THE STATE OF COMPUTER SCIENCE in NATIONAL COLLEGES AND UNIVERSITIES

by
J. A. N. Lee
Professor of Computer Science
Virginia Tech

Report No. CS830023-E
June 27, 1983

THE STATE OF COMPUTER SCIENCE in NATIONAL COLLEGES AND UNIVERSITIES

by
J.A.N. Lee
Professor of Computer Science
Virginia Tech

83/06/27

ABSTRACT

The author served on the State Council for Higher Education in Virginia (SCHEV) task force on computer science education during the academic year 1983; this report was originally offered as the first chapter in the report of that task force. The task force felt that the chapter would be too long and would detract from the remainder of the report and findings on the state of computer science in Virginia.

In this author's opinion the state of computer science (a generic term which is intended to cover computer science per se, information science and data processing) in Virginia is probably little worse than in the rest of the nation, but there are no pinnacles of excellence. The pragmatic nature of the senior institutions may be an advantage in light of the need for technology transfer in the industrial environment of the state. The four year colleges are by no means meeting the needs of the industry and are graduating only a small proportion of the number of graduates needed by industry. On the other hand, this report suggests that the need expressed by the industry is exagerated; the demand should be for highly qualified applications programmers rather than computer scientists.

The community college program is clearly meeting the needs of their communities though there may be some question as to whether the level of expertise in those localities and the consequent level of education of students will permit them to have any sensible mobility in today's society. One major result of that parochialism is the inability of students from community colleges to transfer effectively to four year institutions. While this is not a major mission of the community colleges the number of students who believe this to be a means of gaining a bachelor's degree is too large to ignore.

The reader should refer to the Task Force report for the opinion of the task force on the condition of computing in the Commonwealth.

CONTENTS

INTRODUCTION

Any examination of the state of higher education in the field of Computing in 1982 must be based on the objectives which have been set previously either in early studies or by mutual acceptance of goals set by others. Within the Commonwealth of Virginia, Computer Science has grown from need, instigated by several institutions in response to demands imposed by the environments of those institutions. In several cases, these demands have come from the faculty of the institutions, in recognition of their responsibilities to provide a broad based educational program to their students, while in other cases the number of enquiries for course offerings in the area can suggest the profitability of such programs of study. Peer pressure from sister institutions (both within and without the Commonwealth) and the yearning to satisfy personal needs by satisfying consumer needs can lead to the development of proposals, and eventual implementation of educational programs in almost any field. Computer Science, in any of its myriad implementations, has been no stranger to this phenomenon. Like the space furor of the late 50's and early 60's, Computer Science has been the glamor field of the sciences, bolstered by predictions of mind-boggling break-throughs which will turn around the whole realm of human endeavor. Whereas the space program required extensive facilities which could not always be obtained (or attained) on a campus, computer science could be initiated on a shoe-string, piggy-backing (and sometimes justifying) academic facilities on administrative systems. College administrations do not need a rocket to operate their institution! Today, that ability to start small is even simpler with the advent of the micro-processor and the personal computer.

Individual institutions in Virginia have probably conducted prior self-studies and SCHEV has been subjected to many arguments for the approval of the implementation of Computer related programs in Community Colleges and Universities, but the present study is the first State-wide study that has been conducted. Thus the only basis for judging the achievement of objectives must be the nation-wide studies which have been published elsewhere. Similarly, state needs (both governmental and private industrial) must be estimated on the basis of the prognosis of national needs.

PRIOR PREDICTIONS OF DOOM

No-one likes a bearer of bad news, and so it is perhaps understandable that numerous predictions of doom which have been the results of studies of computer science since the early 1960's have not been acted upon. Like the prediction made in approximately 1920, that every woman in the United States would have to operate a telephone console by 1960, due to the enormous increase in telephone usage, and the consequent overturning of that prediction by the advances in technology (namely the "automatic" exchange), so many have expected technological break-throughs in computing which would obviate the need for more man-power. At the same time, there has been a continual expectation that "automation" would relieve the manpower pressures in other areas whereas in fact there has been an application of "Parkinson's Law" in which the consumer of the automated industry has expected faster turn-around and more efficient service. This has resulted in a compaction of the time-scale of operation and an increase in manpower needs that directly offsets any predicted decline which would have taken place if the existing time-scales had been maintained. Break-throughs in computing have taken place, but they are mainly in the field of hardware design and implementation, while the software side has been hard pressed to maintain the status-quo in the face of new hardware. Most computer programs in U.S. universities and colleges emphasize mainly the software (or theoretical) aspects of the field merely because of their inability to finance their maintenance of a state-of-the-art hardware facility.

<u>The Pierce Report</u>

In 1965 the National Academy of Sciences sponsored studies which led to a report entitled "The Digital Computer Needs in Universities and Colleges", commonly known as the "Rosser Report". This study was targeted to the research needs for computational assistance in the scientific disciplines and omitted any direct references to the use of computers in the educational process, including the notion of computational support for a special area of study such as Computer Science.

While this report expressed the urgent need to provide computational facilities in Universities and Colleges, it did not give any direction for either (i) supporting the educational needs of the disciplines which it addressed with respect to research, thus muting its effectiveness on the basis that research itself would not generate computer-knowledgable staff directly, or (ii) emphasizing the need for support for the fledgling field of Computer Science. Thus the President's Science Advisory Committee established a panel under the chairmanship of John Pierce and produced a report in 1967 entitled "Computers in Higher Education". The major recommendations of this panel were:

1. We recommend that colleges and universities in cooperation with the Federal Government take steps to provide all students needing such facilities with computing service at least comparable in quality to that now available at the more pioneering schools.

2. We recommend that colleges be encouraged to provide adequate computing through government sharing of the cost. Such governmental cost sharing should include special grants to cover transient costs when service is being initiated or larger facilities are being installed. It should also provide a portion of the annual cost of continuing service.

3. We recommend that the present DOD-NSF agreement be extended to other government agencies and private supporters and include both capital and operating cost grants. Additional Federal funds should be made available immediately for support of computing service used for education and unsponsored research activities at institutions presently having the required facilities.

4. We recommend an expanded faculty training program to provide adequate faculty competence in the use of computing in various disciplines.

5. We recommend that the Federal Government expand its support of both research and education in computer

sciences.

6. We recommend that the Government agencies which support computing allow the schools to be free to apply the funds either to the purchase or rental of equipment and the support of staff, or to the purchase of service.

7. We recommend that universities and the Government cooperate in the immediate establishment of large central educational computing facilities capable of serving several institutions.

8. We recommend that universities and colleges develop and use accounting procedures which accurately measure the cost and utilization of computer services. With such information the allocation of computer time for research and education and the anticipation of associated costs should be made on a realistic and measurable basis.

9. We recommend that NSF and the Office of Education jointly establish a group which is competent to investigate the use of computers in secondary schools and to give the schools access to past and present experience. Cooperation between secondary schools and universities, and particularly providing service to secondary schools from university centers, should be encouraged.

10. We recommend that the Federal Government collect meaningful data concerning computers and the jobs, personnel, and education facilities associated with them, and endeavor to make useful annual forecasts.

In the fifteen years since that report was published there is little evidence to show that any of its recommendations were followed or supported. While it is true that many institutions did expand their computational facilities during this period, it was in the main merely to keep up with each other and to ensure that the administrative side of the academic process was well served. There have been a number of institutions which have been able to keep ahead of computational progress, but these were those institutions already in the forefront of other scientific activities. Apart from the aerospace program, the majority of technological/scientific programs in Universities and Colleges can offer the student experience in an experimental science which is abreast of the state of the art and research potential which is ahead of current practice. On the other hand they cannot provide the involvement in practice on the same scale as is possible in Computer Science. That is, most engineering and science programs can only model the real world; in Computer Science real computers are used and real problems can be solved in the University environment. On the other hand, those real problem solutions are constrained by the lack of up-to-date facilities. Many grocery stores and banks have more sophisticated equipment than is available to the general undergraduate. Add to that the fact that the majority of Computer Science faculty have no industrial experience and that teachers of computer science low level courses are generally not trained in the field at all but are in fact specialists in other areas who are attempting to "retread" themselves into a new market, indicates the depth of the problem. The adage of "physician heal thyself" is not possible.

The Feldman Report

The status of Computer Science was broached again in 1979 in the form of a report to the National Science Foundation by Jerome Feldman and William Sutherland. This report emphasized the concept that Computer Science is an experimental science much like other traditional sciences and as such needs nurturing in a like manner. Also of concern was the lack of movement of qualified teachers into the educational field and the recruitment of faculty from Universities and Colleges into industry. Amongst the recommendations of this report were:

Universities

- Recognize the special resource needs of experimental computer science.

- Use appropriate criteria in evaluating experimental computer science programs and faculty.

- Encourage cooperative programs.

Industry

- Exchange and share people and technology with universities.

- Provide funds and equipment, possibly through a private foundation.

Government

- Reconsider tax incentives, patent policy, and consent decrees.

- Develop funding of adequate scale and time horizon for experimental computer science.

- Designate a lead agency responsible for the national future in computing.

Criticism of this report included the contention that it was just another "Chicken-Little" prophesy of doom like the others before it, and that economic conditions would equalize supply and demand for employment in such a manner that the problems mentioned would solve themselves. This led to a series of statements of support, unfortunately and primarily, from members of the academic community. The argument was still bubbling in 1981 when Peter Denning, President of ACM, pointed to the subsequent report, jointly released by the National Science

Foundation and the Department of Education, entitled "Science and Engineering Education in the 1980's and Beyond" which states that there already exists a severe manpower shortage in most disciplines of science and technology, with computer science and engineering the worst affected. Denning argued that industry, being driven by economic motives, is "eating its own seed corn" by taking established faculty away from Universities and attracting the new graduates either directly into industry without further education, or taking new Ph.D. recipients into highly lucrative industrial positions. Not mentioned by Denning*, or any of the other reports, is that many faculty are finding that there is more freedom in industry today to conduct one's own research or to be involved in professional activities than exists in most universities. Moreover, industry researchers are not hampered by administrative tasks, policy committees and lack of clerical help. Industrial laboratories by necessity must keep up with the state of the art and the cost of such maintenance of currency is built into the cost of doing business which in turn puts a bigger bite into the cost of equipment which might be used to equip university and college laboratories.

----

* It is interesting to note that at the time that this report is being written, Denning has announced his own move from many years in academia to an quasi-industrial position.

The Snowbird Reports

Each two years the heads of Ph.D. granting programs in computer science meet in Snowbird UT to plan strategies for the upcoming biennial and where possible to lend strength to each others plans and proposals. Their report of the 1980 meeting was published under the title "A Discipline in Crisis" and noted the symptoms:

- About 250 new Ph.D.s graduated in 1979 (compared with 256 in 1975) as compared with 1300 positions seeking Ph.D.s.
- Fewer than 100 new Ph.D.s sought academic positions as compared with over 650 academic positions known to be open.
- Undergraduate enrollments doubled since 1975 with only nominal increase in lab space and faculty size over the same period.
- Intense competition for computer specialists has produced record salaries: new M.S. degree holders get offers matching full-year academic salaries for new Ph.D.s; experienced assistant professors get industrial offers as high as $45K.

and the immediate (short range) needs:

- Computing facilities capable of sustaining experimental research. (An appendix to this report shows that this requires a capital investment, per researcher, of $30K for good facilities to $75K for advanced facilities).
- More time for faculty to supervise graduate students in research.
- Increased support for graduate research assistants.

One must regret that much of the furor over the state of computer science is coming from groups which are generally considered to be academic in nature. There has never been a truly independent group consisting of responsible industry representatives that has considered these same problems and proferred their solutions which might be directed at their own companies rather than to either governmental (federal and state) agencies or universities and colleges.

The DoD Software Initiative [4]

Though emanating from a governmental agency, the proposed "Strategy for a DoD Software Initiative" points to some immediate problems for the Department of Defense and proposes solutions to those problems. While aimed at a specific problem area within computer science, there is no doubt that it will complement existing programs such as the Very High Speed Integrated Circuits (VHSIC) and Ada* programs within the DoD. This study indicates that "the U.S. gap between demand and supply is measured in terms of 50,000 to 100,000 software professionals, and if nothing is done, this gap will grow to 860,000 to 1,000,000 software professionals by 1990." Moreover, the report points out that besides the migration from universities and colleges to industry there is also now a movement of both personnel and technological leadership from the United States to other countries such as Japan, France* and Great Britain. To implement the strategy, the report recommends the establishment of a Software Engineering Institute which will provide the Dept. of Defense with an environment in which there can be developed management systems, standards and training programs in support of software development, with special emphasis on the technology transfer of

----

* Ada is a Registered Trademark of the U.S. Dept. of Defense.
* It may be significant, for example, that the final extensive stages of development of the Ada Programming Language were subcontracted to a French consortium over the competition of several U.S. based corporations.

the benefits of these programs to the (DoD) marketplace. Also included would be an educational element to provide "experiential training ... and ... training curricula."

Industry Initiatives

While the DoD approach may seem to be significantly new, in fact it is very similar to other programs which exist already in industry but is on a scale which reflects the size and needs of the Dept. of Defense. Wang Laboratories, for example, has established an institute for computer science education from which both they and the rest of the industry will benefit. Other smaller programs have been developed within other corporations to provide professional development for their employees (often using the very same professionals who peddle their professional development seminars in the open market) and to supplement the general education of employees who did not emerge from the computer programs at a university or college. In a sense, industry and government has already pronounced its evaluation of the state of computer science education by establishing their own programs and not basing these programs at universities and colleges.

There have been reported a number of industry initiatives which would improve the education of Computer Scientists in Universities. These include a funding system, currently in effect at the University of Lowell (MA) in which a number of concerned (local) industries have contributed 2% of their research and development budgets to the school to ensure that they will turn out the engineers and computer scientists that they need. Working through the Massachusetts High Technology Council, several other Massachusetts based companies have supported the state college system to sustain high technology growth. Even so, the Council predicts that the state will still only produce 40% of the needs of Massachusetts firms in the next four years!

The Wall Street Journal, 1983 January 15 [11].

A report entitled "Campus Gliche -- Universities in U.S. are Losing Ground in Computer Education -- Lack of Funds Leaves Schools with Too Few Teachers and Inadequate Facilities" reiterated many of the points made earlier in this report, with the added dimension of quotes from faculty at various institutions about their work-load and the "ten-year-old equipment" they have to use. Emphasizing the lack of space for well qualified students and antiquated facilities, the Journal quotes the Vice-Provost for Computing at the University of Washington as saying "This country could blow what is a terrific world lead in computer technology by failing to graduate enough people with the capability to maintain it." The report also notes that the Accreditation Board for Engineering and Technology has granted 31% fewer six-year accreditations this past school year on the basis of inadequate equipment. The University of California has tackled the salary problem by directly raising technical faculty salaries by between 20% for assistant professors to 5% for full professors.

As usual the only solution suggested appears to be for more federal support, though it is suggested that some industry-university joint programs would bear watching. These include a plan at Carnegie Mellon, with IBM, to provide faculty workstations, and others such as the University of Lowell.

WHAT IS COMPUTER SCIENCE?

With all of this clamor for improvements in the deliverance of graduates and technology from the academic field of computer science, it is worth taking stock of what precisely is meant by that term. The classic response from any scientist regarding a definition of his field is "that which I do!" Whereas in the 1950's and early 1960's it was possible to have something more than a rudimentary knowledge of the whole field, the field has expanded so greatly in the last 20 years that such a basic knowledge level is almost impossible to maintain. Not including the myriad applications of computers, computer science has now expanded to encompass numerous other fields each with its own specialities, and each vying to be the core of the subject area.

One of the difficulties facing educational institutions is to decide (determine) the objectives of providing studies in this field. To some institutions, Computer Science is part of their general offerings which provide a liberal education to the student body. It may be a service program much as English and Mathematics serve to support studies in the sciences and engineering, or it may be a totally independent field of study. From another point of view Computer Science can be considered to be an element of a professional school's offerings, the subject itself being at the center of a broad education, but presented at a level where specific implementations are not examined for their own sake, but merely as models of systems which contain specific notions and concepts. Finally, a program may be designed specifically to provide information with respect to specific products, their usage and maintenance, such that the student is well equipped to enter the job market with immediately applicable skills. Within the field which this study encompasses, all three objectives can be met by a comprehensive system of higher education although the totality of objectives may not be met by each and every program. It is to be expected that the goals of Universities, Professional Schools and Community Colleges will differ and the selection of courses of study and their modes of presentation will differ sharply.

DATA PROCESSING

Although the term Data Processing is a relic of the phrase Electronic Data Processing (EDP) which itself was coined by IBM in the 1950's to label its products (EDPM), this term, in the 1980's, is more likely to be used in connection with the applications of computers in the business world. Starting from punched card processing on unit record equipment, this field has finally entered the electronic age due to the reduced cost of equipment which can easily replace manual methods of card processing. The level of programming expected is not high and the languages to be employed are comparatively simple in form while having a scope which is broad and standardized. Day-to-day operations of computer systems are to be included in the field. Data Processing can be considered in the main to include those studies of the uses of computer systems (hardware and software) which are provided by a vendor, the management of those systems and the application of them to new administrative situations. Typically an educational program in Data Processing is closely allied with studies in accounting and management systems.

INFORMATION SYSTEMS*

While the processes of using a computer for problem solving are generally considered to be the fundamental task for which a computer was procured, the increasing complexity of organizations themselves requires the optimum use of the "inside" information regarding the organization to best make decisions with respect to the future of that organization. The monitoring of operations becomes more complicated, reporting requirements more extensive, and planning more difficult. Information systems have always existed in organizations, but with the availability of the computer, their use has been extended to the point of becoming a highly technical subject. This gives rise to the need for professional and technical personnel to staff these activities. Information Systems and managerial effectiveness are becoming interrelated and many institutions look towards the development of information systems as an investment in their managerial

----

* Based on the preface to the "Curriculum Recommendations for Graduate Program in Information Systems", R.L. Ashenhurst, (Ed.), ACM, 1972. See Ref. [5]

capacity. Information Systems personnel must work closely with top management and have the keys to decision making within their organization. Graduates from such programs need to have extensive knowledge and skills in personnel relationships, problem solving systems and model development, systems science, computer systems (hardware and software), organizational structures and societal awareness.

SOFTWARE ENGINEERING

Software engineering is the outcome of the application of systems analysis to the programming process and (bluntly) is the study of the codified techniques and skills of program development. Where a student might have wanted to graduate as a programmer in the late 1960's, the skills and knowledge he possesses in the 1980's should qualify and give him the attendant responsibilities of using the label "engineer". In much the same manner as the job classification "coder" has been relegated to imply the manual processes of translating highly detailed specifications into computer "code", so there is now the trend towards regarding the tasks of a programmer to be not as highly skilled as those of a software engineer. The movement of computer applications into fields which affect the health and welfare of humans requires a greater attention to the correctness and reliability of software than has existed previously. While it was always expected that hardware systems would be correct and reliable, there was a tendency not to have such high expectations of software. The tools for software testing are themselves soft, and the metrics against which software are to be measured are empirical and subject to interpretation. This state of affairs has changed in the past decade to the point where soft tools can now be used to ensure hard results and inspire a level of confidence which was not expected previously. Like Information Systems, Software Engineering is the result of an inward look at the field, in this case programming, and is a subject which is being supported by ongoing research in areas which would not have been supported as basic research in the 1960's. Emanating mainly from industry, Software Engineering is only now being accepted in Universities and Colleges as a teachable subject and is frowned upon by many outside of computer science since it does not provide a quick solution to short term problems. It is unlikely that software engineering will ever be a topic for high school studies but its effects must be an integral part of the freshman course of every computer science student.

COMPUTER ENGINEERING

As Software Engineering is to programs, so Computer Engineering is to hardware. Ensconced mainly in Electrical Engineering departments of Universities, there has been a traditional rivalry between Computer Engineering and Computer Science (in any of its variants) in most US institutions. This has had serious and deleterious effects and has been responsible for the lack of cooperation between hardware and software designs not only in colleges but also in industry. Each has had the attitude that they could implement the other's design better in their environment; thus hardware has not been responsive to the needs of software and software has not taken advantage of the full potential of the hardware. While Computer Engineering is specifically excluded from this study, it cannot be excluded from an overall study of Computer Science any more than any other allied field can be ignored.

Much of the real progress in computers, computing and computation is currently taking place or is being sponsored by advances in hardware; it is imperative that the softside of the house of computing catch up with the hardware side both technologically and socially.

COMPUTER SCIENCE(S)

Computer Science(s) must be considered to be either the core of the field of concern or the umbrella under which all others fit. Depending on the school and the specialities of the faculty, most Computer Science Departments profess to cover a core of the subject both in educational offerings and research. These include five areas which can be broadly classified as

(i) Computer architecture and systems,

(ii) Computer software and systems,

(iii) Computationally related mathematics and theory of computation,

(iv) Computer applications and service, and

(v) The societal impact of computer science and its history.

Naturally each of these areas can be further subdivided to include in (say) Computer software and systems such topics as Programming Languages (itself a topic which can be further subdivided several times), Operating Systems, Data Structures and Software Engineering. Further each area can be presented from several points of view: (a) as a user of a product, (b) as a designer/implementer of a product or (c) as a topic for theoretical review. Computer Science as general topic is much more easily included in a general educational program, and a presentation of topics from a user's point of view presents to an institution the service courses needed in other fields such as science and engineering. This can be an albatross around the neck of a Computer Science Department since the level of detail is probably much greater than that which would be expected of a major in Computer Science and the need to include software engineering design techniques is abhorred by the recipients who just want "to learn to program". This is a similar problem as has been faced (without significant solution) by Mathematics and Statistics programs for many years.

DE FACTO DEFINITIONS

There are four elements which tend to define what it is that a faculty member teaches and thus what is the effective curriculum in an institution: (i) curriculum guidelines available from <u>his</u> professional/educational society, (ii) the needs of local industry and the demand for graduates, (iii) the text books available from the publishing companies, and (iv) the experience of the faculty member.

Curriculum Guidelines

In 1982 there exist many different guidelines for curriculum each of which is targeted to a different career and specified by different professional/educational societies. The Association for Computing Machinery (ACM) has published seven such guidelines [5]:

Graduate Professional Programs in Information Systems
Computer Science Course Program for Small Colleges
Undergraduate Programs in Information Systems
Undergraduate Program in Computer Science
Master's Level Programs in Computer Science
Educational Programs in Information Systems
Associate Level Degree Program in Computer Programming

while the Data Processing Management Association (DPMA) has a Model Curriculum for Undergraduate Computer Information Systems Education [8a]. The Institution for Electrical and Electronic Engineers (IEEE) also supports a model curriculum [8b] although it tends to emphasize the engineering style aspects of the subject area.

Industry Needs

The lack of full-time faculty in computer science leads many institutions to supplement its faculty by adjuncts from local industry who can provide a close liaison between the needs of that industry and the educational institution. On the one hand this provides a continuing dialog between the industry and the institution regarding the acceptability of the curriculum (and in particular the one course being taught by the adjunct) while on the other hand opening up the curriculum to spontaneous (impulsive) modification to meet specific needs. The sheer lack of faculty evidenced by the presence of the adjunct infers that there is little supervision or oversight of the adjunct and without precise planning the intended course can turn out to be merely one person's biased view of that topic.

From an industrial point of view the teaching of specific skills leads to the immediate integration of a graduate from that course into the workplace without any consideration for the educational objectives of that course. Employability wins out over education and premature technical obsolescence. Only with a mutual cooperative program such as an advisory committee or visitors board can the needs of industry be evaluated and satisfied with forethought and care. Changes to curriculum which satisfy the needs of one local industry may destroy the advantages the students had with some other potential employer.

There is a strong need for industry to be involved in educational programs not only to help guide those programs but also to understand their responsibilities with respect to satisfying the further educational needs of their employees. For example, several institutions have removed credit for courses in specific programming languages and have replaced such courses for majors by a single course in the fundamentals of programming languages. Students are encouraged to program in a variety of languages, but the lack of credit does not enforce this policy. Thus employers must be prepared to accept students without specific skills but recognizing the deeper knowledge of fundamentals that will permit the learning of a diversity of languages in a short period.

The codification of the needs of industry which might be met by programs of study in educational institutions has not yet been accomplished, and thus industry is of little assistance in providing curricula guidance at this time.

A report on industry needs [10] by a faculty group needs in terms of job descriptions rather than prerequisites for those jobs. Their major conclusion was the need for more COBOL!

Text Books

Computer Science has gone from a stage in the mid-1960's when there existed very few textbooks, to a stage where there is a multiplicity of texts for the majority of undergraduate courses. The original hesitation to produce texts was due in part to some suspicion by publishers that they would be undercut by the free distribution of manufacturer's manuals and technical reports. However the lack of a forceful entity to impose curriculum guidelines on the educational institutions has resulted in a wide diversity of topics being presented in a textbook as being the relevant study subjects for the corresponding course. This has had the effect of either radically changing course contents as different faculty have used their favorite textbooks (often their own) or continually changing the curriculum as a newer and more up-to-date text has become available. Programming languages offer one short history of change in this matter. Initially a programming language text was merely a primer to accompany the manufacturer's reference manual, but it became apparent to publishers that they could enhance sales by tying their books to a particular implementation. When structured programming became fashionable about 1970, there was an enormous turn-over in programming language texts, and the problem was compounded when several different styles of structured programming were developed. The change of the 1980's is to modify these texts again to emphasize software engineering techniques and to precede the introduction to programming languages by a short course on problem solving. Similar changes are taking place in most of the subject areas which compose the core courses in computer science, to the extent that curricula are changing on an almost annual basis and students of a program are already two to three years behind in their knowledge of a particular topic by the time they graduate! Thus as a guideline for curriculum, textbooks are tending to confuse the issue and cause irregularities in the prerequisite structure of the undergraduate program (and graduate program to some lesser degree).

Faculty Experience

Contrasted with some other science or engineering disciplines, computer science faculty are seriously deficient in industrial experience. This has the effect of not providing to the academic environment any personal injection of industry needs based on prior experience. Some faculty have been able to maintain a small degree of affinity through summer activities or consulting arrangements, but these short term relationships tend to be more closely related to research topics or advanced technology than to the topics of an undergraduate degree program. Almost none of the faculty in university computer science departments have ever been employed in the position for which they believe they are preparing their own students!

INDUSTRY NEEDS

While it is legitimate to regard computer science as simply another subject for study which will meet the needs of a general educational program to develop the "educated man", the demands of the industry and nation must place some priorities on the use of limited facilities and even more limited faculty time. Already some institutions have begun to set limits on the size of their freshman classes so as not to overtax the available resources, while others, required by state law to accept all qualified candidates, cannot guarantee that a student will be able to get access to the courses he requires to graduate within the normal period of residence in a university. Decisions must be made as to which criteria for graduation will be imposed on this highly selective cadre of candidates -- academic or industrial requirements? The academic mission can generally be met by imposing liberal education requirements that balance out other inequities in a specialized program of study; thus an emphasis here on industrial needs is not intended to imply that academic, educational needs are to be ignored. Instead there is a responsibility of the academic community to mute any sharp edges of era-dependence on specific industrial needs and to match these needs with supporting educational (and recreational) programs.

Short Term Objectives

The computer industry is no different from any other with respect to regarding what it needs in the matter of knowledge and skills of the new graduate employee; an ability to immediately be integrated into the workplace and to be productive (and profitable) as soon as possible. Thus employers tend to demand knowledge of specific computer systems and particular software packages. When one considers that the half-life of most computer hardware systems and software packages is only five years, then it is clear that by the time a university integrates studies of these elements into their curriculum that their usage is already on the decline! However, the relationships between potential supervisors of a graduate and the personnel department representative who visits a university or college at placement time is such that if specific named skills or knowledge are missing from a student's transcript then he is not considered further for the position and the educational institution is given a bad name for not teaching the right subjects!

Long Term Objectives

It is to the long term objectives of employment that an educational institution can best respond. Short term objectives are satisfactory guidelines for summer employment or perhaps cooperative educational programs, but the long term goals can be used to mold curriculum and give freedom of choice that is essential to maintain academic flexibility for general (but with a computer science emphasis) education. Whereas in Europe and the Soviet block countries the university student has already obtained his general education in the high school so that his college studies can concentrate almost totally in the area of speciality*, the average U.S. student spends only 16-20 hours per week in a classroom and only 25% of his required studies are in the area of his major. It quite feasible for a non-U.S.A. student to take the same courses as his U.S.A. counterpart and to then supplement these with courses which pertain directly to industrial applications of his theoretical studies. To some extent this could be satisfied by the proposals set forth in the Feldman Report [1] for an experimental science.

We believe that the long term needs of industry, apart from specific needs for skills and knowledge in the core areas of the discipline include:

----

* Most European and Soviet block countries students do not study any other subjects in college other than those necessary for that area. All recreational subjects are extra-curricula.

Will develop and maintain technical awareness

Will be innovative and have profit motives

Will be ready to move up the management ladder

THE SHORTFALLS OF ACADEMIA AND INDUSTRY

While there have been significant advances in computer science in the past years, the shortfalls and missed objectives seem to outnumber the successes. The number of graduates from Computer Science Programs does not match the needs of the industry, the balance has been made up from students whose original major was not computer science but who have obtained some exposure to computer science through service courses or courses taught outside of computer science.

THE EMPLOYMENT CRISIS

The needs of the industry can be classified into three major areas by job description or industry type: (i) those tasks which are concerned with the incidental use of the computer in some other industry such as banking or engineering, (ii) tasks in a computer support group such as a computing center or service center which require the development of applications packages for use in the administrative and operational elements of a non-computer industry and (iii) tasks which are central to the production of hardware and software for delivery to customers by a computer vendor.

THE SHORTCOMINGS OF THE ACADEMIC ENVIRONMENT

The academic environment is at the best a model of a real environment and is today under great pressures to reduce costs while maintaining high numbers of students enrollments with maximal standards of course content. The universities and colleges are places to learn about everything and it is up to "them" to provide the right facilities. Anyone who has contact with the parents of students can realize the naivety of their concepts of college funding. In reality such funding is not available and universities and colleges manage to do some marvellous feats of magic with limited funds. Most industries would have given up long ago if faced with the restrictions of a academic budget.

Lack of Qualified Faculty

With the demand for highly qualified faculty and the low output of the colleges and universities, combined with the lack of competition between academic salaries and those of a highly favored industry, it is not surprising that most institutions have openings in their faculty ranks. The requirements for tenure, promotion and salary increases in the academic setting require that a person be expert in several fields, teaching being merely one of the criteria for advancement. Thus persons who have been highly trained in a advanced degree program to conduct meaningful research are generally ill-prepared and not enthusiastic about teaching and community service. Most Ph.D.s entering universities and colleges are expected to be able to teach without further training and without experience in lesson planning and presentation. And yet it is these very points that student evaluations are based; such evaluations can form a significant portion of the criteria which are reviewed for advancement. In contrast, the industrial employee is judged in much more limited areas of competence and where shortcomings are evident facilities are made available to make up the deficiencies.

<u>Lack of Up-to-Date Equipment</u>

As reported in [11], the cost of computer equipment is such that most institutions plan to amortize their investment over a period which is in fact greater than the effective half-life of that equipment! While the cost benefits of leasing have been shown to be negative from a financial point of view, the "locking-in" of equipment by purchase is a technological factor which most university administrations cannot fathom. Where Computer education is intended merely to prepare the student for employment in "standard", middle-of-the-field applications, antiquated equipment can be tolerated for some time. However in four-year computer science programs which are intended to produce the employees who will move the industry forward, or who will at the very least be capable of maintaining their own technical capabilities, starting them off with equipment which is 5-10 years behind is unacceptable.

This need for flexibility may be one argument to propose that institutions be served by a network of small systems which can be replaced easily and upgraded gradually as the equipment is obsoleted. On the other hand, the development of such networks is still itself in the stage of infancy, unless one is prepared to commit to a single vendor and the restrictions that such a choice will entail.

Space

The space requirements for Computer Science are not greatly different from those of other experimental disciplines; however, Computer Science is the younger "have-not" discipline vying with the established departments for more space. Space, in general is a problem for all institutions and Computer Science is merely caught up in the problem -- from a position of weakness.

Experimental Methods

While it is not common for computer science to be taught as a computer-aided instructional course, the computer itself is a responsive device which under most circumstances will not simply "die" or cause a disaster when improper data is supplied. This quality makes the computer a near-perfect tool for use in an experimental environment, the responses given by the computer being much more reliable than provided by a teacher who has to keep the knowledge of several different systems in mind. Moreover, industry usage of computers includes the use of experimental techniques for the verification of the correctness of products, based on the same techniques as are used in other experimental sciences such as physics and chemistry. While the encouragement of experimentation as a teaching tool would take some of the load off the already overloaded teacher, it does require the use of more computer facilities and power. Thus the experimental approach to undergraduate education in computer science is discouraged. In some cases such freedom is limited not only by resources but also by the fear of overindulgence on the part of students. Computer science laboratories currently spend their time in tasks of programming; there is not reason why those same periods could not be used for experiential learning.

At the research level, computer science in the USA has always lead the world in experimental techniques mainly as a result of the availablity of equipment. Other countries showed much more interest (and some advantages) in theoretical computer science but could not apply these techniques to practice. In many instances US industry was able to take advantage of both national experimental expertise and international theoretical research. This advantage is being lost as research in US universities is not well funded as an experimental or implementation science. Theories about computing abound but their transfer to technology rarely occurs outside of the industrial laboratory. A return to a balanced program between theory and experiment (or implementation) is sorely needed and was recommended in the Feldman report [1].

Industrial Experience and Liaison

Since the majority of computer science faculty have been recruited directly from their degree program, and the differential in salaries discourages industry scientists to return to the universities, there is very little industrial experience amongst that faculty. Perhaps the most common experience is as a programmer or aid in a university computing center during graduate study, though there may be an equal number of faculty who through consulting arrangements have acquired industrial experience. As a part of some industry-college arrangements, industry and colleges can exchange staff for periods, arrangements which can be beneficial to both organizations. Unfortunately, the majority of faculty in this situation want to transfer to the research group within the industrial organization, thereby permitting them little contact with the "real" world. Students involved in education cooperative programs often know more about industrial needs and activities than do their own faculty!

Educational Tools

The promise of computer aided/assisted education has not been fulfilled since th tools for such endeavours have not been provided by computer vendors except in a few very expensive environments. Moreover the time to develop courses has not been made available to faculty. The demand for teachers in existing, traditional courses has not permitted administrations to be able to provide the necessary release time for these projects and the preparation time for a such courses is a factor of magnitude greater than that for a single course offering. Add to this the need to provide terminal access for students involved in computer assisted instruction and it becomes apparent that the environment is not yet ready for such projects.

On the other hand the administrative aspects of course management could easily and readily be implemented by university administrations. These facilities would include class enrollment records, grade recording and computation, assignment submission and grading. For many years universities and colleges have developed programming systems (generally based on established languages) which have assisted students in programming skills and have often contained grading schemes. However, like computer aided instructional systems these have not been picked up by vendors so as to provide the necessary maintenance and continued support. Moreover the majority of university computing center staffs are already overburdened with the maintenance of vendor systems to be able to develop and maintain local developments. There are exceptions to this such as the University of Waterloo, University of California at San Diego and Cornell University who have supported their own environments by marketing their products.

<u>Need to provide Services to other programs</u>

The newness of computer services has made the faculty of computer science departments to be the local experts in the field. As a result they are called upon to provide (free) consultative support to many other elements of the campus environment and even to local governments and businesses. While this imposes considerable strain on already overloaded resources, the need to maintain professional standards and to instill "the right direction" requires involvement. As a result computer science faculty find that their services are in demand and that to refuse can only lead to resentment or, even worse, attempts to develop unstructured programs based on weak foundations. This is particularly prevalent in universities and colleges today where the inablity of computer science faculty to be involved in course development for other programs had lead to the development of courses which are at least one generation out of step with the current state of computing. For example, computer courses in engineering which are still using FORTRAN IV, or beginning education courses which perpetrate BASIC on unsuspecting teachers to be and hence the next generation of students!

Finances

Financial support of computer science programs is generally based on past experiences with programs in other disciplines and is started on a shoe-string by piggy-backing on other facilities such as the computing center. In the past few years the advent of micro-computers or personal computers has lead adminstrations to believe that computer science programs can be established using experienced users in other disciplines and with simpler facilities. This is practical until courses past simple programming are required. At that point the inexpensive approach must be replaced by a adequately supported program. Unfortunately the simple beginnings are not easy to replace in university budgets which cannot be expanded to meet the demand and which cannot be "adjusted" by redistributing funds. Back pocket operations tend to remain as second class programs until someone agrees to place the correct priorities on the necessary line item in the general budget.

In the Commonwealth of Virginia computer science programs are currently teaching approximately 6% of the student credit hours and yet are receiving only 2 to 3% of the college budget. While this must be offset by funding for the computational facilities and by the normal university overhead for adminstrative functions, it still does not represent a level of funding which is needed to support a program which is equipment costly and which needs faculty salaries at a high enough level to compete at (say) the 75% level with industry.

POTENTIAL SOLUTIONS

The obvious solutions of the infusion of large amounts of money are not likely to come about and in any case will need adminstrative and professional leadership to be most effective.

INDUSTRY INVOLVEMENT - ADVISORY GROUPS

The community college system has generally used advisory committees to provide a liaison between the academic world and the local industrial needs. While this relationship must be viewed with some trepidation to ensure that the long range goals of the students are met before the short range goals of local employers, such liaisons can provide the experiential factors generally missing amongst the faculty. Contractual arrangements to support graduate students through research and development activities or cooperative education can be beneficial to both parties.

COOPERATIVE PROGRAMS FOR STUDENTS AND FACULTY

The concept of cooperative education for undergraduate students provides industry with some of the best qualified and most prepared graduates that seek employment. While the student spends an additional year to complete the degree program, the financial assistance provided to the student may be th difference between attending college and not for many students. At the same time, the cooperative employer gains an insight into the student's capabilities and is able to train them in their own procedures at a lower cost than for the new graduate; further the chances for employing that graduate are high.

Extending the cooperative program concept to Faculty could be equally profitable. Through an exchange program, faculty can provide an infusion of current technology into the industry in which they participate and at the same time can gain valuable practical experience to distribute to the classroom. These Faculty programs may be restricted to summer experiences or, with an true exchange of personnel, may be alternating six month periods.

CONTRACT RESEARCH AND DEVELOPMENT

Basic research in computer science is being well supported by institutions which became involved in the field twenty years ago. Institutions such as MIT, Stanford and Carnegie Mellon have the equipment and faculty to carry out basic research and can "roll-over" research grants into continuing projects. The majority of lesser institutions will probably not be able to become involved in basic research but can be supported by industrial contracts and grants to engage in technology transfer. Applications research based on the established work of other institutions can benefit both the grantor and the department involved.

PROFESSIONAL DEVELOPMENT PROGRAMS FOR FACULTY

While it is true that computer science departments are beginning to employ more and more faculty trained in computer science or related fields, the supply is not going to meet the demand for several years -- perhaps a decade. Thus the majority of computer science faculty will be (for the next 10 to 15 years) persons who got into the field before there was a computer science field per se, or recent graduates of other disciplines who had enough computer science courses or experience to qualify them to teach basic courses and to do research in the interface between the two disciplines. Both of these groups of faculty need to participate in professional development programs to enhance their abilities and to maintain their currency. It is possible that some of this same development can be achieved by personal study and research. However, where time is of the essence, a less expensive and quicker excursion into an area can be achieved by organized professional development courses such as presented by the IEEE or ACM.

While there is strong industry demand for computer-aware graduates from the educational institutions, the ability to fulfill those needs is being stunted by both the inability of the schools to attract and keep qualified faculty and by the pressures from government to reduce the costs of education. On the other hand, this author is not convinced that the demands of the computer industry cannot be met by the current facilities; the drain on these programs comes from those application industries who will find the Computer Science graduate over qualified. It may well be the fact that 60% of the graduates of computer science programs eventually find employment in positions which do not require the intensity of knowledge and skills provided by a computer science program. Many solutions exist to this problem; with better counselling in high schools, the number of students who need a program which is centered on Computer Science or Information Systems could well be reduced to a manageable size. The colleges should then expand on their offerings which would support minors in Computer or Information Science and consider the provision of a post-graduate certificate program for graduates whose undergraduate program is an applications area. Ancillary to this, a rigorous freshman course in computer literacy may suffice to fulfill the needs of many disciplines; with a concerted effort to provide materials and supporting personnel, literacy courses can be staffed with computer users in applications areas such as education, agriculture or engineering.

At the same time, colleges must recognize that the demand for graduates is somewhat inflated by a lack of understanding on the part of prospective employers. There still exists in the minds of some personnel representatives that there exists a priesthood of programmers which can only be provided by Computer Science departments. Employers must be educated to understand that their needs may be better met by graduates of (say) a school of business with a computer or information science/systems minor than graduates who are over qualified in computing and under qualified in the necessary applications area.

In conclusion, the problems of computer-related education in universities and colleges can be solved without massive infusion of funds (though some additional support is obviously necessary). Solutions will range from correctly identifying the clientel for Computer Science and Information Science/Systems programs to relieving the burden of service courses by using faculty who are knowledgeable applications users committed to the tenets of a science which is codifying its techniques very rapidly.

REFERENCES

[1] Feldman, J.A., and Sutherland, W.R., "Rejuvenating Experimental Computer Science", Comm. ACM, V22, N9, Sept. 1979, pp.497-502.

[2] McCracken, D.D., Denning, P.J., and Brandin, D.H., "An ACM Executive Committee Position on the Crisis in Experimental Computer Science", Comm. ACM, V22, N9, Sept. 1979, pp.503-504.

[3] Denning, P.J., "A Discipline in Crisis", Comm. ACM, V24, N6, June 1981, pp.370-374.

[4] Druffel, L.E., "Strategy for a DoD Software Initiative", U.S. Dept. of Defense, OUSD(R&E), 1982 Oct. 1, 66+45 pp.

[5] --, ACM Recommended Curricula for Computer Science and Information Processing Programs in Colleges and Universities, 1968-1981, ACM, New York, 1981, 181 pp.

[6] Rosser, J.B., (Chairman), "Digital Computer Needs in Universities and Colleges", Pub. No. 1233, National Academy of Sciences, Washington DC, 1965.

[7] Pierce, J.R., (Chairman), "Computers in Higher Education", Report of the President's Science Advisory Committee, The White House, Washington DC, Feb. 1967, 79 pp.

[8a] Adams, D.R., and Athey, T.H., (Eds), DPMA Model Curriculum for Undergradate Computer Information Systems Education, DPMA Education Foundation, Park Ridge IL, 1981, 67 pp.

[8b] Education Committee (Model Curriculum Subcommittee), IEEE/CS, A Curriculum in Computer Science and Engineering, Committee Report, IEEE Pub. No. EHO119-8, Piscataway NJ, 1977.

[9] Taulbee, O., and Conte, S., "Production and Employment of Ph.D.s in Computer Science", Comm. ACM, V22, N2, Feb. 1979, pp.75 ff.

[10] Cook, J.R., Gallagher, M.C., and Johnston, M.A., An Analytical Study of Industry's Computer Education Needs, Interface, Vol.1, No.1, pp.52-57.

[11] The Wall Street Journal, Vol.CCI, No.10, Eastern Edition, 1983 January 15.