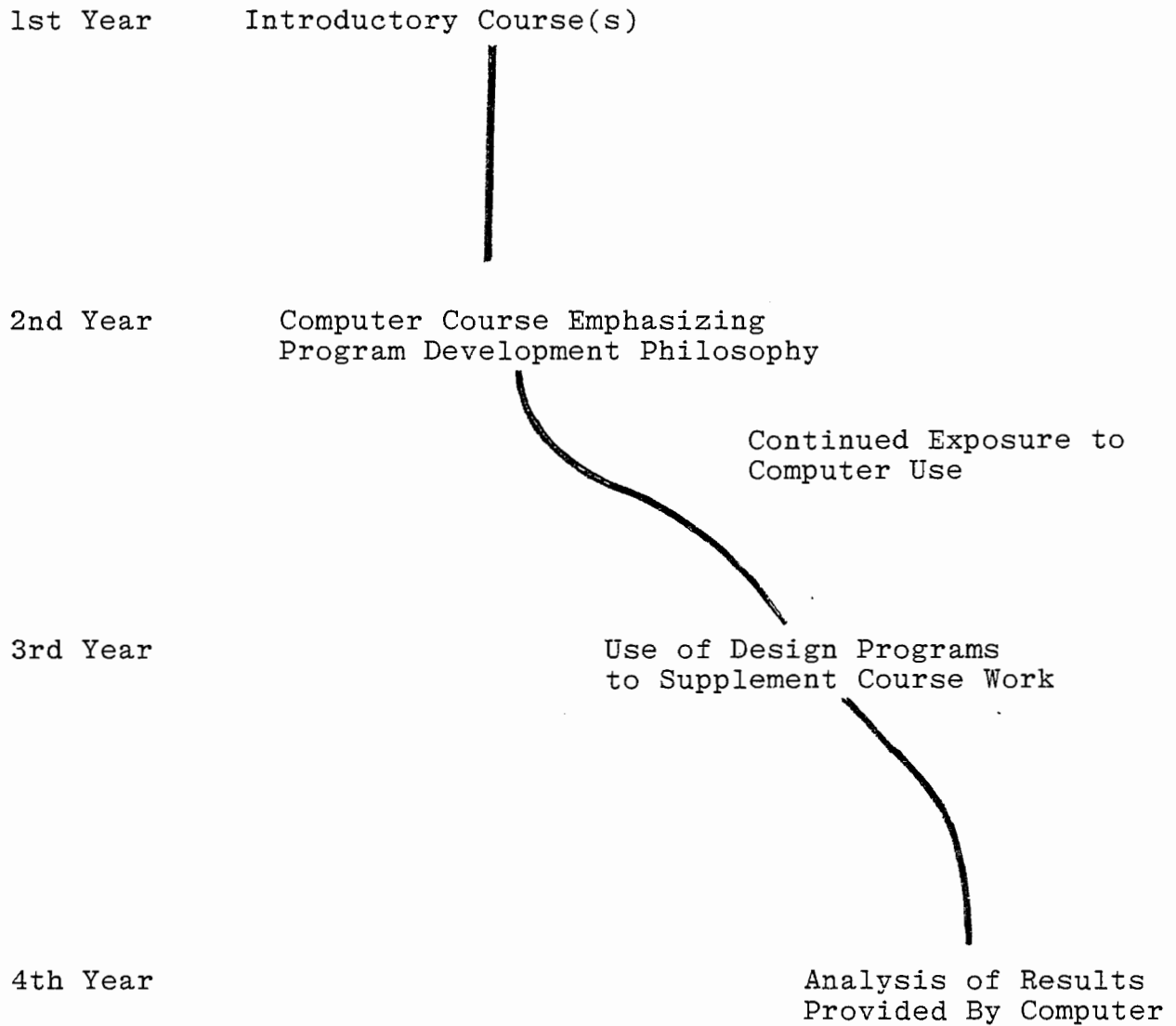


Figure 1 - Change In Computer Use Emphasis in Civil Engineering
Curriculums

provide an opportunity for the development of all round computer expertise.

It is also recommended that staff members also maintain or develop their own computer expertise so that they can set reasonable computer assignments for the students and simultaneously maintain or develop their own computer expertise.

It is believed that implementation of these recommendations will lead to a situation where students who are now graduating with minimal competence in computer use and in many cases actual fear of computer applications will instead graduate with increased competence in computer use and little or no fear of computer application.

Computer education has slipped behind the usage of computers in the civil engineering profession but the situation can be rectified by appropriate changes in both engineering curriculum and the attitudes of engineering staff.

REFERENCES

Hayes, J.M., (1981a) "Editorial Rambling" Indiana Civil Engineer, Spring, 1981.

Hayes, J.M., (1981b) "Engineering Pedagogy" Indiana Civil Engineer, Summer, 1981.