COMPUTER EDUCATION IN THE 1980s, A SOMBER VIEW

Dr. William Mitchell
Department of Computing Science
University of Evansville

Introduction

The discipline of computer science is a child of the 1970s. Its growth in infancy has been impressive, statistically, but so it is with infants. As we enter the decade of the 1980s, the discipline and all of us engaged in computer science education face some difficult choices. It is becoming abundantly clear that in the 1980s computer education cannot be provided for our students in the variety and quality which they demand. It will fall to us, personally, to decide what kind of computer education will be made available. In this next decade we will suffer a national deficiency of computer expertise equivalent to our national deficiency in oil. The cost of this expertise is already inflating at an alarming rate, and we have yet to begin to mobilize programs which in the long-term will stabilize the market. It is therefore inevitable that the 1980s will witness a frantic shift to alternative sources of expertise and a consequent dilution in the quality of computer professionals and computer products. The academic profession must make program decisions now which will serve to minimize the cost which our society will pay as it struggles to fully enter the computer age.

University Pragmatics

Expanding university programs face many problems in the 1980s. In an era of retrenchment, growth will be controlled. Every discipline is looking for new markets, new justification for maintaining resource allocations which are not really justifiable. Areas with solid growth potential, such as computer science, will be slowed in their development by the claims of contracting areas for revitalization capital (17). State and Federal allocations linked to student enrollments will reflect the fact that enrollments in the 1980s will be fairly stable.

But the importance of the computer in the society of the future is obvious to everyone, and particularly to the students.

No discipline in the university can continue to ignore the contributions which the computer brings to the analysis and presentation of the data of the discipline, however it is organized. A minimal familiarity with standard applications packages will become a requirement in most majors, if not a degree requirement (Dartmouth, Harvard) (9,33). Students and faculty who must utilize the computer will naturally wish to understand its processes better, so the demand for various levels of technical "service" courses will escalate. Who will teach these courses?

In addition to the "general education" type of offering, there are several special groups within the university who require specific programs of computer education. Prominent among these are the pre-service secondary and elementary teachers(23). Taylor, Poirot, Powell and Hamblen (30) have recently suggested that <u>every</u> teacher at the elementary and secondary levels should possess computer competencies in programming, analysis, educational applications, and societal impacts of the computer. This competence could hardly be attained in less than three courses (an introduction to computers and society, an introduction to programming and analysis, and a course in educational applications covering timesharing, micros, and CAI). Who will teach these courses?

The university must adjust its offerings and curriculum to serve the adult learner, the lifelong learner. The pool of 18-year-old students in the population will decline steadily through the 1980s. Therefore continuing education curricula need to be developed and implemented.
Pooch, Chattergy, Austing and Mulder (24) characterize these learners as those already having a basic degree but who want technical refresher courses to enable them to keep pace with a rapidly changing discipline. Such a student "desires a 'no nonsense' level of instruction and wants information which he can immediately use on his job." Clearly these cannot be the same offerings which are pursued by the undergraduate major. Who will teach these courses?

Favorable market conditions are precipitating an influx of students to major and minor in computer science programs. We have witnessed similar shifts in student vocational choices in the past in response to high salary opportunities, most notably in Engineering in the mid 1960s. Both Engineering and Business graduates currently share with computer science bright job prospects, and most prognosticators see a strong market through the decade. Consultant Arnold Deutch (38) sees a 20% increase in bachelor level degrees awarded in engineering and computer disciplines by 1986, despite the decline in 18-year-olds. But the demand for technical expertise is seen to be so broad that the shrinking pool of qualified students will be unable, in the short run, to supply sufficient numbers to satisfy all the need.

Currently the market is demanding all kinds of computer specialties, including the traditional academic areas of operating systems, artificial intelligence, graphics, etc. Therefore, there is no immediate need for computer science faculty to rennovate their offerings. The more practical emphasis of the ACM's Curriculum '78 may safely be ignored when ample numbers of students are willing to study the more theoretical areas of dominant interest to the faculty, secure in the knowledge that there is a growing market for such skills. The fabled industry/academia gap has not disappeared; there is just more industry involvement in areas of academic interest.

Market Pragmatics

The computer applications of the 1970s are mere precursors of the computerization of business, industry, government and education which will occur in the 1980s. Appetites for computer applications have been whetted, and the computer industry is certain to try to satisfy them. Business is already irrevocably committed to data processing and is close to the point of commitment on information management systems. Office automation will contribute to the pressure for complex, integrated information systems. The variety and economy of the hardware components available and their growing sophistication mandate a systems perspective well beyond the imagination of most of the practicing professionals. The business world is impatient to reap the benefits of current technology.

The computer industry produced more millionaires in the decade of the 1970s than any industry in history over a comparable period. Today we see a lot of investment capital available to promote computer products, both hardware and software. A highly profitable, high technology industry consumes large numbers of skilled, creative people. The opportunities to spawn new applications compete for talent with the efforts to implement tested approaches. These are the pressures which have created the current employment situation and the reason our graduates have experienced a 29% increase in starting salary offers in 1979 (14). Money is the major reason both computer science faculty and business faculty are being lured away from the university (27). Industry has both princely salaries and state-of-the-art equipment. The Feldman Report points out that researchers who switch to industry enjoy financial benefits yet have little change in lifestyle (15).

The market demand is spiraling. "Predictions that programming as a function and a profession would become obsolete have proved groundless. Clearly, the trend is in the opposite direction, and good programmers today need to be smarter and better trained than their counterparts in previous years (8)." In the 1980s fewer faculty and fewer students will be able to afford a teaching career. Hamblen estimates a current shortfall of post-secondary graduates of one-fourth the needed doctorates, one-sixth the needed bachelors and one-eleventh the needed masters, and offers little expectation that the imbalance can be corrected in the 1980s (16). Conte and Taulbee indicate that Ph.D. production in computer science has actually decreased in 1976 and 1977 (10). In 1979 we are told that there are 600 vacant faculty positions in computer science departments (twice the total annual Ph.D. production), and that over half of the doctoral receipents this year took non-academic jobs (36). The inability of universities to fill authorized positions means an inability to adequately pursue established programs, never mind initiating new ones. But when the university is not able to expand the supply of trained professionals to satisfy the market demand, that demand is intensified, and more of the existing faculty and students are lured away (20). When not enough qualified people can be found, less qualified people will be pressed into service.

What will break the cycle? Clearly business will have to bear the burden of training its own computer experts (1,19,39). We are already witnessing a boom in the commercial professional development organizations, and vendor training services will continue to grow as the market for their products expands. Professional seminar leaders are not usually academics, and they are quick to emphasize that their training sessions are not academically oriented. Businesses will hire professional trainers or press their own data processing personnel to convert operators and applications area specialists into computer para-professionals. These para-professionals will relieve the data processing staff of routine tasks (maintenance) so that they can be involved in the implementation of new applications and technology. Even so, many of the existing computer professionals are technically obsolete and will have their own problems with new techniques and methodologies (5).

What will be the price of this kind of computer education prevalent within the business community during the 1980s? Perhaps the computer industry will not expand as rapidly as predicted due to deficient customer appreciation of its products and services. Perhaps some catastrophic failure, traced to imperfect analysis or incompetent programming will limit computer applications in the same way nuclear power has been recently hobbled (32). Perhaps the development costs of new and complex systems only poorly understood will lead to management disillusionment similar to that associated with MIS systems in the early 1970s. Perhaps we will opt for simpler, less flexible, less serviceable systems which will barter humaneness for reliability.

Computer Education, Who Will Provide It?

This author and many others have been calling for diversification of service offerings by computer science departments (21,22). But can we choose to experiment with service courses when our majors come already introduced to programming in high school and the profession sorely needs extensive offerings for continuing education?

In a time of national need, ought we not direct all of our energies toward the production of professionals and researchers, even at the sacrifice of our non-professional market? Isn't it more important to develop professional masters programs (12,28) and develop undergraduate information systems majors (6,28), than to make available offerings for the general education of non-professionals? If we husband our resources to serve our own, where else can basic computer education be acquired?

Business schools are already teaching programming to their majors, and this practice will most likely have to expand. Yet their faculty are generally not interested in programming or systems analysis, and those that are technically proficient are also strongly pursued by industry. Therefore the business schools will be hard pressed to meet their own instructional needs and will not be interested in a service load. Most other disciplines are even less qualified to offer computing instruction even to their own majors, but they will have little choice. A psychology major who missed out on programming in high school will not care for the first course offered for majors either in computing science or business.

If basic computer education could be delegated to the high schools, there would be no need to provide it in the colleges, but this cannot be expected during the decade of the 1980s. We have yet to formulate a guide for high school computer curricula (the ACM Elementary and Secondary Schools Subcommittee is working in this direction). But even given clear direction, the public school systems are totally unprepared to implement such curricula (25). The National Science Foundation and the National Institute of Education have only recently begun to significantly fund studies for the broad incorporation of technology in pre-college mathematics education (13,31). Years of development lavished on TICCUT and PLATO have not brought centralized timeshared systems significantly closer to the schools (Dr. Bork calls PLATO "a dinosaur"(35)). Micro systems promise greater economy and flexibility, but they currently lack software and require substantial expertise to keep running (11). Schools of Education are not yet comfortable with CAI, much less instructing their students in the computer milieu and the intricacies of programming. When we observe that major business schools do not yet require exposure to computing of all their students (18), we can foresee that the decade of the 1980s could well expire before such a requirement becomes extensive even in pre-service mathematics education. Micro-computers will likely be in the classroom before the teacher is prepared to use them (2).

In the meantime, elementary, secondary, and college students will continue to be exposed to computers in increasing numbers, both in the schools and through recreational and hobby interests. The industry is quite able to provide versatile hardware systems for astonishingly low cost. Exposure to packaged software, will do little to develop computing skills, just as exposure to TV does little to develop understanding of electronics. But because these packaged systems are primitive in their software capabilities, many students will begin to dabble in programming. Unfortunately, lack of insight and direction makes most of this exposure useless in preparing for serious academic or commercial applications (19). Poor perspectives, habits and practices frequently have to be unlearned, causing these students to make slower progress in college than students who begin from scratch. For every student who possesses a natural talent for coding, there will be 10 who will pay dearly for the lack of comprehensive, systematic instruction in programming in high school. Even at the high school level the principle of marginal utility applies: those most competent to instruct computing will be needed (and will seek) to work with the technically able, while those teachers with marginal interest and ability will conduct the broader general education effort. Computer education will be provided, and it will be delivered to an ever expanding audience, but its quality in the 1980s will range from mediocre to terrible.

What can we as computer science educators do to minimize the waste? Do we have a responsibility other than to our discipline and our majors? Some will answer no. Some will say that each discipline is capable of acquiring expertise necessary to pursue its own applications. Some currently claim no interest or responsibility for service functions. But most, I hope, will be swayed by some combination of educational philosophy, societal exigency, or political reality. Most will ask "What can I do?" The Feldman Report addressed some general suggestions to universities, industry and government in response to the present crisis in experimental computer science. Following are some suggestions to departments of computer science concerning the approaching crisis in computer education (supplementing a list addressed to individuals in SIGCSE (3)).

1. Determine what is being done to train elementary and secondary teachers in your university. Identify faculty in the School of Education who are best qualified to press for better computer training and assist them in any ways possible. Make one computer science faculty member the liaison responsible for keeping the computer science department aware of the pre-service and inservice needs of pre-college teachers.

2. Identify other disciplines within the university which have encouraged their students to take computer science courses. Set up a committee representing these principal service markets and attempt to agree upon one or two courses which will minimally meet their needs. Design these courses to make heavy use of graduate and undergraduate teaching assistants, and ask the participating areas to assign some of their students to this task. Consider adjunct faculty for this purpose. Coordinate these service offerings with the development of computer applications within the user departments. Identify for these user departments CS graduate or undergraduate assistants who might be hired to provide technical expertise.

3. Create a committee with the School of Business (and the continuing education office) which will determine the need for continuing education within the local business community. Share responsibility in creating offerings which will most effectively meet this need. Again consider adjunct faculty for this purpose (4,7) (in fact, use this committee to identify local professionals who might be used as adjuncts).

Why should departments of computer science go to such lengths? Because the crises of the 1980s will pass. We will adjust both our resources and our needs to effect a balanced supply and demand. In the 1990s we will witness hardware and software accomplishments which will truly change the nature of education as well as the nature of the computer professional. Less training will be required to effect computer applications, and automated instruction will be economic and effective. But in the 1990s we will continue to believe that computer education is best provided by those of us who make it our specialty, no matter how broadly used the tool may become. Departments of statistics have all but dissappeared, though the study of statistics continues innearly every discipline. Were computer science to become fragmented, viewed merely as a means to an end, the undergraduate major would similarly disappear, and its loss would cripple the graduate program. Computer science departments must accept the responsibility for service instruction in the 1980s if only to insure the concentration of university resources for the continued development of the discipline.

Epilogue

The author does not have a crystal ball nor has he recently completed a national survey. The following references do include such surveys, but they primarily point to a growing consensus among a diverse body of experts in interpreting the available data. These experts are not individually pessimistic, but each is concerned in his own area. Each sees solutions available, but fails to consider that everyone's solution draws on the same limited resources. The resource being contended for in the 1980s is primarily the bachelor level computer science skill (schools, government, graduate schools and industry compete for this resource). Hopefully that degree represents an understanding of algorithms, technology and and methodology which makes for practical, economic and original computer applications. Presently, the applications we wish to implement and the tools we have to work with require this level of expertise. Historically, however, whenever more powerful tools have become available (for example, report generators), we merely turn to more complex problems which again exceed the normal capability of our tools.

There are many who believe that the problems outlined above will solve themselves and therefore are not worth our efforts or concern. If we are not yet able to replace human insight with mechanical intelligence, we can for the short term a) get the job done with less rigorously trained people, or b) discover that, indeed, the kind of training needed is actually easily acquired informally by untapped masses once the motivation is sufficient. The author would like to share this optimism, but after many years devoted to designing and presenting service courses, he cannot substantiate the assertion that if you scratch the average psychologist, economist, physicist or business major you will find a budding programmer/analyst. Hamblen states (16) that we are already overutilizing undertrained computer professionals, and this author asserts that we will continue this dangerous practice through the next decade. The cost of this practice is potentially staggering, and our best efforts, should we choose to extend ourselves, can only marginally mitigate the situation. Yet we must believe that every little bit will help.

Still others would like to separate data processing, information science, computer literacy, and other educational applications from the disciplines of computer science and computer engineering. Yet in pressing for the establishment of "territory" in the academic panoply (elbowing between applied mathematics and electrical engineering) we forget that for the short run we are the stewards of all that is known about the use of this machine(37). Whether we like it or not, people both within and without the university expect that we will provide the expertise and the perspective required to develop the broad spectrum of applications which will be tackled in the 1980s. The less seriously we take these problems, the more grave the consequences for the society and the culture we all share.

REFERENCES

1. "A Crucial Commitment," COMPUTERWORLD, November 19, 1979, p. 30.
2. Aiken, Robert M. "What A Problem We Have With Microes," Interface, Vol. 1, No. 3, Fall 1979, pp. 24-25.
3. Aiken, Robert M. "Chairman's Message," SIGCSE Bulletin, Vol 11, No. 4, December 1979, pp. 1-2.
4. Archibald, J. A., Jr. "A Modest Plan to Ease the Shortage of Computer Science Faculty," SIGCSE Bulletin, Vol. 11, No. 4, December 1979, p. 6.
5. Beeler, Jeffry, "Programmers Found "DP Illiterate'," COMPUTERWORLD, October 8, 1979, P. 1.
6. Beeler, Jeffry, "January Workshop to Begin Tackling University Curriculum for Business DP," COMPUTERWORLD, December 10, 1979, p.2.
7. Benson, Robert J. and Robert A. Rouse, "A Community Based Professional Development Program In Data Processing," SIGCSE Bulletin, Vol. 11, No. 1, February 1979, pp. 136-139.
8. Bond, Robert J., "DP Will Change, But Support Still Necessary," COMPUTERWORLD, September 17, 1979, p. 45.
9. Burns, Bruce H., "The Role of Computer Studies in the Undergraduate Curriculum," Interface, Vol. 1, No. 3, Fall 1979, pp. 36-38.
10. Conte, S. D. and O.E. Taulbee, Production and Employment of Ph.D.s in Computer Science, University of Pittsburgh, 1978.
11. Correa, Marilyn, "The Impact of Microcomputers on Education," Interface, Vol. 1, No. 3, Fall 1979, pp. 43-51.
12. "Dartmouth to Offer Master's in DP Field" COMPUTERWORLD, July 16, 1979, P. 4.
13. Deringer, Dorothy K. and Andrew R. Molnar, eds., Technology in Science Education: The Next Ten Years, (NSF SE 79-57) July 1979.
14. Dooly, Ann, "Hot Prospects Seen For DPers," COMPUTERWORLD, September 24, 1979
15. Feldman, Jerome A. and William R. Sutherland, "Rejuvenating Experimental Computer Science," CACM, Vol. 22, No. 9, September 1979, pp. 497-502.
16. Hamblen, John W. Computer Manpower Supply and Demand, Information Systems Consultants, St. James, Missouri, 1979.
17. Hamblen, John W., "Computer Science Ph.D. 'Gut' Nonexistent," COMPUTERWORLD, April 2, 1979, p. 24.
18. Karten, Howard A., "'B" Schools Found Encouraging DP Studies," COMPUTERWORLD, August 13, 1979, p. 18.
19. Kirchner, Jake, "Software Woes of '70s Seen Continuing in '80s,"COMPUTERWORLD, December 10, 1979, p. 13.
20. McCracken, Daniel, Peter J. Denning and David H. Brandin, "An ACM Executive Committee Position on the Crisis in Experimental Computer Science," CACM, Vol. 22, No. 9, September 1979, pp. 503-4.
21. Mitchell, William, "Another Approach to Service Courses," SIGCSE Bulletin, Vol. II, No. 1 February 1979, pp. 6-9.
22. Mitchell, William, "The Case for 'Paper Programmers'," Interface, Vol. 1, No. 4, Winter 1979
23. Molnar, A. R., "The Next Great Crisis in American Education: Computer Literacy," AEDS Journal, 1978, Vol. 12, No. 1, pp 11-20.
24. Pooch, Udo W., Richard H. Austing, Richard Chattergy, and Michael C. Mulder, "Computer Science and Computer Engineering Education in the 80 's," Computer, Vol. 11, No. 9, September 1978, pp. 69-83.
25. Poirot, James L., "Computer Education in the Secondary School: Problems and Solutions," SIGCSE Bulletin, Vol. 11, No. 1, February 1979, pp. 101-104.
26. Schultz, Brad, "DP Scientists Fleeing Universities," COMPUTERWORLD, July 2, 1979, p. 2.
27. Schultz, Brad, "Academics Seen Flocking to Industry Through '80s," COMPUTERWORLD, September 17, 1979, p. 1.
28. Schultz, Brad, "ACM Committee Cites Lack of University DP Programs," COMPUTERWORLD, November 5, 1979, p. 1.
29. Schuyten, Peter J., "Firms Mount Hard Rush For Software Specialists," N. Y. Times News Service, November 26, 1979.
30. Taylor, Robert P., James L. Poirot, James D. Powell, and John Hamblen, "Computing Comptencies for School Teachers," Proceedings of the National Educational Computing Conference, Iowa City, Iowa, June 25-27, 1979.
31. The Improvement of Mathematics Education Using Information Technology (Program Announcement), National Science Foundation and The National Institute of Education, Fall, 1979.
32. Whitmarsh, John, "DPers Viewed as 'Nuclear Engineers' of '80s," COMPUTERWORLD, June 11, 1979, p. 13.
33. Woodruff, Jay, "New Harvard Curriculum Includes DP Requirement," COMPUTERWORLD, May 28, 1979, p. 15.
34. Woodruff, Jay, "Demand for DPers Continues to Grow," COMPUTERWORLD, August 27, 1979, p. 1.
35. Woodruff, Jay, "Plato in Eye of Florida Education Controversy," COMPUTERWORLD, December 10, 1979, p. 17.
36. Woodruff, Jay, "Afips Backs Proposed Study of DP in Education," COMPUTERWORLD, October 15, 1979, p. 23.
37. Whitmarsh, John, "Minnesota Breaks Ground With DP Literacy Test," COMPUTERWORLD, June 18, 1979, p. 22.
38. Deutsch, Arnold R., "Hot Jobs for the 1980s," Graduating Engineer, Fall 1979, McGraw-Hill Publications Company, pp. 8-11.
39. Slaughter, Gary, and Jack Stone, "The Changing Face Of DP Training," COMPUTERWORLD, October 15, 1979 (first of 4 parts on successive weeks).