COPY

May 29, 1969

Dr. John R. Pierce
Executive Director
Research Communications Sciences Division
Bell Laboratories
Murray Hill, New Jersey 07974

Dear John:

I was interested to see your enthusiasm for the use of simple computer programs in education during the last Computer Science and Engineering Board meeting. We have been working with a number of schools in developing programs that would make computers interesting and educational to students in high schools and liberal arts colleges.

Enclosed is a copy of our handbook for programming our small computers. We hear stories of how first-, second-, third-, and fifth-grade children have learned to program FOCAL from Chapter 9 with no help from adults. These stories might be somewhat exaggerated, but it shows that students can learn if left alone with access to a computer.

Sincerely yours,

Kenneth H. Olsen

KHO:ecc

cc: Mr. Warren C. House
Executive Secretary
Computer Science and Engineering Board
National Academy of Sciences
2101 Constitution Avenue
Washington, D. C. 20418

Sent:

"Introduction to Programming"

Computalk (Newsletter)

DEC Newsletter Vol I No 6

" " Vol I No. 11

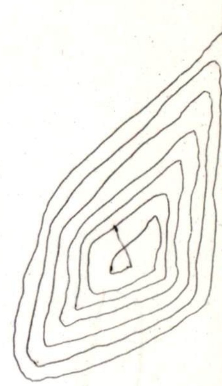
Project Local

Computers in the Classroom (new issue)

DECUS Proceedings - Spring, 1968

Carnegie-Mellon University

Department of Computer Science
Schenley Park
Pittsburgh, Pennsylvania 15213
[412] 621-2600
[412] 683-7000



February 4, 1969

LETTER OF INVITATION

Dear

A conference to study computer science education in the United States will be held July 21 through 25, 1969 at Woods Hole, Massachusetts. The conference is being sponsored by the National Science Foundation Computer Science and Engineering Research Board under a grant from the National Science Foundation.

The purpose of this letter is to invite you, as one of approximately 40, to participate in the work of this conference.

The conference will be organized to make maximum use of the participant's capabilities in the time available. It is planned to hold all day meetings during the entire week and to focus on two specific topics:

1. Graduate education in computer science
2. Education in software (and hardware) systems.

The conference discussions and conclusions may broaden considerably beyond these two areas; nevertheless they seem reasonable for initiating and focusing discussion. With each of these issues there will be two major technical concerns:

A) Economic: By economic is meant the creation of input-output models relating the development of programs, production of students and faculty, and the needs of industry and government for people so trained. Furthermore, a timetable establishing the velocity and acceleration of these programs should be produced. In accord with the postulated growth, a study should be made of the resources (plant, people, and money) required to provide this educational development.

B) Content: A thorough study should be made of the content of the undergraduate and graduate programs to be labeled as computer science. Furthermore, an audit of existing programs should be made to gauge what distances exist between what is being done and what should be done. Furthermore the subject of accreditation and standardization should be treated. Similar treatment should be accorded to education in software (and hardware) systems.

It is planned to organize the meeting as a sequence of open plenary sessions with the entire group meeting to discuss the partial results obtained in one of the above areas; and in working sessions divided into working technical groups. A tentative schedule for the two major work groups (Content -- Working Group I and Economics -- Working Group II) follows:

	9:00 - 12:00 a.m. (morning)	1:30 - 4:30 p.m. (afternoon)	7:00 - 10:00 p.m. (evening)
Monday	Keynote Plenary Session I	Working Sessions	Special Lectures
Tuesday	Working Sessions	Plenary Session II Report of Working Group I	Special Lectures
Wednesday	Plenary Session III Report of Working Group II	Working Sessions	Working Sessions
Thursday	Plenary Session IV Report of Working Group I	Plenary Session V Report of Working Group II	
Friday	Draft of reports of working groups	Plenary Session VI Draft of final report - content and conclusions	

There are a largen number of questions that the conference should attempt to answer. Among them are:

- Of the reasonably large number of graduate departments of computer science now existing, are these programs producing in kind and in number the graduates that are needed?
- Are there needs, insofar as computer science is concerned, which these programs are not meeting?
- Are these programs separating the mathematical from the engineering too much?
- What alternatives to this mode of educational development can be proposed?
- Does there exist a natural education sequence in the field of computer science like that, e.g., in another mathematical science? Thus, how does one characterize education in computer science through the range of junior college, B.S., B.A., M.S., M.A., Ph.D., and professional degree?
- In the field of computer science what are the goals of the various degrees?
- Is the education program best organized so that students from the lower degree programs provide the major source of the students in the advanced degree programs?
- Will computer science departments become as introverted as has happened, for example, in mathematics?

- How do the programs now in operation compare with those outlined by study groups such as the ACM Curriculum Committee and COSINE?
- Are the professional societies the approprite groups to recommend or set curricula? What orderly alternatives are there?
- Are there large problems in software production and use that are largely caused by the lack of well trained software specialists?
- If there are such large problems, should they be solved within a formal education system by educating specialists at various degree levels?
- Or can this matter be best solved by those now responsible for the production of software using on-the-job training?
- Thus, can hardware manufacturers be depended upon to supply the software systems that are needed and also train the personnel produce and service them?
- Would not software education in a university environment produce technological derelicts since the software problem seems to change so rapidly?
- Put another way, won't the very nature of software make the solutions to these problems be solved by meta software produced by a very small number of specialists?
- If one speaks of software engineering, then why not let the engineering schools and disciplines define and develop the programs?
- Is it possible to meaningfully separate the software problem from the hardware problem?
- How can national institutes of computer science, several of which are now being proposed, contribute to education in computer science?

Other questions will arise during the course of the discussions, but certainly the goal of the conference should be to focus not only on the nature of the problem but to prepare recommended solutions.

Though it is not requred for participation, the attendees would be pleased to receive from you any written comments that you might care to make prior to the meeting. While formal papers are not being asked for, careful organization of your thoughts on these or other related matters would be appreciated. If a working paper can be provided by June 15th copies will be made available to all the participants to study before the meeting commences. These working papers will undoubtedly provide a strong basis for discussion during the conference.

During the conference, duplication and secretarial facilities will be provided for quick preparation of additional working papers and intermediate reports. The goal of the conference will be the preparation of a report outlining the results of the conference. Toward that end, in each of the two areas (resources and content), a chairman and two younger recording secretaries will have the responsibility of preparing the draft of each section, and these two reports will then be coordinated into a final report.

You may be familiar with a report of the National Academy of Science entitled "The Mathematical Sciences: A Report (NAS publication 1681:1968, xiv + 256 pages, paper, $6.00). This report, and preceding reports by the Peirce Committee and the Rosser Committee are the sole widely based surveys conducted under federal auspices on computer science education. It is hoped that the report of this conference will provide a major technical expansion of the requirements and goals of computer science education.

Please let me know as soon as possible, and in no case later than March 15, if you will participate in this conference.

Very truly yours,

Dr. Alan J. Perlis, Head
Department of Computer Science

AJP:dg

The current list of invitees is:

1) The Economic Group:

B. Gilchrist	Secretary of AFIPS
G. Forsythe	Chairman, Department of Computer Science, Stanford
J. E. Rowe	Computer Operations, Union Carbide Corp.
J. W. Carr III	Chairman, Department of Computer Science University of Pennsylvania
A. J. Perlis	Head, Department of Computer Science, CMU Chairman of this working group
T. A. Standish	Assistant Professor, CMU, Department of Computer Science -- Recording secretary of this working group
A. VanDam	Associate Professor, Computer Science, Brown University -- Recording secretary of this working group
L. Zadeh	Professor of Computer Science and Electrical Engineering, University of California (Berkeley)
J. Snyder	Professor of Computer Science, Physics and Computer Center, University of Illinois
W. Humphrey	IBM, Manager, Software Systems
F. Brooks	Chairman, Department of Computer Science, University of North Carolina
John Hamblen	SRE Board
Richard Jones	Manager, Applied Data Research (Private software house)
Tom Jones	President, University of South Carolina (university administrator and electrical engineer)
R. Tanaka	Calcomp, Electrical Engineer and Systems Designer
Andrew Schultz	Dean of Engineering College, Cornell University
Charles Bowen	IBM
William F. Sharpe	UCLA and RAND
Chuan Chu	Vice President and Assistant General Manager, Honeywell Data Processing, Wellesley, Mass.
Ross McDonald	Vice President and Director of Research, Texas Instruments, Dallas, Texas.
Simon Ramo	Bunker-Ramo Corporation
Isaac Nehama	International Computing Company

Walter Ramshaw	Hartford hospitals
Burton Colvin	Boeing Scientific Research Laboratory
George A. Garrett	Director, Information Processing, Lockheed Missiles and Space Company, Sunnyvale, Calif.

2) The Content, Audit and Accreditation Group:

E. McCluskey	Professor of Electrical Engineering Stanford University, Group Chairman
J. Gries	Assistant Professor, Computer Science Stanford University, Recording Secretary
F. Gruenberger	Educator, SanFernando State College Recording Secretary
R. Spinrad	Software Manager, Scientific Data Systems
J. Hartmanis	Chairman, Department of Computer Science Cornell University
S. Conte	Chairman, Department of Computer Science Purdue University
R. Hamming	Computer Science, Bell Telephone Laboratories
F. Corbato	Project MAC, Massachusetts Institute of Tech.
J. Schwartz	SDC
W. Bauer	President, Informatics
R. Andree	Computer Educator, Professor of Mathematics University of Oklahoma
J. Harr	Director of Software, AT&T, Central Office Computer Systems
A. R. Zipf	Computer Operations, Bank of America
V. Vyssotsky	Software Management, Bell Telephone Laboratories
D. Knuth	Professor of Computer Science, Stanford Univ.
T. Climis	Manager of Software, IBM
C. G. Bell	Computer Systems Designer, Professor, Carnegie-Mellon University
R. Graham	Director of Computing Operations and Software Production, University of Waterloo
Charles Missler	Manager, Computer Application/Engineering Staff, Ford Motor Company, Dearborn, Michigan.
Jerry Noe	University of Washington, Seattle
Roger Lazarus	Head, Computer Division, Los Alamos Scientific Laboratory, Los Alamos, N.M.

Charles DiCarlo	President Sarah Lawrence College
Earl Althoff	Joint Computer Evaluation Group, Eastman Kodak, Bldg. 205, Kodak Park, Rochester, N. Y.
Ruth Davis	National Library of Medicine
Howard Campaigne	National Security Agency
Ned Irons	Institute for Defense Analyses Princeton
Paul Brock	NASA, Houston, Texas

(DRAFT)

REPORT OF THE CONFERENCE ON COMPUTER SCIENCE EDUCATION

sponsored
by
THE NATIONAL ACADEMY OF SCIENCE
COMPUTER SCIENCE AND ENGINEERING RESEARCH BOARD

held
in

HILTON MOTOR HOTEL
ANNAPOLIS, MARYLAND

JULY 21-25, 1969

TABLE OF CONTENTS

INTRODUCTION

A conference to study computer science education in the United States was held July 21 through 25, 1969 at the Hilton Hotel in Annapolis, Maryland. The conference was sponsored by the National Academy of Science Computer Science and Engineering Board under a grant from the National Science Foundation.

The Computer Science and Engineering Board has been formed to provide a focus for those aspects of the computer field that are important to science in general and the federal government. Attached is a document that describes the purposes of the Board.

The conference was organized to make maximum use of the participant's capabilities in the time available. It is planned to hold all day meetings during the entire week and to focus our attention on two specific topics:

1. Graduate education in computer science
2. Education in software (and hardware) systems

The conference discussions and conclusions may broaden considerably beyond these two areas; nevertheless they seem reasonable for initiating and focusing discussion. With each of these issures there will be two major technical concerns:

B) Content: A thorough study should be made of the content of the undergraduate and graduate programs to be labeled as computer science. Furthermore, an audit of existing programs should be made to gauge what distances exist between what is being done and what should be done. Furthermore the subject of content and standardization should be treated. Similar treatment should be accorded to education in software (and hardware) systems.

It is planned to organize the meeting as a sequence of open plenary sessions with the entire group meeting to discuss the partial results obtained in one of the above areas; and in working sessions divided into working technical groups. A tentative schedule for the two major work groups (Content -- Working Group A and Resources -- Working Group B) follows:

Morning		Afternoon
	Monday	
Introduction		Work
	Tuesday	
Work		Work
	Wednesday	
Report A → B		Report B → A
	Thursday	
Work		Draft
	Friday	
Final Reading		

There are a large number of questions that the conference should attempt to answer. Amont them are:

- Of the reasonably large number of graduate departments of computer science now existing, are these programs producing in kind and in number the graduates that are needed?
- Are there needs, insofar as computer science is concerned, which these programs are not meeting?
- Are these programs separating the mathematical from the engineering too much?
- What alternatives to this mode of educational development can be proposed?
- Does there exist a natural education sequence in the field of computer science like that, e.g., in another mathematical science? Thus, how does one characterize education in computer science through the range of junior college, B.S., B.A., M.S., M.A., Ph.D., and professional degree?

- Does there exist a natural education sequence in the field of computer science like that, e.g., in another mathematical science? Thus, how does one characterize education in computer science through the range of junior college, B.S., B.A., M.S., M.A., Ph.D., and professional degree?
- In the field of computer science what are the goals of the various degrees?
- Is the education program best organized so that students from the lower degree programs provide the major source of the students in the advanced degree programs?
- Will computer science departments become as introverted as has happened, for example, in mathematics?
- How do the programs now in operation compare with those outlined by the study groups such as the ACM Curriculum Committee and COSINE?
- Are the professional societies the appropriate groups to recommend or set curricula? What orderly alternatives are there?
- Are there large problems in software production and use that are largely caused by the lack of well trained software specialists?
- If there are such large problems, should they be solved within a formal education system by educating specialists at various degree levels?
- Or can this matter be best solved by those now responsible for the production of software using on-the-job training?
- Thus, can hardware manufacturers be depended upon to supply the software systems that are needed and also train the personnel to produce and service them?
- Would not software education in a university environment produce technological derelicts since the software problem seems to change so rapidly?
- Put another way, won't the very nature of software make the solutions to these problems be solved by meta software produced by a very small number of specialists?
- If one speaks of software engineering, then why not let the engineering schools and disciplines define and develop the programs?
- Is it possible to meaningfully separate the software problem from the hardware problem?
- How can national institutes of computer science, several of which are now being proposed, contribute to education in computer science?

Other questions will arise during the course of the discussions, but certainly the goal of the conference should be to focus not only on the nature of the problem but to prepare recommended solutions. Naturally, any additional questions that you feel should be discussed will be considered. We would appreciate any feeling you may have concerning the priorities of the various topics which have been raised.

Though it is not required for participation, the attendees would be pleased to receive from you any written comments that you might care to make prior to the meeting. While formal papers are not being asked for, careful organization of your thoughts on these or other related matters would be appreciated. If a working paper can be provided by June 22nd copies will be made available to all the participants to study before the meeting commences. These working papers will undoubtedly provide a strong basis for discussion during the conference.

It is hoped that this conference will provide a reference for the field of computer science -- at least in the two major areas -- that will be a natural first source for information about the field. The conference will be attempting to obtain in one week what the more established sciences have developed over many years -- an overview of the present state, logistics, and future directions of the field. Naturally it could not hope to be complete, but it will provide a first overview of the field that up to now has not existed.

During the conference, duplication and secretarial facilities will be provided for quick preparation of additional working papers and intermediate reports. The goal of the conference will be the preparation of a report outlining the results of the conference. Toward that end, in each of the two areas (resources and content), a chairman and two younger recording secretaries will have the responsibility of preparing the draft of each section, and these two reports will then be coordinated into a final report.

You may be familiar with a report of the National Academy of Science entitled "The Mathematical Sciences: A Report (NAS publication 1681:1968, xiv + 256 pages, paper, $6.00). This report, and preceding reports by the Pierce Committee and the Rosser Committee are the sole widely based surveys conducted under federal auspices on computer science education. It is hoped that the report of this conference will provide a major technical expansion of the requiements and goals of computer science education.

Please let me know as soon as possible, and in no case later than June 9th, if you will participate in this conference. Upon receipt of your willingness to participate in the conference you will be receiving a set of preliminary documents on or about June 15th. These documents will include the full list of attendees, copies of the above mentioned report of the National Academy and the Pierce Committee, a report of the ACM Curriculum Committee, and working papers as they become available. A partial list of attendees and the groups to which we have tentatively assigned them is attached. I would appreciate additional names of people whose presence would materially improve the conference.

Sincerely yours,

Alan J. Perlis (dg)

Dr. Alan J. Perlis, Head
Department of Computer Science
Carnegie-Mellon University

AJP:dg
enc.

LIST OF ATTENDEES

Prof. Richard Andree
Dept. of Mathematics
University of Oklahoma
Norman, Oklahoma 73069

Dr. Bruce W. Arden
Associate Director
Computing Center
University of Michigan
Ann Arbor, Michigan

Dr. C. L. Coates
Electronics Research Center
University of Texas at Austin
Austin, Texas 78712

B. H. Colvin
Head, Mathematics Research Laboratory
Boeing Scientific Research Laboratories
P.O. Box 3981
Seattle, Washington 98124

Dr. Ruth Davis
National Institutes for Health
National Library of Medicine
Bethesda, Md.

Dr. George and Alexandra Forsythe
Computer Science Department
Stanford University
Stanford, California 94305

Dr. John Giese
Chief, Applied Mathematics Division
Department of the Army
U.S. Army Ballistic Research Laboratories
Aberdeen Proving Ground, Maryland 21005

Mr. Bruce Gilchrist
Executive Director
American Federation of Information Processing Societies
210 Summit Ave.
Montvale, N.J. 07645

Prof. J. W. Graham
Computing Centre Director
University of Waterloo
Waterloo, Ontario, Canada

Prof. Fred Gruenberger
Department of Accounting
San Fernando Valley State College
Nothridge, California 91324

Dr. John Hamblen
Southern Regional Education Board
130 6th Street N.W.
Atlanta, Georgia

Dr. Walter W. Jacobs
1812 Metzerott Road
Adelphi, Maryland 20783

Mr. Scott E. Moore
Manager of SDD Technical Education
IBM Systems Development Division
Department H77, Building 962
Box 390
Poughkeepsie, New York 12602

Saul Rosen, Director
Computer Sciences Center
Mathematical Sciences Building
Purdue University
Lafayette, Indiana 47907

Dr. Samuel Seely
Associate Graduate Dean
University of Massachusetts
Amherst, Massachusetts 01002

Professor J. N. Synder
Associate Head of Computer Science
University of Illinois
Urbana, Illinois 61801

Dr. Robert Spinrad
Scientific Data Systems
701 South Aviation Boulevard
El Segundo, California 90245

Professor John W. Tukey
Department of Statistics
Fine Hall, P.O.Box 708
Princeton, New Jersey 08540

Dr. John Carr, III
Moore School of Engineering
Department of Computer Science
University of Pennsylvania
Philadelphia, Pennsylvania

Professor Juris Hartmanis
Department of Computer Science
Cornell University
Ithaca, New York

Professor E. J. McCluskey
Electronics Department
Stanford University
Stanford, California 94305

Mr. Robert Morris
Bell Telephone Laboratories, Inc.
Room 2C-524
Mountain Avenue
Murray Hill, New Jersey 07974

Mr. James Rowe
Union Carbide
270 Park Avenue (41st floor)
New York, New York

Dr. T. L. Jordan
University of California
Los Alamos Scientific Laboratory
P. O. Box 1663
Los Alamos, New Mexico 87544

RECOMMENDATIONS

1. We support the second recommendation of the COSRIMS report which we repeat here:

> "We recommend that at the national level special priority be given to support of the expansion of research and graduate study in computer science. Appropriate actions would include: special support for developing and updating courses, support for research during the academic year when needed, grants to departments to cover costs of computer usage in research, special attention to needs for space, and expansion of numbers of research assistantships and traineeships to stretch the capacity of all departments of high quality."

2. We recommend that universities, industry and the Federal Government cooperate in the development and support of excellent baccalaureate programs in computer science. While it is recognized that there may be a multiplicity of such programs at a university accenting different aspects of computer science, it is important that the development of the programs be entrusted to one faculty group that, if necessary, cuts across college boundaries.

Furthermore, we recommend that universities take steps to define master's degree programs in computer science that function to award a degree of consolidation built on the content of solid undergraduate programs in computer science and to deaccent master's programs whose major function is the conversion of baccalaureates from other fields to computer scientists.

Furthermore, we recommend that these baccalaureate programs contain strong elements of laboratory training in the development and utilization of computer systems.

The computer industry should be urged and encouraged to make major contributions to the development of computer science education in the universities.

In particular we deplore the recent trend toward the reduction and elimination of discounts to universities by computer manufacturers for the purchase of computing equipment.

We feel that the advantages to the whole computer industry far outweigh possible disadvantages to smaller computer manufacturers.

The computer industry has a strong vested interest in supporting the university programs that are their major source of supply of trained personnel. It is clearly in the interest of the whole industry to support university computer science programs.

3. Many of the existing and new Ph.D programs in computer science (in addition to that group of key institutions supported by large research grants oriented not specifically to educational problems) are drastically limited by the lack of support for competent graduate students.

At present, because of the restrictions of NDEA and NSF traineeships to already existing science and engineering disciplines, there are few fellowships available specifically to computer science graduate students.

It is recommended that new computer science graduate programs, in addition to those already supported by massive research grants, be supported in their initial and continuing stages by (1) graduate teaching and research fellowships, (2) post-doctoral teaching fellowships to aid in acquisition of new faculty, and (3) support of new and different computer facilities, such as satellite computers and processors for film and TV animation for instructional purposes, hybrid computers, converters to and from other systems, and new up-to-date equipment continuously being developed as a result of the investment of resources in national research and development through the defense, space, and scientific research programs.

4. It will be essential to the universities and colleges to greatly expand their students' opportunities to learn the essentials and principles of all elements in problem formulation to computing realization, and to be aware of the part that computer science wishes to play in offering such opportunities, and the cooperation of individual departments should be encouraged and supported, and departments with competent and interested staff should be encouraged and supported in providing opportunities for students to gain insight and knowledge in part or all of this area, and all reasonable efforts should be made to encourage interdepartmental cooperation in this whole area. And finally, that both research in the general area of application and materials preparation directed toward teaching deserves support, especially when each is planned to support the other.

5. In order to guarantee that the student body in this new undergraduate and graduate education in computer science be spread evenly geographically and economically across the United States, and in order to make sure that the result of this program is not the concentration of computer science

activity and talent in a small number of key prestige institutions, it is recommended that specific techniques be employed in the distribution of resources to guarantee grass-roots growth in this area throughout the United States.

To this purpose, it is recommended that undergraduate support be distributed on a pro rata student population basis throughout the states, similar to but not necessarily as in the National Defense Education Act, to the intent that students in all locales, including inner city and under-supported schools, can participate in this highly important program that will upgrade markedly the performance and productivity of many individual human beings.

6. Even in a relatively stable field like Mathematics, a strong need has been felt for up-to-date information about the nature of education and research in the field, and the amounts and sources of its funding. These needs resulted in the NSF-sponsored Survey of Research Potential and Training in the Mathematical Sciences (c. 1957), and the reports of the Ford Foundation-sponsored Survey Committee of the Conference Board of the Mathematical Sciences (c. 1967). The later committee is apparently to maintain a continuous inventory from now on.

In the rapidly changing field of computing sciences up-to-date information is needed even more, and is harder to get. Under NSF sponsorship, the Southern Regional Education Board has prepared surveys of college and university educational activity in the computing sciences, but apparently no agency is doing anything similar for research in our field. At the same time, graduate departments have a great need for, but possess very little information on what research in computing sciences is being sponsored; who does the research, who sponsors it, and at what levels.

We recommend that the NAS Computer Science and Engineering Board seek authorization, personnel, and funding for a continuing research survey committee, with some full-time staff, whose mission it would be to maintain a continuous inventory of research in the computing sciences.

7. It is recommended that the Computer Science and Engineering Board of the National Academy of Sciences make definite approaches to Congress to recommend that in the next budget legislation those funds authorized by the Higher Education Act for construction and the funding of computer equipment be made available to the National Science Foundation and the Office of Education so that a Federal program to support recommendations one through five can become operative on an appropriate scale.

National Academy of Science Conference
on Computer Science Education

SCHEDULE FOR THE WEEK OF 21 JULY 1969

Monday

A.M. General Discussion

P.M. Working Groups
Resources -- (Gilchrist)
Content -- (McCluskey)
What is Good Development-- (Arden)
Who will teach and who will do
research in systems development -- (Tukey)

Tuesday

A.M. Committee Reports

P.M. Working Groups
Resources (Perlis)
Content (Miller)
Goals and Tools (Morris)

Wednesday

A.M. Working groups
(same as Tuesday P.M.)

P.M. Committee Reports
Resources
Content
Goals and Tools

Evening Working groups
An Undergraduate Program in Computer Science

Thursday

A.M. Final Session

Drafts Prepared During the National Academy of Science Conference on Computer Science Education

1. Report of the Gilchrist Committee--What Kinds of Computer People are Needed? (Gilchrist)
2. What is Good Development? (Arden)
3. Problem Formulation and Matching--A Vital Segment (Tukey)
4. Retreading (Forsythe)
5. Hardware and Software Integration (Carr) (to be added in next draft)
6. A Potentially Large Manpower Requirement (Carr)
7. The Need for Increased Education in Software Engineering as a Subset of Computer Science (Carr)
8. Software Engineering (Rosen)
9. Purdue Masters Degree Program in Computer Science (Rosen)
10. Need for Links of Computer Sciences to the "Soft Sciences" and Life Sciences and for Continuing Monitoring of the Implications of the Effect of Computer Science on the Future of Civilization (Carr) (to be added in next draft)
11. Report of the Resources Committee (Perlis)
12. Education in Computer Science (Spinrad)
13. Computer Systems Laboratories (Miller)
14. A Graduate Curriculum in the Various Computing Sciences (Andree)
15. Report of the Goals Committee (Morris)
16. A Proposed Undergraduate Curriculum in Computer Science at Carnegie--Mellon University (Perlis)
17. Sub-College Education (A. Forsythe)

Report of the Gilchrist Committee

WHAT KINDS OF COMPUTER PEOPLE ARE NEEDED?

Exclusive of installations involving special purpose equipment, or equipment for specific special purposes (e.g., process control), as well as those involving very small machines, there are on the order of 25,000 installations in this country. Very roughly, they are organized by size and purpose like this:

		Scientific	Commercial
Large	1000:	800	200
Medium	10,000:	5000	5000
Small	14,000:	4000	10 000

The Committee agreed that for purposes of designing college-level training and education programs, the bottom group would have to g be disregarded. That group (typically users of 360/20's) has as great a need as the others for competent people, but unfortunately the proper person soon moves up. By default, that section of the computer world becomes staffed by poorly trained people.

For the first two groups, then, the people to be trained fall into these groupings:

Researchers

Systems Analysts

Systems Programmers (Large
(Small

Applications Programmers

User/Programmers

Users

The distinction between large and small in this outline was made by Gilchrist: those who qualify as "large" can modify OS360 to suit their firm's needs; those who are "small" can detect trouble in the operating system and know to whom to turn for help in fixing it. The group labelled "users" are those who also know whether or not the results are correct.

Scott Moore suggested a breakdown of people needs a different way, given in the following outline:

RESEARCH — Devises new tools and applications

Needs specialists in (hardware
(software
(combinations of the two

DEVELOPMENT — Develops those tools and how to use them.

Also needs experts in (hardware
(software
(combination of the two

APPLICATIONS — Requirements (or results) which have to do with data.

How to (select) a system.
(use)

How to measure effectiveness.

→ and all three groups deal with both theory and practical applications.

Computer science as a discipline is concerned with men involved with the theoretical design of tools and applications of them. As an industry (the committee agreed that it is not a profession), some 500,000 people are engaged more or less full time, but it has an abnormally high proportion of very incompetent people.

Attention was turned to a different view again: what sort of person does an employer look for and hire? We listed these specifications:

1) A certain gleam in the eye, vaguely defined as motivation.
2) Some knowledge of the mechanics of computing; e.g., the applicant has run some computer programs.
3) Problem solving adaptibility.
4) Communications skills (in both directions)
5) Ability to be self-critical
6) Elementary knowledge of statistics (this last is weak, or optional)

x————————x

The balance of the committee's time was spent listening to Prof. Graham's description of the training program at the University of Waterloo. The Committee recommends that he be asked to repeat this description for the entire group.

What is Good System Development?

Committee Report, Bruce Arden, Chairman

In the opening session many of the current problems of computing were related to poor development. Accordingly, the title question was posed. It is predictable in advance that no definitive solution to these problems will be produced but it bears consideration because of its possible educational implications.

First, the committee limited consideration to large systems, and rather arbitrarily designated this to be programs of over 200,000 words. Development can be divided into design and implementation. The design part can be further subdivided into specifications and partitioning. Throughout the discussion an attempt was made to identify these parts which are art as opposed to these where there can be some a priori system of types imposed. By art is meant bringing native ingenuity and experience to bear on the problems. A number of comments were made about the importance of the specification phase. The specifications are difficult in general purpose systems; in general, they have not been handled well. There are a couple of design tools. One of these is analysis. In its simplest form analysis is a very elementary arithmetic attempt to define the loads and processing times and make checks and predictions of system performance. Another specification tool is simulation, and here the aspect of art is apparent. In the extreme a simulation can be as complex as implementing the entire system; if it is too simple, it does not produce insight into the performance of the system which is being designed. Much depends upon the right choice.

Some time was spent on implementation structure. There were several independent estimates of how many people could be handled by a technical supervisor. The figures ranged from 5 to 10. If this estimate is accepted, then large systems impose a hierarchical structure. Actually there is more than one structure.

Actually there is more than one structure. There is the supervision structure, the system structure, and the documentation structure. It is not necessary that all of these structures be the same. However, for large systems it is certainly advantageous if the three structures are identical. Also, it is certainly clear from experience that if there is a single person that has a good overview of large systems, that this enhances the possibilities of success. When systems get very large this tree-like hierarchical structure may not resolve the difficulties. At every level as one progresses toward the root of the tree, there is some information loss and it is conceivable that, at the root, there cannot be any comprehensive knowledge of the system. During the discussion of very large systems, it was conjectured that the person at the top of this hierarchy could have only statistical knowledge of the details of the system.

In the implementation it is important to have early symbolic binding of the interfaces at the higher levels of the structure. This obviously permits the vital independence for large projects by permitting change within functional modules without perturbing others.

There was some discussion on "upward connectivity." That is, two supervisory groups, should communicate only through their common supervisor. This rule introduces order in the design structure, but it also can introduce, in an absolute sense, inefficiency in the resulting structure. This kind of inefficiency may have to be regarded as an unavoidable result of producing working, large systems.

As mentioned earlier, the documentation hierarchy referred to is very important, and should closely match the structural organization with increasing detail downward. Here again, there is an aspect of art. If the documentation level is well selected, it is relatively easy to enter at any level in such a tree-like structure and see the local structure. Defining appropriate documentation level well is very difficult. Early workbooks for big systems, illustrate both too much and too little detail. There were also opinions expressed that, although this hierarchical structure is vital, there is probably a need for more communication than can be obtained by staying on the branches of this tree. One suggestion was that a person be responsible for a certain band across this structure; that is, that he know the content of the level above and the level below, as an example.

If the system is not too large, a total group meeting may be workable. Such group or sub-group communication is independent of structure or documentation methods. It is simply a way to extend system understanding beyond the local task. There was discussion of the interchange problem. With computer personnel as dynamic as they are, interchangeability is desirable. It becomes important to be able to insert personnel into an ongoing, large development.

Certain structural attributes were discussed. One of these is the built-in data acquisition (or instrumentation). It is vital for large systems to provide for such measurement at the design level. This means that the means to accumulate data is incorporated in the program's operation. Also, there should be some facility for self-measurement. That is, the overhead involved in such procedures should itself be measurable. Another attribute is that the system should be piecewise operable. This property was broken into two categories. One category might be called incremental in that at every stage of the development there is a stand-alone system which can operate in a real environment. As an example, the development of a multi-programming system could proceed by developing first a mixed partition strategy changing it to use virtual storage. It would be operable at every pointin the development. Such a strategy is not always possible. The other category is the design of all modules for independent testing. Also important is specification of symbolic interfaces; that is parameterized structures where entities such as sizes and devices appear symbolically. It should be possible for example to simulate devices transparently, that is, simulate by software devices which are not physically available. There was discussion of auditing system development. The hierarchical structure and symbolic interface permit an independent audit procedure which checks documentation as well as function. The system audit group should do this by operating solely from documentation and interface description.

Another oft--neglected but desirable process is "post simulation," or design verification. It should be used after the system is operating to measure and refine the system and permit continued development.

Some time was spent on computer aided design of computing systems. Discussion as to whether there is something inherently different in designing computer systems using computers than in some other design areas. The conclusion was that there probably is not. The abstraction of any design process leads ultimately to algorithms which manipulate algorithms. Computers are well suited for such tasks but this does not seem to be unique to computer system design.

PROBLEM FORMULATION AND MATCHING

A VITAL SEGMENT

If we are to realize the potentialities of computing systems at a reasonable rate, we must look forward to the education and development of men and women across a very broad spectrum. It is easy to recognize the inevitable needs for certain kinds of people, such as:

- researchers into the understanding and expansion of what algorithms and computing systems can do.
- systems programmers competent to guide, lead, and do the development of major software systems.
- operators and routine programmers to run tens of thousands of installations.

As we attend to such clearly recognized needs, and, as well, to such crucial needs as increasingly effective attention to "wholeware"-- to the hardware and software of a computing system as a whole -- planned together as well as working together. We must not forget the vital segment of the spectrum associated with matching the problem to the computing system.

Problems do not arise in forms suitable for attack by computing systems. Those that seem to us "just made for a computer" came to that state by much human effort. If we are to tackle new problems -- or new versions of old problems -- effectively, bravely, and pioneeringly, and successfully, it will be because individuals or small groups have done a good job of problem formulation, because individuals or small groups have used the available computing systems and applications programs to deal effectively with these well and carefully formulated problems.

Neither phase of this task can be done wholly alone:

-Problem formulation often requires both repeated trial and exploration and insightful understanding of what computing facilities are really at hand.

-Bringing a good foundation to successful computing often requires guidance, sometimes repeatedly, from a version of the problem more true to life than the given formulation.

Recommendation:

It will be essential for the universities and colleges to greatly expand their student's opportunities to learn the essentials and principles of all elements from problem formulation to computing realization.

Recommendation:

Where a department of computer science wishes to lead in offering such opportunities, or to cooperate in offering them, that department should be especially encouraged and supported.

We feel it would be quite unrealistic to expect all departments of computer science to commit significant resources to this problem. (Indeed, there may prove to be no one area to which all these departments will, or should, attempt significant contributions). The need is large, all who can and would should help to shoulder the burden.

Recommendation:

Other departments with competent and interested staff should be encouraged and supported in providing opportunities for students to gain insight and knowledge in parts of all of this area. All reasonable efforts should be made to encourage interdepartmental cooperation and co-working.

If opportunities are to become widely available, there will have to be significant investments of time and efforts to develop materials ranging from case studies to organized presentations. Research into the credentials of how these problems are effectively formulated and brought to computation can and should have relation to mutual support with the efforts to develop materials.

Recommendation:

Both research and materials preparation deserve support, especially when each is planned to support the other.

George Forsythe

RETREADING

We find that there is a major national crisis of leadership in the areas of software engineering and computer science. To begin to meet this crisis will require the yearly addition of 3,000 persons to the level of a master's in computer science and 500 at the level of a Ph.D. in computer science.

We also find that each year U. S. universities are now graduating some ~ 1100 persons with a Ph.D. in physics. Many of these persons have the talent and desire to become faculty members at good universities, or research staff members at good research laboratories. Unfortunately, there is little demand for these persons in such organizations in their own field of physics.

We furthermore note that many physicists are interested in and able to turn their attention to a career in computing. We believe that many of these recent Ph.D.s in physics could be converted into top applications programmers in about a year, into good systems programmers in about two years, and into computer science faculty members in about three years. In all cases these times seem about two years less than corresponding times required for students newly entering graduate school in computer science. Moreover, the actual amount of faculty time required for these post-doctoral fellows would appear to be about half to two-thirds of that required for graduate students.

The principal advantage of the retreading approach is the speed-up in creating new computing experts over starting with conventional new graduate work. One cost would be the substantially larger salaries required for post-doctoral students than for graduate students. If there were an overload of, say, 100 post-doctoral students, there would be a substantial cost in finding faculty members to deal with them.

What has been said about physicists may apply also to mathematicians and, with lesser force, to some other fields.

Recommendation:

We therefore recommend that great attention be paid to the opportunity for creating applications programmers, systems programmers, and computer science faculty and research persons by retreading recent Ph.D.s in other fields.

22 July 1969

A POTENTIALLY LARGE MANPOWER REQUIREMENT

The effect of commercial time-sharing on manpower requirements for "professional" computer personnel will be a highly important one. For the first time, highly competent professional help will be given to those 15,000-20,000 small installations around the country using lower cost computers in small data-processing facilities.

For the commercial time-sharing groups to compete effectively, they will have to specialize their services for some segment of the organized technology, for example: machine tool tape preparation, type-setting and hyphenation, wholesale accounting, small-scale inventory control.

Each of these time-sharing organizations must have highly effective computer systems programmers to develop languages, generalized routines, "hand-tooled" algorithms, etc., to satisfy the individual needs of the user.

Users will try competing services against each other for cost, speed and breadth of capability. Those time-sharing commercial groups with the most professional staffs (all other characteristics--management, marketing, etc., being equal) will survive this very intensive competition.

It may be that the 15,000-20,000 small machines, most of which do not have any professional computer staff, will be merged into the commercial time-sharing networks, with this many (15,000-20,000) professionals needed to work for them indirectly. These men and women must be professionally trained in structure of time-sharing systems, managerial processes, data structures, operating systems.

J. W. Carr IV
Monday, July 21, 1969

John W. Carr, III
23 July 1969

The Need for Increased Education in Software Engineering as a Subset of Computer Science

One presently arising class of computer problems differs in both quantity and quality from those that have been most important up until now. Such problems are characterized by:

1. Large size
2. Complexity of structure
3. Lack of formal descriptions
   (here follows one or more further characteristics)

Examples of such problems today include operating systems for large-scale computers; manufacturing systems for large aircraft; construction, retrieval, and analysis of large data bases; air and ground traffic control; management information systems, command and control systems; (here follows a list of other problems)

These problems fall into a category that represents an important area concurrent to and perhaps a part of computer science. The study in and of this technology has been proposed to be called "software engineering"; some of what has been called "systems engineering" or "operations research" falls directly into this problem area.

Such systems have in the past been organized out of groups of human beings as control elements, human-accessed data storages, and direct human communication. The coming of the computers, as well as the expansion of applications of physics and technology, now requires effectuation and automation of systems in which humans can no longer play a detailed part. Where in the older system they served as local control elements, the response time and data rates required no longer allow this participation.

Such systems must now be developed by teams of human beings no one of whom, in general, can view the problem as a whole. The digital computer now serves as data storage, communications device and monitor, control element, and manager of the overall activity. Humans interface the system and must be satisfactorily served. The systems are characterized by large numbers of program steps, complex mappings into present-day computer structures, and need for optimization within a set of complex constraints.

The design of such systems, and their prototype construction via computer programs, is today in its infancy. Examples up until now have ranged from successful special purpose systems for one-problem applications (such as airlines reservations) to less successful general purpose systems for improvement of computer utilization (such as batch and time-sharing operating systems.)

It is in this area of design and development of large computer programs for such large systems that there appears to be a lack of organized instruction in higher education, here or anywhere, at the present time.

Without the educational development of persons who can work on the computer-oriented portions of such problems, the problems will be able to be attacked only on an intuitive <u>ad hoc</u> basis. It is expected that the fundamentals of computer science will serve as a scientific basis for the education of such persons, but that special areas and tools of application must also be taught.

The products of such an educational curriculum will serve as the cadres of the teams that will construct the computer program portions of such systems. (continue)

One of the requirements of such an educational experience is the availability of an effective laboratory experience. (continue)

"Software engineering" is not a good phrase and its use should be discouraged. The reasons are as follows:

1. Hardware and software are intimately related. Ten years from now many functions that are now handled by software will be either hardware functions or shared hardware software functions. The term "software engineering" emphasizes the distinction. It is very important to emphasize the interrelationships. "Computer Science" is a far better term for this than "software engineering."

2. A curriculum in "software engineering" at a university would of necessity be housed in the School of Engineering. This could create great confusion in schools in which Computer Science is not currently housed in the School of Engineering.

Purdue Masters Degree Program in Computer Sciences

There are 4 major areas:

1. Numerical Analysis
2. Logic and Automata (including formal Languages)
3. Systems
4. Application areas.

Most terminal masters degree students are in the systems area. They generally take one course in Numerical Analysis, one in the area of Logic and Automata and the following Systems courses:

1. Computer Organization
2. Programming Languages
3. Compilers
4. Operating Systems

They will also take 2 or 3 courses in the applications area. Courses in applications include:

1. Information Retrieval
2. Artificial Intelligence
3. Simulation
4. Mathematical Programming
5. Computer Graphics

A total of 33 hours are required. Students entering with a strong undergraduate background in computer science may get the degree with fewer credits. A thesis may replace 9 credit hours but is not currently encouraged because of staffing problems.

The above program leaves room for several electives. Students are encouraged to take courses in Mathematics, Statistics and Electrical Engineering.

Note that there also exists a new joint masters degree program between Computer Science and Industrial Management.

REPORT OF THE RESOURCES COMMITTEE

Dr. Alan Perlis

July 23, 1969

We have a number of figures and tables which have come out which might be of interest. In education, for example, the University of Waterloo has chosen to commence with the Bachelor of Science program in computer science and to develop from that upward to the MS and Ph.D programs. In the United States development in the opposite direction has generally been followed. It is recognized by Waterloo that the first approach, their approach, is a somewhat more difficult path to follow, it being more difficult to upgrade a Bachelor's program than to downgrade a Ph.D program.

However the committee strongly feels, and this is the first recommendation, that major educational efforts should be spent in the development of Bachelor of Science programs in Computer Science in the USA over the next few years. Furthermore, the committee concurs with the Waterloo experience that the program should include significiant amounts of practical, hands-on experience with real computer systems problems. Hence the committee feels, and this is a second recommendation, the BS program will be greatly aided by and should include laboratory courses and/or cooperative ventures with industry and government during the school semesters or over summer periods. The committee does not feel that the development of MS programs has the same priority as the two extremes, BS and Ph.D. Indeed, the MS program contains material only superficially different from the BS program and serves mostly as a springboard for those switching fields and as consolation prizes for those unable to complete Ph.D. programs. The committee next considered the needs of the non-computer scientist being educated in the universities, since it became clear it would not be feasible to educate as many specialists as one might need in this field in the next 10 years. The first calculation we made we call the Waterloo computation. At Waterloo there is an IBM 360/75, costing 125K per month. Student jobs account for 1/10 of the system time on that machine or if you will costing about 12.5K per month. Considering cost in the support or overhead equal to that of hardware we have a cost of $25,000 a month for student jobs. For that cost the productivity is 5,000 runs a day or 100,000 runs per month. Considering a productivity of four cracks at the machine per problem, this means that that system is capable of absorbing 25,000 problems per month. Consequently, given a student population size and a number of problems one can come up with various estimates as to what it costs to provide undergraduate computer experience for the non-computing specialist, i.e., someone who does problems of a relatively small size. We came up with one figure assuming 25,000 students in the university of one dollar per problem per student per month. The size of those problems is that their programs are limited to one second of cpu time and the students are not charged to disc file time but they generally do not include much file work.

We might at some later time have a few words to say about the overall picture of the way the system flows at Waterloo. In any event over a ten month academic year a system of this kind could support students giving them 10 problems over an academic year at a cost of 10 dollars per student per year in a 25,000 student population which almost reaches the student population of the largest universities we have in the United

States today. Now this figure is substantially below the figure in the Pierce report which runs closer to 50 or 60 dollars per year. That means if we wish to attain the Pierce report figure we could have the student doing 50 problems per year, which is probably much too heavy a load for non-computing specialists!

Now this leads us to make a third recommendation. We recommend that funds be made available so that a cost analysis study can be made of the specification and use of various systems for handling bulk student jobs for the non-computing specialist at different student population levels. It would be hoped to provide a study that would say - at the cost level at which we have spoken, given a student population of 1,000, system A would provide computation at the rate of $50 per year at a level of between 10 and 50 jobs or problems per year. At a student population of 5,000 system B will similarly provide at 10,000 system C, at 30,000 system D, etc. Such a specification of systems is not now available to the educational community. Of course, these systems need not be unique. There can be many systems in each of these four categories. Nevertheless, it is the feeling that at all four of these student population levels, 1,000, 5,000, 10,000 and 30,000, systems can be found which are of economical comparison to the Waterloo system.

We arrived at an estimate that to turn out 300 Ph.Ds per year in computer science, we were talking about an estimated machine cost of $9 million a year. This is the machine cost required to support Ph.D theses and Ph.D education at the level of 300 Ph.Ds per year. Thus: to produce 300 Ph.Ds per year it is estimated that it will take 30,000 dollars per Ph.D in machine time or a total of 9 million dollars in machine time for the Ph.D production of 300 Ph.Ds.

For the Bachelor of Science program in computer science, assuming six courses in their program that are in the core of computer science, thus not counting auxiliary courses, and an education program that will turn out 15,000 B.S. computer science students per year, a figure of 15 million dollars per year in computer time was arrived at. The calculations will be laid out in more detail in the report.

For the Master of Science program, a figure of 5 million dollars per year in hardware costs was obtained.

The total cost in hardware is 29 million dollars per year. One of the figures that we used was that the EDP industry would be taking in about 100,000 people per year. What percentage of these should be Ph.Ds? Figuring that one percent should be Ph.Ds we get a desirability of producing a thousand Ph.Ds a year. Our feeling on the matter was that by 1975 we might be able to produce 1000 Ph.Ds in computer science, but that we would not be able to produce 1000 Ph.Ds per year by 1975. If you can get up to about 300 by 1975 this would be about what we could expect. It seems to double about every two years.

From whence comes this figure of 15,000 BS students per year? Is it attainable? At the present time in engineering and mathematics the output per year is of the order of 50,000. Now assuming there is no major change within engineering and science schools but that quality computer science undergraduate programs do come into being, how many of the 50,000 per year could we siphon off into computer science? We believe that we could without a great deal of heavy advertising or pressure of any sort get 20-30% of the present undergraduate enrollment that are now in mathematics and engineering programs diverted into computer science programs, if there were existing quality undergraduate programs in computer science. That means of the 100,000 per year that are required in the EDP area, 85,000 are probably going to have to remain or be non-computer science baccalaureates. We also made an estimate of computer science faculty costs and came up with an estimate of 45 million dollars per year for that part of computer science faculty costs devoted to computer science education alone at the three levels being well aware, of course, that there are other costs associated with their education outside the computer science department. But we're talking now about cost of a faculty of about 1500. Waterloo argues that they are producing 200 Baccalaureates per year to service 1,000 computers in the province of Ontario. There are 67,000 computers in the USA. Consequently, if we assume that the ratios are comparable, this leads to 13,400 output in the USA to service these computers, if we adopt that ratio. This compares reasonably well with our 18,000 figure.

John Giese came up with another set of figures arrived at differently from the figures just cited which tend to corroborate this level by about 1975:

A conservative estimate of the prospective demand for the products of the Computer Science educational system.

A. In the long run the overwhelming majority of computer science graduates at all degree levels will go to non-academic employment. For the estimates we shall make later, we shall need to estimate the number of "computer science" positions which should be filled with computer science trained people if possible at computer installations in the U.S.

(i) It has been said that there are about 67,000 computers in the U.S. in 1969.

(ii) Let us <u>assume</u> the following distribution of sizes of installations and staff.

SIZE OF INSTALLATION	LARGE	MEDIUM	SMALL
NUMBER OF THIS SIZE	1000	10,000	56,000
AV. CS EMPLOYEES PER INST.	100	30	3
AV. NO. OF PH.Ds PER INST.	5	1	0

Then the desired number of TOTAL "CS" EMPLOYEES

$1000 \times 100 + 10{,}000 \times 30 + 56{,}000 \times 3 = 568{,}000$

and the desired number of TOTAL "CS" PH.Ds = 15,000.

(iii) These positions are not now filled by computer science graduates. We assume it would be desirable to replace them gradually by computer science graduates to upgrade the computing profession

B. Let us assume that the computing profession remains static at about this level, i.e., that increases in efficiency make new people available for an inexhaustible set of new problems. Let us assume that we have a rather rigid slowly varying working population, like the Civil Service. This may not be too unreasonable to assume, since these professionals might become union-organized (as teachers are now). If we assume a working life of about thirty years, then in the steady state we shall have to replace about $\frac{568{,}000}{30} = 19{,}000$ "CS" employees per year and about $\frac{15{,}000}{30} = 500$ "CS" Ph.Ds per year.

C. Composition of 19,000 computer science graduates.

If we assume that about 20% of these graduates seek advanced degrees, this means about 4,000 advanced computer science degrees per year. If we claim 500 PH.Ds per year, this leads to a need for

500 Ph.Ds
3,500 Masters per year
15,000 Bachelors

in the computer science area.

D. Conservatism of this estimate.

(i) The assumed static "CS" employee pool is about $\frac{500,000}{200,000,000} = 0.25\%$ of the total U.S. population.

(ii) 19,000 graduates per year is about half the number of engineering grads (40,000) per year. That doesn't sound unreasonable. Computer technology should be about as widely appliable as engineering.

(iii) For comparative purposes consider the fraction of our man-power resources devoted to medicine and associated subjects. We produce about 9,000 physicians per year. They must be backed up or supplemented by about 18,000 nurses, technicians, dentists, and various forms of physiologists, etc. As a guess, about 27,000 graduates per year are devoted to problems of health.

You might argue that since medicine absorbs a fairly small fraction of our economic output, and since computing is (or will be) involved in all of man's activities, including medicine, perhaps the output of computer science graduates could safely be increased to the level of medicine (and associated graduates) or 27,000 eventually.

(iv) Some "CS" enthusiasts assert that the growth of "CS" jobs may be 100,000 per year.

In a steady state process, with thirty-year working life, this would lead to a CS employee-pool of

$$30 \times 100{,}000 = 3{,}000{,}000.$$

If the population of the U.S. remains static at 200,000,000, this would mean that the pool would contain about 1.5% of our population.

E. You have Bruce Gilchrist's estimates of staffing requirements to provide faculty for these hordes of computer science students.

F. Nothing has been said about the provision of refresher courses for the people in the pool who will constantly become obsolete. If you provided a "refresher" or updating course once every five years, this comes to 0.2 course (three weeks?) per year. Even if you restricted this updating to the lucky employees at the large and medium installations, somebody would have to provide about

$$0.2 \times 400{,}000 = 80{,}000$$

student courses/year. Even if these things operate at 100 students per section, you would have to run about 800 refresher course-sections per year.

If we aren't so generous and send only 10% of the pool to refreshers, this cuts the total to 80 course sections per year. That ought to be a tolerable burden for the educational system.

G. Nothing has been said about providing computer "service" courses for non-computer science students.

The other computations I performed are very original notes merely paralleled (for very assumed populations) the calculations of Gilchrist. I have therefore not repeated them here.

It may seems ridiculous to staff the small installations with graduates. To handle this I suggest that we reinterpret our imagined program. Let us say that we provide instruction and facilities to produce 190,000 graduates per year. If about a third of these drip out after the first two or three years, they would probably have to be content to work at the small institutions. Actual graduates go to medium or large places.

I would assume that the <u>computer industry</u> would be included as part of the large installations.

One final point. The figure of 15,000 baccalaureates is considerably lower than we would like. Arguing that 100,000 entries into the EDP area a year are needed, we figure that 25,000 come from business schools and industrial administration programs, 25,000 by upgrading from their current positions. This leaves 50,000 coming from colleges, and we're only providing 1/3 of that. That means that 35,000 are going to come from a lower educational level than baccalaureate computer science programs. Jim Rowe mentioned that one of the consequences of providing 15,000 baccalaureates in computer science will be a temporary diminution of the number of people needed in the field. But we all agreed that this diminution would be temporary. The more trained people that you have presumably the less total number you need. However, Rowe felt that he would really prefer that all 100K came out of baccalaureate programs in computer science.

We merely want to point out that the figure of 15,000 per year is, in our judgment attainable right now, if baccalaureate programs are introduced.

John Giese
J. W. Graham
Bruce Gilchrist
James Rowe
A. J. Perlis

23 July 1969
Spinrad

Education in Computer Science

We see Computer Science as a coherent academic discipline. The educated Computer Scientist will be trained in both hardware and software--the inextricably interwoven elements of his field. Graduate study will, at first, lead to a broader understanding of complex hardware/software systems. Further study, (to the Ph. D. level) will naturally lead to a more penetrating specialization.

We believe that there is a core of knowledge fundamental to the undergraduate's education and independent of his future course of study. For this reason we specifically reject the notion of a "homogenizing" entry year of graduate study whose object is to correct the deficiencies (soft or hard) in the student's previous education (hard or soft). For this same reason we reject the concept of two educational paths-- one leading to a terminal professional degree and the other leading to further graduate study.

We find no compelling reasons that lead us to suggest that Computer Science is appropriately placed within any particular classical academic discipline. Our strong concern is that in a given university, there be only one undergraduate program concerned with the science and engineering of computing. (A student wishing to enter Computer Science from an "adjacent" field will have the traditional academic remedy of "making up" the necessary prerequisites.)

In broad terms, the areas of study we consider essential and at the core of the Computer Science undergraduate program are:

1. Mathematics
2. Physics
3. Hardware Technology
4. Programming
5. Logic Design
6. Software Structures
7. Subsystem Design
8. Computer Organization
9. Compilers
10. Systems Programming
11. Computer Systems Laboratories
12. Systems Applications

These are, of course, in addition to the fundamental education traditional to the undergraduate curriculum.

(Revised)

Miller
23 July 1969

Computer Systems Laboratories

We consider the laboratory-experimental aspect of the training of students in computer science to be vital to their development. We therefore recommend the establishment of computer systems laboratories as part of the curriculum of both undergraduates and graduates in computer science.

There are many substitute plans that could conceivably serve to fulfill the same purpose as the computer systems research laboratories, e.g. summer employment in industry, cooperative work projects with industry, or part-time employment in a computation center on campus. Each of these alternatives was explored by the committee and considered to be difficult for one or more reasons. Principally, these substitute plans lacked the supervised directed planning of an organized laboratory. The success of any of these alternatives is quite personnel dependent.

In the laboratory course the students are expected to work in a team of about six students under close supervision of the faculty member and teaching assistant. The student team is expected to concentrate on design, documentation, scheduling of their work, performance evaluation, efficiency, error recovery, diagnostics, maintainability and other features of a well-engineered system.

It is expected that each student should take the equivalent of two of the below laboratories during the course of his study.

We propose the following computer systems laboratory courses as basic to a graduate computer science curriculum:

C.S. Lab. 1. Construction of Assemblers and Computers

C.S. Lab. 2. Construction of Operating Systems

C.S. Lab. 3. Construction of Terminal Systems (both typewriter and graphics)

C.S. Lab. 4. Construction of Switching, Communication and Process Control

C.S. Lab. 5. Construction of Large Data Base Systems

In addition, we consider two additional laboratory courses that could be given in addition to or in place of the above five:

C. S. Lab. 6. Management of a Computer Facility

C. S. Lab. 7. Construction of Large Application Systems

The above laboratory courses, particularly the first five, are graduate level courses given concurrently with or following a lecture course covering the subject matter. It is intended that the lecture course cover the theory, models, and formal aspects of the subject matter. The associated laboratory is intended to provide the student an experience that will sharpen his understanding of the theory and, so will, have given him an understanding of the practical problems of implementing large systems.

The companion lecture courses associated with the above listed laboratory courses are given below:

Laboratory Course	Lecture
C. S. Lab. 1.. Construction of Assemblers and Compilers	Lecture course such as I5 and/or A1 from Curriculum 68, A Report of the ACM Curriculum Committee on Computer Science. Includes definition of formal grammars, arithmetic expressions and precedence grammar, algorithms for syntactic analysis, recognizers, semantics of grammar, object code generation, organization of assemblers and compilers, meta-languages and systems.

C. S. Lab. 2. Construction of Operating Systems	Lecture course such as I4 and/or A2 and/or A3 from Curriculum 68. Includes operating systems characteristics, structure of multiprogramming systems, structure of time-sharing systems, addressing structures, interrupted handling, resource management, scheduling, file system design and management, input-output techniques, design of system modules, sub-systems.
C. S. Lab. 3. Construction of Terminal Systems (both typewriter and graphics)	Lecture course such as I4 and A6. Includes text editors, string manipulations, data structures for text editors, job control languages, data structure for pictures, syntax and semantics of terminal and graphics language, control of the console system, meta-languages and systems.
C. S. Lab. 4. Construction of Switching, Communication Systems, and Process Control	Lecture course such as I4 and/or A2 of Curriculum 68. Includes traffic control, interprocess communication, system interfaces, realtime data acquisition, asynchronous and synchronous control, telecommunication, analog-to-digital and digital-to-analog conversion.
C. S. Lab. 5. Construction of Large Data Base Systems	Lecture course such as A5 and A8 of Curriculum 68. Includes organization of large data base systems, data organization and storage structure techniques, data structuring and inquiry languages, searching and matching, automatic retrieval, dictionary systems, question answering.

These laboratories will require a certain amount of "hands on" use of a substantial computer facility. In some installations it may be possible to carry out the entire project in a subsystem or partition of a larger system. In that case the use of the subsystem would have to be dedicated to the project for a substantial portion of time.

We believe that a team of six students can be given a very significant experience for $1,000.00 per student or $6,000.00 for the whole team for a one-quarter laboratory.

These laboratories are presented as examples of laboratories that might be given. Each school will have different staff and facilities available and will present variations on this proposal. The important emphasis is the supervised hands on experience with attention to the practical aspects of the system.

Subcommittee
Miller, Chairman
Coates
Andree
Gruenberger
Spinrad
A. Forsythe
Seely

First Draft

R. Andree

<u>A GRADUATE CURRICULUM IN THE VARIOUS COMPUTING SCIENCES</u>

The initials I+CS used in this paper may be read as Information and Computing Sciences or as Information Science or Computer Science or as any other phrases that describe the science and the art concerned with the study of the complex structure that surrounds computers.

The phrase "Computer Science" will continue to have changing meanings as staff interests and abilities change. The committee recognizes that valid graduate programs may differ widely in structure, purpose and implementation, and feels that such diversity should be encouraged, not stifled. We suggest that the following observations merit serious consideration in all graduate programs in the various aspects of I+CS.

1. The core of a graduate program in I+CS should contain a blend of

    Pure theory (Math, Physics, etc.)

    Hardware-software systems

    Laboratory experience involving both hardware and software

    Applications of existing hardware-software systems to realistic problems from various areas

    Administrative management (operations research)

   This should provide an extension (not a repetition) of the students' undergraduate experience.

2. A person who holds a master's degree in I+CS should be able to read and understand (with reasonable effort) more than half of the articles <u>in his area of specialization</u> which are printed in the existing computer related journals. A person with a Ph.D degree

should be able to understand a much higher percentage of the articles in his area and in related areas and should <u>be able to create</u> similar journal articles.

3. A student whose primary interest is in an existing discipline (electrical engineering, chemistry, mathematics, business administration, industrial engineering, economics, etc.) should continue to earn the Ph.D degree in the appropriate department possibly with a <u>minor</u> in Information and/or Computing Sciences rather than creating myriad diverse Ph.D's in the "Applications of I+CS". The Ph.D in I+CS should be primarily for students interested in computing (including hardware-software and abstract theory) rather than in the applications of computers to research and work in other specific areas, vital as this may be.

4. (a) The masters program of the person who will become a "professional practitioner" of the computer art should not differ markedly from that of the pre-Ph.D in I+CS.

   (b) There should be both undergraduate and graduate "service courses" in I+CS which include appreciably more than mere programming in compiler language. They may be the same or different courses from those of 4(a), but should be substantial in nature and include an understanding of the basic concepts of hardware-software interface as well as related elementary theories. Possibly there should be a second, very broad brush masters degree for students from other disciplines who will then return to their own disciplines either for employment or for further training in that discipline.

5. Courses in computer related subject matter which are currently being well taught in existing departments should continue to be taught by those departments (possibly with crosslisting). If new courses are needed, which existing departments are well qualified to teach, they should be urged to do so before the I+CS department undertakes additional teaching duties.

6. A department of I+CS should be <u>aware</u> of the publications related to curriculum including at least

A.C.M. recommendations in Curriculum 68

C.U.P.M. recommendations for a curriculum in Computer Science

D.P.M.A. recommendation for certified Data Process Certificate

A.C.M. recommendations for a curriculum in Business Data Processing (being prepared by M. Tondow and others)

COSINE recommendations on Engineering Computer degrees (now being prepared)

7. Students of I+CS at both the undergraduate and the graduate levels should have both theory courses and <u>related</u> laboratory experience (the critical word is <u>related</u>) which will focus their attention on the organization, implementation, and documentation of larger scale computing systems.

8. [Your suggestions are welcome.]

9. The committee hesitates to recommend specific course material other than that suggested in 6 above, but does sincerely recommend the creation of two courses not readily available at present.

   a. <u>Discrete Mathematics (with an awareness of computers)</u>
   To contain material on matrices, probability, logic, graphs, combinatorics; automata theory, computability, linguistics and possibly some simulation theory at a level suitable to build on the students' undergraduate preparation, but not in such depth that a reasonable selection cannot be completed in one or two terms. Suitable references for future reading are essential.

   b. <u>Basic Computer Components (hardware and software)</u>
   To contain current information on hardware-software interphase and their symbiotic relations and hang-ups as well as possible near future changes. Should be possible in one semester.

7/22/69

Morris

REPORT OF THE GOALS COMMITTEE

We propose that computer science is going to be the handmaiden of the sciences. We can make models of the external world inside automata such that vital human problems can be solved.

The main goals of progress in computer science apply broadly to computation, computing systems, and computing processes. We can conveniently identify six such goals:

Gaining insight through organization

Gaining factual knowledge

Attaining new feasibilities

Improvement of productivity

Better interaction with people

Export of insight

The character and breadth of these goals can be better appreciated if they are illustrated by short lists of major problems.

Problems were chosen whose scope can easily be understood by persons other than those who are experts in some part of computer science.

Some of the problems overlap other disciplines but their solution would require the development of currently non-existent tools of computer science.

Gaining insight through organisation

1. Develop a formal language which can unambiguously and systematically represent the content of natural language sentences.
2. Determine a general procedure for associating semantics and syntactic structure.
3. Develop models for the structure of modern operating systems.
4. Develop a theory of the algebraic structure of computer algorithms.
5. Develop general theories that match both computing structures and language structures to the structure of problems.
6. Develop a deterministic model for machine synthesis by learning.
7. Systematise the methods that human beings use to solve problems.
8. Develop general methods for degarbling data and instructions.
9. Invent new machine organisations for large or novel problems.
10. What are the steps by which metasystems are constructed (e.g., metacompilers, metaalgorithms, metamachines).
11. Understand in adequate detail the requirements placed on large information service systems. These can include speed of access, diversity of access, security against loss or failure, privacy, economic feasibility, need for catastrophe-free changes.
12. Understand how to approach the fulfillment of these requirements through overall system structure and through software architecture and fabric.
13. Develop a more and more adequate theory of large information service systems.

Gaining factual knowledge

1 to 5. Determine the problem solving efficiencies of:

- parallel computers
- associative memories
- distributed logic machines
- memory-centered computer systems
- parallel languages

6. Determine the minimum number of operations to invert a matrix.
7. Find out how to design arbitrarily large nets that function reliably out of faulty elements.
8. How much memory is needed to test whether a given string is a sentence of a given language.

9-10. What is the least amount of work (or storage) required to compute function f.

New feasibilities - internal

(We divide new feasibilities into three parts)

1-4. Develop an adequate language for describing:

- languages
- data
- graphic objects
- machines

5. Develop a good (effective and handy) algebraic symbol manipulation language.

6. Write a very high performance compiler for PL-1 on System 360.
7. Find out how a computer can have very fast access to a memory of 10^9 words or more.
8. Learn how to plan systems and programs that are much more likely to allow easy insertion of unexpected changes.
9. Develop precise descriptions of this and that.

New feasibilities - service

1. Implement good interactive data analysis.

2-4. Develop systems automating:

- the operations of elementary analysis
- the solutions of ordinary differential equations
- the solution of increasingly large classes of partial differential equations

5. Produce an effective algebraic symbol manipulation processor.
6. Produce a program to play championship chess.
7. Learning to effectively instrument computer systems to identify bottlenecks to productivity, and publishing the results. (This may require us to restructure our systems.)
8. Make significant advances in theorem-proving.
9. Find better methods for degarbling data and instructions.
10. Find out how to synthesize high-quality visual images (TV or movie quality).

<u>New feasibilities - external</u>

1. Plan, and show the feasibility of, programs and computing facilities needed to allow a robot to operate effectively in environments hostile to humans (Mars, deep ocean, etc.).

2. Plan, and show the feasibility of, programs and computing facilities sufficient to operate an automobile in ordinary traffic.

3. and 4. Develop techniques for:

- speech recognition
- left-thumbprint recognition (as a substitute for signatures)

5-7. Find automatic methods of interpretation for:

- medical X-rays
- EEG's
- bubble-chamber pictures

8. Implement mechanical translation between natural languages.

<u>IMPROVEMENT OF PRODUCTION</u>

This is a goal of great importance, toward which progress is usually made in smaller steps.

1. Bootstrapping from easy problems to hard problems.
2. When to interpret and when to compile.
3. When to bind and when to leave free.

<u>Export of insight</u>

1 and 2. Contribute to the development of realistic models of:

- thought processes
- the brain

3 and 4. Contribute to the development of effective

- sensory prostheses
- intellectual prostheses

5. Contribute to learning how to do computer-aided instruction well.

Relationship with people

1 and 2. Learn more about human input/output capabilities, including:

- understanding the potentialities of graphic input to human intuition.
- understanding human output capability as a parallel channel mechanism, and the possibility of exploiting this as a computer input.

One presently arising class of computer problems differs in both quantity and quality from those that have been most important up until now. Such problems are characterized by:

1. Large size
2. Complexity of structure
3. Lack of formal descriptions
4. Possible real-time requirements
5. Requirement for multiple access to programs and data
6. Constantly changing data bases
7. Extremely high penalties for major system failures

Examples of such problems today include operating systems for large-scale computers; manufacturing systems for large aircraft;

construction, retrieval, and analysis of large data bases; air traffic control; management information systems, command and control systems; (here follows a list of other problems)

These problems fall into a category that represents an important area concurrent to and perhaps a part of computer science. Some of what has been called "systems engineering" or "operations research" falls directly into this problem area.

Such systems have in the distant past been organized out of groups of human beings as control elements, human-accessed data storages, and direct human communication. The coming of the computers, as well as the expansion of applications of physics and technology, now requires effectuation (and automation) of systems in which humans can no longer play a detailed part. Where in the older system they served as local control elements, the response time and data rates required no longer allow this participation.

Such systems must now be developed by teams of human beings no one of whom, in general, can view the problem as a whole. The digital computer now serves as data storage, communications device and monitor, control element, and manager of the overall activity. The systems are characterized by complex mappings into present-day computer structures, and need for optimization within a set of complex constraints.

The design of such systems, and their prototype construction via computer programs, is today in its infancy. Examples up until now have ranged from successful special purpose systems for one-problem applications (such as airlines reservations) to less successful general purpose systems for improvement of computer utilization (such as batch and time-sharing operating systems.)

It is in this area that there appears to be a lack of organized instruction in higher education, here or anywhere, at the present time.

Without the educational development of persons who can work on the computer-oriented portions of such problems, the problems will be able to be attacked only on a case by case basis. It is expected that the fundamentals of computer science will serve as a scientific basis for the education of such persons, but that special areas and tools of application must also be taught.

The products of such an educational curriculum will serve as the cadres of the teams that will construct the computer program portions of such systems. (continue)

One of the requirements of such an educational experience is the availability of an effective laboratory experience. (continue)

Conventional undergraduate mathematics

* Important
** Important need more

Mathematics

*Logic

*Algebra (???--

Analysis

Geometry (linear, projective)

*Information/communications theory

**Probability

Mathematical sciences

**Mathematical Statistics

**Data Analysis

**Operations Research (queuing theory, optimization theory, mathematical programming

*Math linguistics

**Design of experiments

*Numerical analysis - optional control theory

Engineering

Solid state devices — Computer circuits; Computer memories

Mechanical engineering

Philosophy/history of science/mathematics

**"Scientific method" (Polya/Poincare study)

Psychology

Learning theory

Central nervous system

**I/O channels - to/from memory and intuition

**Problem solving

**Perceptual mechanisms--gestalt, etc.

Areas of application

Any/all

Computer Science

- Formal languages
- Hardware behavior
- Computer algorithms
- Automata/machines
- Error detection/reliability
- Pattern recognition
- Interactive programming
- Artificial intelligence

Linguistics

- Descriptive
- Semantics
- Syntactics

Areas in the mathematical sciences which students with an undergraduate major should give increased attention as certainly useful for research in computer science:

Special attention needed, with certainty:

- More algebra
- Probability
- Mathematical statistics
- Data analysis
- Operations Research (queuing theory, optimalization theory, mathematical programming)
- Design of experiments
- Combinatorial analysis

Special attention deserved, good bets:

- Human I/O channels, memory and intuition, including perceptual mechanisms
- Scientific method/problem solving (history/philosophy of science, Polya/Poincare, psychology)

Is there similar list for engineering and physics bachelor graduates?

A PROPOSED UNDERGRADUATE COMPUTER SCIENCE PROGRAM

Dr. Alan J. Perlis

		Freshman	Sophomore	Junior	Senior
1st Sem.	1.	Anal I	Alg I	Prob & Stat I	OR II
	2.	Prog I	Prog III	Comp.Sys. II	Abstr. Sys. III
	3.	Phys I	Anal III	Lab II	Elect. II
	4.	Hum.	OR I	Abstr. Sys. I	Elect. III
	5.	Hum.	Hum.	Hum.	Hum.
2nd Sem.	1.	Anal II	Lab I	Prob & Stat II	OR III
	2.	Prog II	Alg II	Abstr Sys. II	Comb. Anal.
	3.	Phys II	Prog IV	Lab III	Administration and finance
	4.	Hum.	Comp. Sys. I	Elect. I	Elect. IV
	5.	Hum.	Hum.	Hum.	Hum.

NOTES:

Hum = Humanities
Prob. & Stat. = Probability and Statistics

Programming I - IV

1. Algorithms, programs and language organized by data
2. structures
3. Machines and their programs
4. Problems associated with the management of programs: file systems, libraries; and Proofs of termination and correctness; Verification, representation and documentation of programs

Computer Systems I and II

1. Devices
2. Representation
3. Synthesis
4. System design

Abstract Systems I to III

Logic: Propositional Calculus; 1st order Predicate Calculus
Automata Theory: Finite state machines and regular expressions
Turing machines
Computability
Stages of computability
Math, Linguistics, correspondences (recognizers as machines)

Operations Research

OR I Optimization Techniques
OR II Simulation Techniques and modeling
ORIII Processing requirments of large data systems

Computer Science Laboratory I - III

1. Building, enhancing, auditing a sub-routine library
2. Interfacing two systems
3. Design of a system
4. Completion of a system
5. Managing a system design and construction

Sub-college education

A. Forsythe

It seems appropriate that this group give some consideration to pre-college education.

1. The increasing impact of computers on the life of the average American requires that every person understand something about them. Secondary school is a natural place for this. Since schools are pre-organized into departments such as mathematics, science and business which should teach computing? What should be taught?

2. The resources group sees 100,000 persons going into computing each year. All these people go through secondary school--even those who never get to college. Would their education be more relevant if they had had a computing course in highschool? What computing? How?

3. The present trend in highschool mathematics is toward including more and more calculus. Two full semesters of college level calculus are incresingly offered at the secondary school. Should this trend be encouraged? Would a computing course be preferable? Or some other mathematics?

4. Many processes that children learn to execute in elementary and secondary school are easily expressible as algorithms. In fact some children understand and remember an algorithmic expression (by flowchart for example) better than they do the succession of English sentences or mathematical equations. If this type of activity were explicitly encouraged by this group, it might be taken up by more curriculum creators.

Summary: Computing activity in secondary schools is growing very rapidly. Guidance and information exchange are urgently needed.

File

Carnegie-Mellon University

Department of Computer Science
Schenley Park
Pittsburgh, Pennsylvania 15213
[412] 621-2600
[412] 683-7000

June 16, 1969

Mr. Kenneth Olsen, President
Digital Equipment Corporation
899 Main Street
Maynard, Mass. 01754

Dear Mr. Olsen:

Enclosed is a copy of the invitation that was sent to the invitees for the Computer Science and Engineering Board Conference on Computer Science Education. As a board member you are invited to attend and participate in this conference.

You will receive preparatory material as it becomes available.

Sincerely yours,

Alan J Perlis (dg)

Dr. Alan J. Perlis, Head
Department of Computer Science

AJP:dg
enc.

Carnegie-Mellon University

Department of Computer Science
Schenley Park
Pittsburgh, Pennsylvania 15213
[412] 621-2600
[412] 683-7000

May 20, 1969

A conference to study computer science education in the United States will be held July 21 through 25, 1969 at the Hilton Hotel in Annapolis, Maryland. The conference is being sponsored by the National Academy of Science Computer Science and Engineering Board under a grant from the National Science Foundation. The purpose of this letter is to invite you, as one of approximately 40 invitees, to participate in the work of this conference. Naturally your travel and living expenses will be provided by the Board; though if your organization can support your expenses it would be appreciated since the grant supporting the meeting is of a limited amount.

The Computer Science and Engineering Board has been formed to provide a focus for those aspects of the computer field that are important to science in general and the federal government. Attached to this letter is a document that describes the purposes of the Board.

The conference will be organized to make maximum use of the participant's capabilities in the time available. It is planned to hold all day meetings during the entire week and to focus our attention on two specific topics:

1. Graduate education in computer science
2. Education in software (and hardware) systems

The conference discussions and conclusions may broaden considerably beyond these two areas; nevertheless they seem reasonable for initiating and focusing discussion. With each of these issues there will be two major technical concerns:

A) Resources: By resources is meant the creation of input-output models relating the development of programs, production of students and faculty, and the needs of industry and government for people so trained. Furthermore, a timetable establishing the velocity and acceleration of these programs should be produced. In accord with the postulated growth, a study should be made of the sources (plant, people and money) required to provide this educational development.

B) Content: A thorough study should be made of the content of the undergraduate and graduate programs to be labeled as computer science. Furthermore, an audit of existing programs should be made to gauge what distances exist between what is being done and what should be done. Furthermore, the subject of content and standardization should be treated. Similar treatment should be accorded to education in software (and hardware) systems.

It is planned to organize the meeting as a sequence of open plenary sessions with the entire group meeting to discuss the partial results obtained in one of the above areas; and in working sessions divided into working technical groups. A tentative schedule for the two major work groups (Content -- working Group A and Resources -- working Group B) follows:

Morning		Afternoon
	Monday	
Introduction		Work
	Tuesday	
Work		Work
	Wednesday	
Report A → B		Report B → A
	Thursday	
Work		Draft
	Friday	
Final Reading		

There are a large number of questions that the conference should attempt to answer. Among them are:

- Of the reasonably large number of graduate departments of computer science now existing, are these programs producing in kind and in number the graduates that are needed?
- Are there needs, insofar as computer science is concerned, which these programs are not meeting?
- Are these programs separating the mathematical from the engineering too much?
- What alternatives to this mode of educational development can be proposed?

- Does there exist a natural education sequence in the field of computer science like that, e.g., in another mathematical science? Thus, how does one characterize education in computer science through the range of junior college, B.S., B.A., M.S., M.A., Ph.D., and professional degree?
- In the field of computer science what are the goals of the various degrees?
- Is the education program best organized so that students from the lower degree programs provide the major source of the students in the advanced degree programs?
- Will computer science departments become as introverted as has happened, for example, in mathematics?
- How do the programs now in operation compare with those outlined by the study groups such as the ACM Curriculum Committee and COSINE?
- Are the professional societies the appropriate groups to recommend or set curricula? What orderly alternatives are there?
- Are there large problems in software production and use that are largely caused by the lack of well trained software specialists?
- If there are such large problems, should they be solved within a formal education system by educating specialists at various degree levels?
- Or can this matter be best solved by those now responsible for the production of software using on-the-job training?
- Thus, can hardware manufacturers be depended upon to supply the software systems that are needed and also train the personnel to produce and service them?
- Would not software education in a university environment produce technological derelicts since the software problem seems to change so rapidly?
- Put another way, won't the very nature of software make the solutions to these problems be solved by meta software produced by a very small number of specialists?
- If one speaks of software engineering, then why not let the engineering schools and disciplines define and develop the programs?
- Is it possible to meaningfully separate the software problem from the hardware problem?
- How can national institutes of computer science, several of which are now being proposed, contribute to education in computer science?

Other questions will arise during the course of the discussions, but certainly the goal of the conference should be to focus not only on the nature of the problem but to prepare recommended solutions. Naturally, any additional questions that you feel should be discussed will be considered. We would appreciate any feeling you may have concerning the priorities of the various topics which have been raised.

Though it is not required for participation, the attendees would be pleased to receive from you any written comments that you might care to make prior to the meeting. While formal papers are not being asked for, careful organization of your thoughts on these or other related matters would be appreciated. If a working paper can be provided by June 22nd copies will be made available to all the participants to study before the meeting commences. These working papers will undoubtedly provide a strong basis for discussion during the conference.

It is hoped that this conference will provide a reference for the field of computer science -- at least in the two major areas -- that will be a natural first source for information about the field. The conference will be attempting to obtain in one week what the more established sciences have developed over many years -- an overview of the present state, logistics, and future directions of the field. Naturally it could not hope to be complete, but it will provide a first overview of the field that up to now has not existed.

During the conference, duplication and secretarial facilities will be provided for quick preparation of additional working papers and intermediate reports. The goal of the conference will be the preparation of a report outlining the results of the conference. Toward that end, in each of the two areas (resources and content), a chairman and two younger recording secretaries will have the responsibility of preparing the draft of each section, and these two reports will then be coordinated into a final report.

You may be familiar with a report of the National Academy of Science entitled "The Mathematical Sciences: A Report (NAS publication 1681:1968, xiv + 256 pages, paper, $6.00). This report, and preceding reports by the Pierce Committee and the Rosser Committee are the sole widely based surveys conducted under federal auspices on computer science education. It is hoped that the report of this conference will provide a major technical expansion of the requiements and goals of computer science education.

Please let me know as soon as possible, and in no case later than June 9th, if you will participate in this conference. Upon receipt of your willingness to participate in the conference you will be receiving a set of preliminary documents on or about June 15th. These documents will include the full list of attendees, copies of the above mentioned report of the National Academy and the Pierce Committee, a report of the ACM Curriculum Committee, and working papers as they become available. A partial list of attendees and the groups to which we have tentatively assigned them is attached. I would appreciate additional names of people whose presence would materially improve the conference.

Sincerely yours,

Alan J. Perlis (dg)

Dr. Alan J. Perlis, Head
Department of Computer Science
Carnegie-Mellon University

AJP:dg

enc.

II. ORGANIZATION AND APPROACH OF THE COMPUTER SCIENCE AND ENGINEERING BOARD

Introduction

In view of the rapid evolution of the field of computer science and engineering, the National Academy of Sciences has decided to establish a Computer Science and Engineering Board comprised of a distinguished group of experts in the field of computer and information science and related areas. The Board will be available to provide advice to federal agencies and to other organizations which may have problems in which the Board can be helpful. This step is in keeping with the official role of the National Academy of Sciences to provide advisory assistance to the federal government in matters of science and engineering.

Since the field of computer science and technology is developing rapidly, the Board will have a special and continuing obligation to keep itself well informed. It should be capable of perceiving the current state and the future prospects of computer science and engineering, and of its professional practices in order to advise the government concerning the intellectual capital and the manpower resources necessary to insure continuing U. S. leadership in the field. It should be able to evaluate in technical terms the true meaning of the enormous and somewhat heterogeneous growth of information processing technology as it affects the public and private sectors of the nation. It should, in general, be capable of assessing the implications of advances in this branch of science and technology for the national welfare.

The Board should therefore take a broad view of this subject and of its applications to research and education in other branches of science and engineering as well as to the workaday needs of government, commerce, industry and education. Consequently, it should interact with other boards or committees under the various subdivisions of the Academy.

The Organization of the Board

This view of the Board's broad role implies a need to set priorities among areas of potential interest by weighing the importance attached to these areas.

The following recommendations on organization and priorities reflect the thought the Planning Group and its guests (Annex A) have given to these questions.

To function with a balanced and broadly representative group of individuals without losing the working efficiency of smaller groups, the Planning Group recommends that the Board organize itself into several committees, each subsuming panels created to meet specific needs.

Between plenary sessions of the Board, the committees would meet on schedules tailored to the work of the panels or working groups under their wing. These panels or working groups should be created as needed, often on a temporary basis. They should be chaired by a member of the parent committee and staffed for appropriate competence and breadth of representation by members of committees other than the parent committee and also by the most competent individuals in the nation representing significant points of view whether or not they belong to any committee of the Board.

Specific capabilities the Board should have at its inception were studied by panels of the Planning Group. The initial areas spelled out by these panels can be covered by starting the Board with the following three committees:

1. Education
2. Research and Development
3. National Programs

The interests and responsibilities of these three committees clearly overlap. The committees should therefore have overlapping membership. This mechanism for insuring balanced coverage of all significant points of view can be supplemented by the creation of joint panels to deal with specific subjects. The staffing and the mission of such panels would be determined by recommendations of the affected committees to the chairman of the Board, who would be responsible for assuring broad and balanced representation. Since competence and partiality often go hand in hand, broad and balanced representation should be interpreted as assurance of full and free expression of contending professional points of view.

Committees of the Board

The Committee on Education should be prepared to advise on educational questions, for example how to overcome the prevalent shortage of personnel in computer science and engineering. This committee very likely will need a panel on data-gathering to make recommendations about adequate statistics for describing manpower needs.

This committee should perform for education in computer science and engineering in a continuing, comprehensive and nationally representative fashion the role that the earlier committees chaired by Rosser (in NAS) and Pierce (in OST) (Annex B) could perform only for a limited time under restrictive charters.

The Research and Development Committee should be concerned with assessing the current state of the art and perceiving future directions for research and development. Three principal panels recommended for initial creation under the Research and Development Committee would study (a) the application of computers, (b) the science of machines and programs, (c) systems directions.

The first panel may advise on research policy leading to better applications methodology for extending current computer applications and for developing new application areas.

The second panel may advise on the development of a formal theoretical foundation for the developing science of machines and programs.

The panel on systems directions may foster the development of new systems concepts and organizations. The systems problems continue to be of the most difficult type, heightening the importance to be attached to great improvements in the depth of understanding and of skills for tackling the wide variety of such problems which confront all levels of organization, both government and private. Panels concerned with specific functional areas, e.g., data retrieval, can be formed in cooperation with the Committee on National Programs.

Under the Committee on National Programs, panels dealing with specific requests by governmental organizations would be formed as needed.

The Committee on National Programs should perceive and assess developments in computer science and engineering that affect national programs providing direct support to policy formulation and policy execution. It should advise on how human, equipment, and methodological resources may be combined to maximize the effectiveness and efficiency of federal, state and local governmental organizations.

The implications of the current state and future prospects of computer science and engineering on the formulation of government policy affecting computer science and engineering and related fields should also be a prime concern of this Committee.

Membership of Committees

The initial organization of the Board into three major committees leads to natural emphases on membership for the three corresponding areas. For example, the Education Committee should include people representing the universities, primary and secondary schools, the professional societies, and such business organizations or government agencies as are concerned with education and training. Members outside the computer science field, per se, should be included to assure satisfactory representation of other significant points of view.

The Research and Development Committee should include the individuals most knowledgeable in affected substantive areas without regard for the institutional character of their primary affiliation.

The National Programs Committee should include among its members people chosen primarily for their familiarity with relevant aspects of national civilian or military programs as well as experts in computer science and engineering.

Liaison groups should be established to inform other organizations within the Academy of the discussions and plans of the Computer Science and Engineering Board and to keep the Board informed of the needs of computer users in various areas of science and technology. As the need arises, more formal joint panels can be created in conjunction with other boards or committees.

DRAFT 2

DRAFT PROPOSAL ON A STUDY OF "Computer Research and Utilization in Universities and Colleges" conducted by the Computer Science and Engineering Board of the National Academy of Sciences

DRAFT
THE INFORMATION CONTAINED IN THIS DOCUMENT IS IN A PRELIMINARY FORM AND MAY BE SUBJECT TO ERROR, OMISSION OR AMBIGUITY.

(Draft prepared by W.F. Miller, 5/7/69)

CONTENTS

I. The Charge

A succession of reports have addressed themselves to various aspects of the needs and uses of computers in universities and colleges. The first of these, the Rosser Report, "Digital Computer Needs in Universities and Colleges", Publication No. 1233, National Academy of Sciences, 1967, addressed itself to particular needs and uses of computing in universities as well as the history of both within the universities. The second report was the Pierce report entitled, "Computers in Higher Education", Report of the President's Science Advisory Committee, The White House, Washington D.C., February 1967. This second report addressed itself to the computation facilities for universities, the use of computers in teaching and the educational needs of colleges and universities. The third report, the COSRIMS report, National Academy of Sciences, 1968, addressed itself to needs for support of research in the mathematical sciences. This report made a special appeal for increased support in the area of research for computer science. These three reports have been very helpful in guiding national policy in a very general way. There is a great need now for a report more directed toward the style considerations, man power considerations, and organizational and financial considerations for the research and teaching programs as well as for the institutional service programs in the universities. None of the previous reports addressed itself to the institutional service programs, that is, the use of computation in the administrative areas, the libraries, student records, and so forth. There are a number of important questions to be answered on the basis of current investigations. All of these are connected with how to better utilize available resources. Should colleges and universities find small, de-centralized computation centers, should they join in regional networks, should they join in big brother relationships, what are the factors that will contribute to the success of any one of these kinds of programs? There is a variety of experience now available to draw on in each of these areas,and a national study that could provide guidelines for government policy and for guidance of the universities and colleges would be of immense importance at this time.

II. Plan of Attack

This proposal is for a 12-month study into the needs and opportunities of universities, colleges, and junior colleges in the area of computers fo (1) their educational programs, (2) their research programs, and (3) their institutional services (administrative, etc.) programs. The proposal is not intended to carry out research in these areas, but is intended to accumulate and interpret information that is now available or may become available.

This study would address itself to such questions as:

1. What segment of the educational programs are receiving the most attention in colleges and universities, and what segments are receiving relatively little attention?
2. What will be the impact of the deficiencies uncovered above?
3. What are the experience factors of the colleges and universities in terms of the amount of computer time or money needed per student per unit of instruction for various types of courses, what kind of faculty attention is required, what kind of manpower and computer systems are available to provide these services?
4. What factors would contribute most to the success of a regional network shared by a number of colleges and universities? What factors would contribute most to the utilization of small, independent computers?

5. What are the needs and current plans of universities and colleges in the institutional service programs, that is the administrative data processing, libraries, etc.? What cost data is available on these programs, what threshold has to be obtained for the success of these programs? What other factors might contribute to the success or failure of institutional service programs involving use of computers?

The study group would plan to utilize the information that is being accumulated at a number of universities engaging in their own self-study as well as the information accumulating at regional centers and a number of other institutions that have achieved success with one style or another of computer utilization. It would also look into what factors contributed to the failure of certain styles of utilization in institutions where this is known to have occurred.

III. Budget

The budget is for a project from August 1, 1969 to July 31, 1970.

DRAFT
THE INFORMATION CONTAINED IN THIS DOCUMENT IS IN A PRELIMINARY FORM AND MAY BE SUBJECT TO ERROR, OMISSION OR AMBIGUITY.

BUDGET

Direct Costs

1.	Project Head	
	Full-time one month August 1969 Full-time one month July 1970	$ 5,000
	No charge for remaining 10 months	
2.	Executive Director	
	Full-time 14 months	26,000
	Overhead and benefits	?
3.	Three student assistants at 1/2-time, 3 months each (or 1/4-time for 6 months)	
	One on Teaching Requirements	1,000
	One on Research Requirements	1,000
	One on Institutional Service Programs	1,000
4.	One student assistant 1/2-time for 6 months	2,000
5.	One secretary full-time for 12 months	6,600
6.	Materials and Services (including telephone)	4,000
7.	Travel	5,000
	Subtotal	$ 51,600
	Academy Expenses - Overhead (?)	+ (?)
	TOTAL	

DRAFT 2

May 7, 1969

DRAFT Report of NSF Survey Panel
W. F. Miller, Chairman

Introduction: The Charge

The panel was charged with the investigation of patterns of support from the computer industry to the colleges and universities of the country. The panel undertook the survey of a few companies in the computing industry and a number of the officers of colleges and universities. Our approach was to see on the basis of a quick sample whether we could identify any changing patterns of support and whether it was necessary and/or useful to go into a second phase. The companies and universities sampled and interviewed are listed in the appendix with the written replies from their representatives.

Academic Discounts

One of the forms of support to colleges and universities that has been most prevalent until recently has been the academic discount (or educational allowance, as it is sometimes called) for computing equipment. The usual form of such support was a discount by the manufacturer for either the purchase or the rental of equipment. There have been some restrictions on the utilization of the equipment so acquired but the form of these restrictions has also changed over the years.

Before 1962 the IBM educational allowance agreement prohibited the use of the discounted machine for "sponsored research". Sponsored research here referred to work done by faculty and/or students on a federal government contract or grant. In 1962, IBM changed the nature of this restriction

DRAFT 2

DRAFT PROPOSAL ON A STUDY OF "Computer Research and Utilization in Universities and Colleges" conducted by the Computer Science and Engineering Board of the National Academy of Sciences

(Draft prepared by W.F. Miller, 5/7/69)

DRAFT
THE INFORMATION CONTAINED IN THIS DOCUMENT IS IN A PRELIMINARY FORM AND MAY BE SUBJECT TO ERROR, OMISSION OR AMBIGUITY.

CONTENTS

DRAFT
THE INFORMATION CONTAINED IN THIS DOCUMENT IS IN A PRELIMINARY FORM AND MAY BE SUBJECT TO ERROR, OMISSION OR AMBIGUITY.

I. The Charge

A succession of reports have addressed themselves to various aspects of the needs and uses of computers in universities and colleges. The first of these, the Rosser Report, "Digital Computer Needs in Universities and Colleges", Publication No. 1233, National Academy of Sciences, 1967, addressed itself to particular needs and uses of computing in universities as well as the history of both within the universities. The second report was the Pierce report entitled, "Computers in Higher Education", Report of the President's Science Advisory Committee, The White House, Washington D.C., February 1967. This second report addressed itself to the computation facilities for universities, the use of computers in teaching and the educational needs of colleges and universities. The third report, the COSRIMS report, National Academy of Sciences, 1968, addressed itself to needs for support of research in the mathematical sciences. This report made a special appeal for increased support in the area of research for computer science. These three reports have been very helpful in guiding national policy in a very general way. There is a great need now for a report more directed toward the style considerations, man power considerations, and organizational and financial considerations for the research and teaching programs as well as for the institutional service programs in the universities. None of the previous reports addressed itself to the institutional service programs, that is, the use of computation in the administrative areas, the libraries, student records, and so forth. There are a number of important questions to be answered on the basis of current investigations. All of these are connected with how to better utilize available resources. Should colleges and universities find small, de-centralized computation centers, should they join in regional networks, should they join in big brother relationships, what are the factors that will contribute to the success of any one of these kinds of programs? There is a variety of experience now available to draw on in each of these areas,and a national study that could provide guidelines for government policy and for guidance of the universities and colleges would be of immense importance at this time.

II. Plan of Attack

This proposal is for a 12-month study into the needs and opportunities of universities, colleges, and junior colleges in the area of computers fo (1) their educational programs, (2) their research programs, and (3) their institutional services (administrative, etc.) programs. The proposal is not intended to carry out research in these areas, but is intended to accumulate and interpret information that is now available or may become available.

This study would address itself to such questions as:

1. What segment of the educational programs are receiving the most attention in colleges and universities, and what segments are receiving relatively little attention?
2. What will be the impact of the deficiencies uncovered above?
3. What are the experience factors of the colleges and universities in terms of the amount of computer time or money needed per student per unit of instruction for various types of courses, what kind of faculty attention is required, what kind of manpower and computer systems are available to provide these services?
4. What factors would contribute most to the success of a regional network shared by a number of colleges and universities? What factors would contribute most to the utilization of small, independent computers?

5. What are the needs and current plans of universities and colleges in the institutional service programs, that is the administrative data processing, libraries, etc.? What cost data is available on these programs, what threshold has to be obtained for the success of these programs? What other factors might contribute to the success or failure of institutional service programs involving use of computers?

The study group would plan to utilize the information that is being accumulated at a number of universities engaging in their own self-study as well as the information accumulating at regional centers and a number of other institutions that have achieved success with one style or another of computer utilization. It would also look into what factors contributed to the failure of certain styles of utilization in institutions where this is known to have occurred.

III. Budget

The budget is for a project from August 1, 1969 to July 31, 1970.

BUDGET

Direct Costs

1.	Project Head	
	Full-time one month August 1969 } Full-time one month July 1970 }	$ 5,000
	No charge for remaining 10 months	
2.	Executive Director	
	Full-time 14 months	26,000
	Overhead and benefits	?
3.	Three student assistants at 1/2-time, 3 months each (or 1/4-time for 6 months)	
	One on Teaching Requirements	1,000
	One on Research Requirements	1,000
	One on Institutional Service Programs	1,000
4.	One student assistant 1/2-time for 6 months	2,000
5.	One secretary full-time for 12 months	6,600
6.	Materials and Services (including telephone)	4,000
7.	Travel	5,000
	Subtotal	$ 51,600
	Academy Expenses - Overhead (?)	+ (?)
	TOTAL	

to prohibit classified research or research not done as a part of the academic mission of the university or college. Their decision to change was based on the idea that they could not police source of funds but could better judge on other criteria such as openness and the association with faculty and students.

A second restriction imposed is if the equipment is resold within a five-year interval after purchase, the educational institution must rebate to the manufacturer a pro-rated amount of the discount.

The amount of discount made available to the colleges and universities has been decreasing over the last several years. There are a number of forces clearly moving in the direction of the elimination of this form of support to colleges and universities. In the mid-1950's the discount was often as high as 60 percent; that is, the college or university would pay 40 percent of the listed price of equipment.[1] This discount would apply either to the purchase of equipment and subsequently to the equipment maintenance contract, or to the rental (including maintenance). In the case of the rental contracts it was common for the university or college to pay 40 percent of the first shift rental and be permitted to utilize the equipment on as many other shifts as possible with no additional charge. Discounts have been decreasing[2] in percentage until currently they are about 20 percent average over the whole line of equipment for IBM and either about 20 percent, or in many cases nothing,[3] from other manufacturers.

---

1. Reference will be to a specific contract still being identified.

2. G.S.A. reference (1966)

3. Letter from James G. Miles, Vice President, Control Data Corporation, to W. F. Miller, Stanford University, 13 March 1967.

In the opinion of the panelists and the representatives of academic institutions surveyed, the academic discount was a very important form of support in the early years. It contributed immensely to the growth of the computing industry in the country. The computing industry grew in its most spectacular growth "from the ground up". When the colleges and universities began to graduate engineers, scientists, business school graduates, etc., who had been introduced to computing through introductory courses (and often had taken advanced courses in computing), they began to introduce computer methods into their respective businesses. This in turn stimulated the great demand for computers and the spectacular growth of the computer industry in the early and mid-1960's. There is no doubt that the colleges and universities who first introduced large teaching programs in computing would not have been able to support these educational courses on such an extensive scale without the benefit of the academic discount.

Before the so-called Carnegie decision[4] the colleges and universities were able to treat the academic discount as a gift and utilize that contribution solely for support of their educational and unsponsored research programs. This practice was eventually disallowed. Also academic discounts began to decrease in percentage contribution. Colleges and universities now have to look to other sources of support for their computing equipment to carry out their educational programs.

It is quite clear to the panel that this form of support will soon be very small or completely eliminated. Control Data Corporation[3] has

---

4. Carnegie Institute of Technology (1964) ASBCA No. 4299, 1964 BCA 4026. Credits against computer rental - A non-profit institution contractor using an IBM 650 computer for sponsored research could not include the full rental for the computer as a research cost under a cost-reimbursement contract since it was allowed a 60-percent deduction in rental payments for a so-called educational contribution regardless of whether or not the prerequisite to the taking of the deduction was fulfilled.

completely eliminated the academic discount. It does support research at the colleges and universities in areas of interest and/or unusual merit. The IBM Corporation[5] has indicated that their tendency is toward unrestricted grants of a general type. In the interview with Dr. Spinrad of Scientific Data Systems he made it clear that the academic discount was utilized only when necessary to keep them competitive and that they followed the lead of the larger companies in this area.

There is an additional force that will very likely contribute to the vanishing academic discount. In the anti-trust suit of the U. S. Government against the IBM Corporation,[6] the IBM Corporation is charged with the utilization of the academic discount as a means of affecting a monopolistic position. It is clear that the recommendation will be to enjoin IBM to cease and desist the offering of the academic discount. In the civil suit of the Control Data Corporation against the IBM Corporation,[7] CDC also charges IBM with damaging them through use of special pricing mechanisms to control the market. These pressures will certainly encourage IBM in the direction of the elimination of the academic discount whether or not the Control Data Corporation and the Justice Department suits are successful. It is clear from the letter of Dr. Piore that IBM is tending in that direction anyway.

---

5. Letter from E. R. Piore, Vice President, IBM Corporation, to A. G. Oettinger, Harvard University, 19 February 1969.

6. Civil Action No. 69 CIV.200, U. S. District Court for the Southern District of New York, Filed: January 17, 1969. See COMPLAINT § 20(d) and PRAYER § 4.

7. Civil Action No. 3-68-312, Filed December 11, 1968, in the District Court of the United States for the District of Minnesota Third Division. COMPLAINT § 23(f) PRAYER FOR RELIEF § (2).

Other Support

Aside from the area of the academic discount, the trend for support of research and teaching seems to be taking two different turns. IBM on the one hand is tending to turn toward a general university support and in the form of funds that may be used at the discretion of the president of the university and may not necessarily be directed toward computer research or computer education. Control Data Corporation and Scientific Data Systems on the other hand are emphasizing support of relatively specific research projects that might be aimed at advancing the capabilities and techniques of the computer industry. These two tendencies are leaving a widening gap in the area of general educational support of the universities and colleges. These institutions are having to turn to other sources of funds, both internal and external, for their teaching and general educational programs. The support of Scientific Data Systems and Control Data Corporation[8] is normally aimed at those facilities which have acquired their company's machines. In any case, there seems to be no indication that there are very large amounts offered in support of research although we are unable to get precise quantitative data.

---

8. "Practice and Procedure for Sponsored Research", Control Data Corporation, Minneapolis, Minnesota, March 22, 1968.

Draft of letters to be sent to educational institutions.

Dear Mr.

The Computer Science and Engineering Board of the National Academy of Sciences is conducting a census to assess the impact of industrial support on computer-related activities in educational institutions. This study is being carried out under a contract from the National Science Foundation. We believe that the results of this study will be invaluable to the Board in its deliberations and recommendations concerning support for computers and computer science.

We are initially interested in determining the internal and external factors which impact the nature and effectiveness of industrial support. We would like to inquire:

1. In what forms do you now receive industrial support for computing from equipment manufacturers, software companies, or user companies such as banks, oil companies, and so forth? By forms of support we would include equipment discounts, unrestricted grants, value received research contracts, or other.

2. Can you fully take advantage of this support or are there auditing or government research administration policies that are detrimental to this end?

3. Do you have any policies within your own institution that restrict the form in which you can receive industrial support?

We should like to set up an informal interview between the appropriate person in your institution and Professor W. F. Miller of Stanford University who is chairman of the Board panel that is conducting this study.

Would you kindly let me know at your earliest convenience the person to whom we may speak on the topic.

Respectfully,

Anthony G. Oettinger
Chairman, Computer Science and
Engineering Board

Mr. Lyman Spitzer, Chairman
University Research Policy Committee
Princeton University
Princeton, New Jersey 08540

Professor James G. Brophy
Vice President for Academic Affairs
Illinois Institute of Technology
Chicago, Illinois 60616

Mr. W. F. Miller
Associate Provost for Computing
Stanford University
Stanford, California 94305

Professor A. G. Norman
Vice President for Research
University of Michigan
Ann Arbor, Michigan 48104

This is a draft of a letter to be sent to the manufacturers and software houses for the NSF study on patterns of industrial support. There will be one each for IBM, Control Data Corporation, Scientific Data Systems, and UNIVAC. The addressees are listed below.

Dear Mr.

The Computer Science and Engineering Board of the National Academy of Sciences is conducting a census to assess the impact of the industrial support of computer-related activities for our educational institutions. The study is being carried out under a contract from the National Science Foundation. We believe that this information will be of great importance to the Computer Science and Engineering Board in enabling it to make its recommendations on national programs.

We are principally concerned with the internal and external factors which contribute to policy of the industry. In particular, we should like to determine:

1. What needs in the educational institutions does your company believe it is meeting?

2. What direct or indirect returns do you expect for your company or for the computer industry in such areas as manpower training, research and development, or sales?

3. What facets of federal government policy such as taxation, research support, or research administration influence the type or level of industrial support?

We should like to set up an informal interview between the appropriate officer of your company and Professor W. F. Miller of Stanford University who is chairman of the Board panel that is conducting this study.

Would you kindly let me know at your earliest convenience the person to whom we may speak on the topic.

Respectfully,

Anthony G. Oettinger
Chairman, Computer Science and
Engineering Board

Dr. E. R. Piore
Vice President and Chief Scientist
IBM Corporation
Armonk, New York 10504

Mr. Max Palevsky, President
Scientific Data Systems
1649 Seventeenth Avenue
Santa Monica, California

Mr. R. McDonald, President
UNIVAC
Box 8100
Philadelphia, Pennsylvania

Mr. William Norris, President
Control Data Corporation
8100 34th Avenue South
Minneapolis, Minnesota

Mr. Fletcher Jones, President
Computer Sciences Corporation
1901 Building, Suite 1900
Century City, Los Angeles 90067

March 4, 1969
W. F. Miller

Summary of Interview with Dr. Robert Spinrad
Vice-President, Programming
Scientific Data Systems

1. SDS does not make grants to universities or colleges.

2. Academic Discounts are on the basis of field experience. SDS views universities and colleges as a source of business (like any other source of business). Field experience means that SDS follows the lead of larger companies such as IBM and CDC.

3. Research and Development Contracts to colleges and universities are mostly on a services rendered basis. Spinrad described this support as "enlightened self-interest". The R and D contract may not call for an immediate payoff, but SDS does not engage in very much (if any) speculative R and D.

4. SDS has a summer student program intended to introduce students to SDS and to computing research and development. It has as a secondary goal the support of students.

JAN 28 [illegible]

SPERRY RAND CORPORATION

UNIVAC DIVISION
EXECUTIVE OFFICES
P.O. BOX 8100, PHILADELPHIA, PA. 19101 • TEL. (215) 646-9000

January 24, 1969

Mr. Anthony G. Oettinger, Chairman
Computer Science & Engineering Board
Aiken Computation Lab.
Harvard University
Cambridge, Mass. 02138

Dear Mr. Oettinger:

Your letter of January 21, 1969 to Mr. McDonald has been turned over to Mr. Frank D. Sweeten, Vice President of Personnel, for response. Mr. Sweeten is currently out of the country and will not be back until February 3. As soon as he returns, your letter will be called to his attention.

Sincerely,

J. R. Stahl, Director
Employee Benefits

JRS:dmh

CC-F. D. Sweeten

JAN 30 Rec'd

ILLINOIS INSTITUTE OF TECHNOLOGY
CHICAGO 60616

OFFICE OF THE VICE PRESIDENT

January 28, 1969

Mr. Anthony G. Oettinger,
Chairman, Computer Science
& Engineering Board,
Aiken Computation Laboratory,
Harvard University,
Cambridge, Mass. 02138

Dear Mr. Oettinger:

A relatively small fraction of our financial support for IIT's computer related activities is derived from industrial sources, with the exception of educational allowance for equipment purchases. We will, however, be pleased to meet with Professor W. F. Miller to discuss our situation at his convenience. Prof. Miller should make arrangements for his visit with my office (312/225-9600, Ext. 521-522) for I feel he should meet with me as well as Professor P. G. Lykos, Director, IIT Computation Center.

We are most pleased to participate in this effort of the Computer Science and Engineering Board.

Very truly yours,

James J. Brophy
Academic Vice President

JJB/dla

cc: Professor P. G. Lykos

THE UNIVERSITY OF MICHIGAN
ANN ARBOR, MICHIGAN 48104

A. G. NORMAN
Vice-President for Research

January 28, 1969

Dr. Anthony G. Oettinger, Chairman
Computer Science & Engineering Board
Aiken Computation Laboratory
Harvard University
Cambridge, Massachusetts 02138

Dear Dr. Oettinger:

In reply to your letter of January 21, we will, of course, cooperate in supplying your committee the information requested, though frankly we are becoming a little tired of responding to sub-contracted questionnaires from the National Science Foundation. You are, of course, aware of the very extensive one handled by the Southern Regional Education Board last year.

I believe that as far as the University of Michigan is concerned the answer to the specific questions you pose are:

(1) There is very little industrial support for computing, direct or indirect, other than that which may be present in setting leasing rates or purchase prices to educational establishments generally.

(2) & (3) There are no constraints that would inhibit acceptance of support

For more detailed information, I would suggest that Professor Miller get in touch with Dr. Robert Bartels, Director of the Computing Center (area 313) 764-2412.

Yours sincerely,

A. G. Norman

A. G. Norman

AGN/mg

cc: Dr. Robert Bartels

NAS

CORPORATE HEADQUARTERS
8100 34TH AVENUE SOUTH, MINNEAPOLIS, MINNESOTA 55440 / 612-888-5555

13 March 1969

CONTROL DATA
CORPORATION

Professor William F. Miller
Stanford University
Computer Sciences Department
Stanford, California 94305

Dear Bill:

It was a pleasure to talk with you this morning regarding the study that you are conducting for the National Academy of Sciences regarding the impact of industrial and financial support of computer-related activities for educational institutions (I refer to Anthony Oettinger's letter of January 21, 1969, to William C. Norris, President of CDC.).

I am enclosing two copies of CDC's PRACTICE AND PROCEDURE FOR SPONSORED RESEARCH (revised 11/22/68) that best states CDC's objectives, policies and procedures for sponsored research.

As I mentioned to you this morning, two years ago CDC changed its policy with respect to grants to universities and other non-profit research institutions from a policy of granting discounts in prices on computer systems to a policy where we will quote only full list prices on computers to education and research institutions, and at the same time consider the sponsoring of research programs by which CDC pays the qualifying institutions for research work to be done on programs of interest to CDC and/or which CDC believes have unusual merit. We have specifically concentrated in the past two years on grants re hospital/medical and CAI, as well as the development of specific new softwares and applications.

I believe this generally answers the question raised by Dr. Oettinger's letter.

I will look forward to seeing you at the time of your forthcoming trip to Minneapolis to view the 7600 computer and STAR. I would also appreciate the opportunity to schedule you to see some of our systems directed toward some of our business management data systems in line with Stanford University's interests.

Very truly yours,

CONTROL DATA CORPORATION

Jim Miles

James G. Miles
Vice President

JGM:fah
encls.

IBM International Business Machines Corporation

Armonk, New York 10504

Office of Vice President
and Chief Scientist

February 19, 1969

Professor Anthony G. Oettinger
Aiken Computation Laboratory
Harvard University
Cambridge, Massachusetts 02138

Dear Tony:

Your letter of January 21, on behalf of the Computer Science and Engineering Board of the National Academy of Sciences, inquires into the nature of IBM support of computer-related activities of educational institutions. First let me point out that IBM's educational support program is not restricted to computer-related activities, and an increasingly large portion of our support is, in fact, unrestricted. Nevertheless, I will attempt to provide meaningful answers to the questions asked in your letter:

1. What needs in the educational institutions does your company believe it is meeting?

IBM's program of support to educational institutions falls generally into the following categories:

Unrestricted Support:

Because unrestricted support is the most useful to a college president, IBM is tending toward more unrestricted grants. Such grants should be of assistance in helping the institutions to cope with their over-all financial problems, including those which may be associated with computer-related activities.

Special Program Support:

A good example of this type of grant is IBM's support of the Harvard University Program on Technology and Society.

While it is not directly computer-related, the effects of technological developments, which include the computer, are under study in this program.

Another example is a grant made to one university to assist in the development of an engineering design curriculum. Other examples would be support toward the development of a PhD program in computer science or toward the improvement of undergraduate mathematics teaching.

Equipment Education Allowances:

One traditional method of support is IBM's educational allowances, applying to a variety of equipment.

Graduate Fellowships:

IBM maintains a regular program of fellowship support to leading graduate schools. The selection of fellows is made by the institutions and their schools or departments.

Post-doctoral Fellowships:

IBM awards a small number of post-doctoral fellowships directly to institutions each year. In addition, some faculty members are provided the opportunity for post-doctoral research in IBM laboratories.

Visiting Professors:

IBM encourages professional personnel exchanges between faculty members and its professional employees. Several IBM scientists are engaged in full-time teaching and research on work assignments, and a large number contribute through part-time teaching.

Negro Educational Support:

IBM provides both unrestricted and program support to a number of historically Negro colleges. In addition, IBM supports several fellowship programs for black students in other institutions.

Contract Support:

At any given time, IBM, through its divisions, sponsors specific research tasks through contractual relationships and joint studies. The scope of this activity ranges from applied technology to software development.

2. What direct or indirect returns do you expect for your company or for the computer industry in such areas as manpower training, research and development, or sales?

Since most of IBM's financial support is in the form of unrestricted grants or program support aimed at specific institutional needs, any returns we would receive would be very indirect and not easy to measure. The benefits accrue more to the institutions than to us, although obviously we, as others, are dependent upon the output of colleges and universities in terms of educated manpower and basic and applied research. In cases of specific research sponsorship, however, in the category described above as "contract support," IBM anticipates a direct return commensurate with our investment.

3. What facets of federal government policy such as taxation, research support, or research administration influence the type or level of industrial support?

We have been unable to identify any federal government policies relating to taxation, research support, or research administration which have any specific influence on the type or level of IBM support to educational institutions.

We are delighted that this study is being made. We hope that its results will encourage broader support on the part of all segments of industry not only for computer-related activities but for higher education generally.

Sincerely,

Manne

E. R. Piore
Vice President and Chief Scientist

ERP:mk

For reproduction + distribution to Board at May meeting - for final discussion of this draft + follow-on actions.

170
85
255

COMPUTERS AS MANAGERIAL TOOLS
IN THE FEDERAL GOVERNMENT

~~It seems that~~ The Nixon Administration has ~~may~~ convened a new "Hoover Commission" to review managerial practices in the federal government. This could offer an excellent opportunity to evaluate the role and capabilities of computers as managerial tools in the federal government.

Such a review would be justified on several grounds. Many feel that the computer has not been used as effectively as a managerial tool within the public as in the private sector. This is of course merely a suspicion, and formal documentation of the point is certainly lacking. Nevertheless, a good deal of informed opinion tends to this view.

This is not to assert that computers are not used in the federal establishment. The federal government is the single largest customer of the computer industry. It has also shown foresight and a commendable willingness to bear risks in developing new and better computers.

The question, rather, concerns the use made of the computer within the government as an aid to management and particularly whether that use is as effective as it should be. Above all, the question arises of whether the "information revolution" induced by the computer has really permeated public managerial procedures in the way that it should if the computer is to make a maximum contribution to improving government management and operating practices.

The computer should not be envisaged as simply a device for reducing data processing costs or the total clerical or computational bill. Computer installations very often do not create the "savings" that "justified" or otherwise rationalized the original purchase of the computer. However, even when expected "savings" do not materialize, the computer application usually does result in some jobs being done more thoroughly or better or in entirely new tasks being undertaken. In particular, management, after replacing clerical help with computers, often steps up demands for information or for the speed with which information is prepared for management.

Often, nonrealization of "expected cost savings" is taken as a sign of failure. It is highly probable that computers have often been applied where they should not have been. Nevertheless, the fact that the total clerical bill does not always decline when a computer is installed is hardly significant as evidence of failure: it is sometimes merely a sign that the computer was bought on improper grounds.

There are good economic reasons why one would not necessarily expect the total clerical bill to decline with the advent of computers. The same arguments apply to other major technological innovations. The "total transportation bill" of most modern societies has not declined with the advent of the railroad and automobile. It is higher today, and life is different in many ways, some presumably more productive and better. Much the same applies to the computer. It would be impossible for some sectors of our complex society and economy to operate in their present manner without the availability of large computers.

The phenomenon observed in these cases of major technological change is described by economists as the case in which price elasticity of demand for a service or product is greater than unity. This means that the total consumption of the good or service rises more than proportionally to any price reduction. If you observe a 10% reduction in the unit price or cost of rendering a particular service, a greater than unity price elasticity would imply that you would observe more than a 10% increase in the quantity consumed. There are many goods or services for which greater than unity price elasticities seem to hold. If you make a good thing cheap enough, you should not be surprised if, in the aggregate, people spend more rather than less on it as its price is reduced.

The essence of the "computer revolution" is that once an initial investment has been made, information becomes relatively cheap and, as a consequence, larger amounts of information are consumed or used. The increase in consumption of information may well rise to the point where the total bill for processing or obtaining information actually goes up even though the unit cost of obtaining that information falls dramatically. From the standpoint of managerial practice, the real question then becomes just exactly what does the increase in availability and quality of information really imply?

The direct effects are reasonably obvious. Management demands and receives quantitative information more promptly than before. Usually more and sometimes better analyses of data are brought to bear upon individual management decisions. Management feels freer to go back to subordinates and to ask that additional numbers be processed or additional facts be obtained before reaching decisions in marginal or difficult cases. Simply put, demands are made for analyses that could not have been conceived or justified before the computer. As a consequence, fewer decisions are made on the basis of hunch or so-called "rules-of-thumb", or other simple short cuts that tend to reduce the information requirement. Justifiably or not, greater emphasis than before is put on the quantifiable factors in decision-making.

The indirect effects upon management of the computer information revolution may, however, be more important than the direct effects. Indeed, it seems highly likely that we do not even yet comprehend all of these indirect influences.

Perhaps the most obvious of these secondary effects concerns personnel policy. When the computer comes on the scene, the personnel requirements for an organization shift in important ways. Fewer clerical people performing rote tasks are needed, and those that are required need different skills, such as keypunching. More people with mathematical and analytical skills will be sought for employment. Such highly talented people do not easily fit into existing patterns of pay and responsibility. Necessary changes in recruitment policy can have subtle effects upon the organization. It not only changes the quantities of different talents or skills available for promotion, but also tends to influence those particular talents singled out for more responsibility. Similarly, to the extent that adoption of new technologies attracts the more venturesome and imaginative elements in the work force, the organization that accelerates computerization will find itself with a better quality or at least a more flexible and innovative work force, at least for awhile.

Whatever the particulars, it is clear that there are many ways in which personnel policies and organizational development can be and are being altered with the advent of the computer. Questions concerning public administration arise from all of this. Do civil service rules and other rigidities that characterize employment relationships in government inhibit the best use of computers in government? Does governmental observance of rather strict seniority rules tend to insure that government administrators realize fewer managerial advantages from computerization than would otherwise be the case?

It is often observed that many sectors of the so-called service industry, and in particular medicine and government, have lagged badly in terms of productivity gains compared with other sectors of the economy. Yet the computer is particularly adaptable to service industries. The natural question is:

have various kinds of government record keeping or bookkeeping activities (and that after all constitutes a fairly considerable proportion of total government activity) experienced the same gains in productivity as equivalent record keeping or bookkeeping activities in the private sector?

If government hiring and personnel policies have not or are not able to adapt in the same way as those in the private sector to the advent of the computer or other technological changes, this may induce personnel selection policies that are self-reinforcing and negative. For example, some think that the government's inability to recruit its "fair percentage" of the more imaginative and innovative spirits in recent college generations is a chronic and worsening condition. The long-run tenor and success of government operations may be better measured by the Post Office than by the Apollo program, and the evidence is disquieting.

If there is any merit in the notion that the computer has not made the same contribution to public as to private administration, and that the costs of this omission can accumulate over time, this should be known to responsible men in the government and to the public as well. While it could be argued that inefficiency can create jobs, it can be a quite expensive means of doing so. One wonders whether "deliberate or allowed inefficiency" is ever a very constructive approach to alleviating unemployment. What we all need to know is whether there is remedial inefficiency. Then we can choose whether or not to remedy it.

Some other indirect effects have a revolutionary impact. Some of our major textile firms, for example, are using linear programming to control the entire operation of their fabric production and merchandising through the astute application of linear programming techniques. They control production down to the individual loom in plants which are widely dispersed geographically, Likewise, the sales mix and the marketing effort are largely determined by considerations of profitability relative to various capacities which are analyzed by linear programming techniques. Clearly, here is practice which would not be possible without computers and which ten or fifteen years ago would hardly have been within the vaguest understanding of top management.

On the other hand, another secondary or indirect effect of the use of computers in management might be termed "the rapture of the measureable". Contemporary management is fashionable seeking to become more "scientific" in its methodology or practice. This thrust causes a preoccupation with forcing the decision-making process to become algorithmic or formula-like; and this in turn depends on management's ability to quantify or measure input data relevant to a decision and to quantify or measure the results or probable results of alternatives. The availability of computers tends to facilitate such formalized decision-making since computers make possible data and algorithm testing on a vastly more efficient and practical basis than before.

The rub comes when -- because all sorts of data are available -- we begin to factor into our decision making only those factors which are measurable and to judge the results of our decisions only by the measurable factors.

Many decisions -- perhaps even most decisions -- involve factors which are not now measurable. While the computer can help in these decisions, judgment, style, and inspiration are ingredients the computer cannot handle. A management in its rapture with the computer may overlook this limitation and ossify rather than mechanize its practices.

The depth of consequences of mechanization of management practices is not well-understood. Machine systems in fact may tend to diminish the range of managerial and administrative choices in many situations, especially those in which any interpretation of the rules comes into play. A chosen alternative will, of course, most likely be one provided within the framework of the system, which, in turn, can be expected to take its own side in any question of interpretation. Likely pragmatic result: over-simplication of any situation not anticipated by the system designers.

Perhaps the vast middle of the managerial hierarchy is of greater concern here than are the extremes. Top management is generally entitled to augment and even to contravene the system in order to arrive at sound decisions, while at the supervisory level and below there are always people who are expected to beat the system when necessary to get the job done (though the civil service is a bit deficient in this regard).

Middle-management, on the other hand, has seldom been characterized by either imagination or enterprise, and to reduce its freedom of choice to a level even below its present, rather modest level of improvisation might well accelerate hardening of the civil arteries. It would be unfortunate indeed if we unwittingly lost what few middle-management flexibilities do exist in managerial and administrative procedures through algorithmic zeal.

This concern is scarcely a new one, of course, and there is always the hope that rapid, organized access to great masses of information will jazz up even middle-management. But we all know that system designers are poor, finite beings of well-bounded prescience and that the Machine tends to dazzle and dominate.

Finally, computer systems themselves, like the ways of using them, are still in a state of rapid flux. Until now, we have seen mainly "single-user" systems. Obviously, in the case of, say, the Navy procurement program or Standard Oil management information system, these are very large systems, with management at various levels having access to appropriate sets of procedures and data, and, one hopes, being prevented from accessing data inappropriate to the level of the inquirer. From now on, however, some government systems are likely to go in the direction of multiple-user systems, with several agencies (federal, state and local) having access to a system. The value and by-products of such systems may no longer be measurable solely by the criteria used for the earlier single-user systems. In particular, the degree to which an agency may maintain control and self-confidence may be as important a measure as the integration and streamlining achieved.

In short, an important task of any new "Hoover Commission" should be to consider the extent to which government administrative practices hinder or aid adaptation to and use of effective new managerial technologies, broadly construed.

HOOVER COMMISSION STATEMENT

John Griffith Version

a current ~~The~~ trend is ~~toward~~ the use of greater amounts of capital equipment in many areas of industry and government. In the administrative areas of government alone, the amount of capital equipment is increasing at a ________% rate per year and ______% of this figure is data processing equipment. (ask Ann Lamb to fill in blanks)

Indent more — Even though the federal government is the largest single customer of the computer industry, there are many areas in which the computer is not being used as effectively as is possible. If a "Hoover" commission is appointed to review managerial practices in the federal government, this seems like an excellent opportunity to evaluate the role and capabilities of computers as administrative or managerial tools in the federal establishment.

A seemingly important question that arises is how can capital equipment be better brought to bear on the administrative and managerial problems of the federal government? This question is actually part of a deeper question: how can major technological innovations be introduced into this establishment? The emphasis here is on the innovative characteristics of the problem rather than on the aspects of utilization (such as effectiveness).

The essence of the "computer revolution" is that, once the investment in capital equipment has been made, information comes relatively cheap and as a consequence larger amounts of information are consumed or used. The increase in consumption of information may well rise to the point where the total bill for processing or obtaining information actually goes up even though the unit cost of obtaining that information falls dramatically. From the standpoint of managerial practice, an important question then becomes just exactly what does the increase in availability and quality of information really imply?

This question is best answered by analogy. A computer can play a better game of checkers than anyone but a master. Who taught it to do this? Not a master checker player. ~~checker master~~. It was done by a programmer who instructed a machine to vary and improve its decision procedures while ~~by~~ playing against experts. What is significant here is the machine was not instructed by a person who was deeply knowledgeable about checkers and who painstakingly worked out endless possible moves and responses. Success, in this case, was ~~is~~ a result of adaptive behavior in automatic equipment. Much more effort will be required to generalize from the simple task of playing checkers to more complex types of "moves" and more subtle measures of success. The machine modifies its action ~~This exercise forces us to concentrate on~~ decision procedures rather than on ~~of~~ moves. ~~It takes the programmer a little bit further from the subject matter on which the machine is operating.~~

When this type of system is utilized by managerial personnel, it will tend to remove them from a specific understanding of the phenomena on which the computer is acting, and ~~although~~ it will demand of them ~~him~~ a higher understanding in terms of procedures for dealing with generalized situations and phenomana.

Hence, the implication of a vast increase in quantity and quality of information is that the federal government managerial establishment ~~in the federal government~~ will be able to concentrate on the <u>methods</u> of decision making, rather than on the subject matter itself. Since the system will be able to adapt itself to the user, there should be less apprehension in regard to technological innovation.

At present, the efforts of managers both in and out of the government tend toward one-of-a-kind solutions to problems. This is often time because of a tendency to emphasize the characteristics of the solution instead of the characteristics of the problem.

A principal task of a "Hoover" commission is to delineate a plan for introducing technological innovation into the administrative and managerial areas of the federal government. There are many problems associated with the introduction of automatic equipment which exhibits adaptive behavior.

The problem of bias is a major concern. As these systems are used and adapt to the situations with which they are presented, the machine may introduce bias into the decisions it makes. As managers rely increasingly on automatic equipment these biases may become harder to detect because the managers become increasingly further removed from the subject with which he deals. As the activities become more complex the manager faces with increasing seriousness the problem of undetected errors which may introduce subtle influences into the results he derives from computers.

The very mystique of these operations may lead to an overreliance. This overreliance may be especially distressful when we are attacking problems which are not otherwise accessible to analysis. As we move the human intellectual problem farther from the subject matter itself, we lose the reliance of the user on his intuition and we must substitute great care, formality, and rigor. Alternatively, we must develop new levels of intuition and judgment in our managerial personnel.

Hence the "Hoover" commission must plan not only for the ~~installation~~ introduction of capital equipment but for the upgrading of personnel who must develop an intuition which is capable of reaching through the automatic equipment to a continuing understanding of the administrative science which ~~lias~~ lies beneath it.

NATIONAL ACADEMY OF SCIENCES

PROPOSED STUDY FOR

FEDERAL COMMUNICATIONS COMMISSION

OF

COMMON CARRIER/USER INTERCONNECTION

STUDY FACTORS

SERVICES:	VOICE, DATA, COMMON USER COMMUNICATION
PARTIES:	PRIVATE COM, TEL COMPONENT MANUFACTURER, GOVERNMENT, DOD, DATA SERVICES AND MANUFACTURER, METER READ
PROBLEM:	IMMEDIATE NET CONTROL, QUALITY, PROTECTION, METER READ NEAR- AND FAR-TERM TRENDS AND RECS
AREAS:	TECHNICAL, DEMAND, COSTS, LEGAL

OBJECTIVES

NEAR-TERM TRENDS

- TECHNOLOGY
- REQUIREMENTS
- IDENTIFY PROBLEM AREAS AND POSSIBLE SOLUTION

IMMEDIATE PROBLEM

- INFORMAL CONFERENCES - USERS/TELCO
- RESOLUTION OF DIFFERENCES
- TARIFF RECOMMENDATION

FAR-TERM TRENDS

- DEVELOP FRAMEWORK FOR FUTURE CONFIGURATION

APPROACH

HIGH LEVEL COMMITTEE

- TECHNICAL
- LEGAL/ECONOMIC

SUBCOMMITTEES

- NEAR- AND FAR-TERM TRENDS
  - DATA SERVICES
  - VOICE SERVICES
  - COMMON CARRIER
- IMMEDIATE PROBLEMS [FCC]
  - INTERCONNECTION

MONTHLY OVERALL AND SUBCOMMITTEE MEET
TO COORDINATE AND RESOLVE

ORGANIZATION AND PROBLEMS

LARGE GROUP (APPROXIMATELY 10 MEMBERS), FLEXIBLE

COMMUNICATORS, DATA USERS, PRIVATE COMMUNICATION NETWORKS, LEGAL/ECONOMIC, INDUSTRY, GOVERNMENT, NOT-FOR-PROFITS, UNIVERSITIES

HIGHLY INTERESTED

CONFLICT OF INTEREST

INFLUENTIAL IN COMMUNITY

AVAILABILITY OF BACK-UP SERVICES

SPREAD WORK LOAD

MUCH CONTROVERSY

MINORITY REPORTS

CONFIDENTIALITY OF INFORMATION

BUDGET

	DAYS	TRIPS
MAIN PANEL + SUBPANEL 10 MEMBERS		
12 MEETINGS/YEAR 2 DAYS/MEETING	240	120
INFORMAL CONFERENCES 3 MEMBERS		
4 TASK AREAS, 5 MEETING/AREA, 1 DAY/MEETING	60	60
	300	180

COSTS

TRAVEL $150/TRIP 180 TRIPS	$27,000
EXPENSES $25/DAY 300 DAYS	7,500
CONSULTANTS	
2 STUDIES/TASK AREA, $3K EACH	24,000
	$58,500

NEXT STEPS

APPROVAL	
BOARD	
NAS	
FCC	END OF MAY
RECRUITING	
LETTERS TO ORGANIZATIONS AND INDIVIDUALS	
DECISIONS	
SEND INFORMATION OUT	EARLY JUNE
ORGANIZATION MEETING	
INFORMAL CONFERENCE NOTICE	MID- TO-LATE-JUNE
INFORMAL CONFERENCES	MID- TO LATE-JULY

The Commonwealth of Massachusetts
University of Massachusetts
Amherst 01002

GRADUATE SCHOOL
Office of the Dean of the Graduate School
and Coordinator of Research

Munson Hall

APR 15 REC'D 9:00 70

Olsen

April 13, 1970

Mr. J. F. Kettler
National Academy of Sciences
Computer Science and Engineering Board
2101 Constitution Avenue
Washington, D. C. 20418

Dear Mr. Kettler:

I return herewith those pages of the report, Computer Science Education, on which I have made changes. On the whole, I feel that Project Salvage has done a fine job.

There is one point that warrants special attention. The recommendations go to great length to support the idea of Computer Science degrees at all levels, and yet the very extensive review of Computer Science at the University of Waterloo (and I recommend that most of this be eliminated) suggests that Computer Science be an option in Mathematics. These are not critical disparities, but perhaps somewhere on pg. 3 a sentence should be included to the effect that opinions differ as to the most appropriate path for developing the computer scientist.

Sincerely yours,

Samuel Seely

Samuel Seely
Associate Graduate Dean

SS/s

PREFACE

In July 1969, a group of approximately thirty experts from industry, government, and education met in Annapolis, Maryland to discuss ~~the future of~~ certain aspects of education in Computer Science. The chairman of this conference was Alan J. Perlis. In March 1970, a smaller group met in Washington D.C. to prepare this report which presents the findings and recommendations of the original conference. Final editing of the report was done by Thomas H. Bredt.

1. INTRODUCTION

Scope of the Conference

A conference to study Computer Science Education in the United States was held in Annapolis, Maryland, in July, 1969. The conference was sponsored by the National Academy of Sciences' Computer Science and Engineering Board under a grant from the National Science Foundation. The conferees attempted to identify both the proper goals for a college or university education in Computer Science and, in a broad sense, the routes and structures for achieving those goals. Discussion and recommendations focused on the estimated 20% of the computer field personnel who, even in the near future, must be college trained. ~~However,~~ the bulk of the people working in computing for the next 5 to 10 years will not be four-year college graduates but will, rather, receive their final training in high schools or two-year colleges. This meeting did not address the critical problem of training such people. A conference on this ~~subject~~ should be organized.

A Profession or a Science?

A recurrent question that threaded its way through most of our discussions was whether "Computer Science" ought to be a "professional" or "scientific" study. The issue, which in the end proved to be somewhat vacuous, was whether the graduate was to go on to "design things" (like the engineer) or "illuminate truth" (like the mathematician). The conclusion of the group was that ~~a variety of~~ graduates ~~is~~ necessary and should be produced, but that the distinction in their education should be achieved by the extent and, therefore, the depth and richness of their education ~~in~~ Computer Science and not by "separate tracking" or education in different disciplines. Computer Science, therefore, is a discipline which has both practitioners and scholars.

The Numbers Problem

The first issue addressed was the numbers problem: How many and what kind of people do we need to educate? Two approaches were used in attacking ~~this~~ problem:

(1) extrapolation of equipment-support requirements.

(2) reasoning by analogy with other fields.

The first approach assumed the existence of about 10,000 computers in the United States in the 1975-1980 period (these are machines that need the support of computer professionals). A further assumption was that the number of computers and the staff needed to support them would "plateau" thereafter. (as ~~This provided~~ a base for the manpower computation.)

With due consideration for the "mix" of large, medium and small installations, a support group of about 600,000 professionals was deemed necessary. The long term, steady-state condition (about 30 years from now) assumed that these people will all be college trained. Then, assuming a working life of 30 years, the replacement rate will be about 20,000 per year.

The current college-educated "professional" population in the computer industry is believed to be about 100,000. This number is ~~so~~ far below that needed to competently staff the nation's computer installations and ~~that~~ even a 30,000 per year influx of trained people would be desirable. Such a "production" rate, of course, is not currently possible. This number is presented only to support our contention that a 20,000 per year rate is a reasonable national goal.

The second approach to the "numbers game" adopted was to estimate the support population in relation to other, better understood and more mature, disciplines. Compared to the 40,000 engineering graduates per year (from all engineering disciplines) our chosen target of 20,000 per year seems reasonable.

Compared to the 10,000 medical doctors per year and the 20,000 nurses per year, the anticipated rate retains its plausibility.

We, of course, understand the fatuity of attempting to make these kind of predictions for periods beyond the next five years. Our chief concern is that we are not overstating the need; We believe we are not.

The Mix of Graduates

Influenced by estimating that Twenty thousand graduates per year is the goal. What kind of training should they have? What is the "mix" of degrees? These were the next issues addressed.

Many factors influenced the assessment of the proper proportions. In net, it is the judgment that the 20,000 per year should be broken down as follows:

500	PhD's/year
3,500	Master's/year
16,000	Bachelor's/year

The Educational Path

As mentioned above, Computer Science is seen as a single, coherent academic discipline in the same sense as other broad ranging professional areas. We ~~reject the notion~~ adopt the position that "theoretical" Computer Science and "practical" Computer Science are ~~so different, that they cannot~~ founded on a common study and should share the same base. For that reason we recommend that there be, in any university, a single basic Computer Science "track". An undergraduate "core" curriculum (with electives from a number of related and user-fields) will produce a bachelor-level graduate who has a thorough grounding in the fundamentals of his field. Graduate study will lead, inevitably, to deeper understanding and greater accomplishment.

Single-tracking the Computer Science student creates a number of serious problems as to the content of the core curriculum. We understand the complexity of the issues raised and do not, here, propose or recommend any particular curriculum. We are convinced, however, that the benefit of not having a splintered discipline far outweighs the disadvantages induced by the necessities of accomodation.

The Neglected Majority

An important though paradoxical result of our deliberations is what we might call the plight of the neglected majority. At the present time, about 80 percent of the computing done in the United States is in support of business applications. (Programming in COBOL), business data processing systems design, development, and operation represent the sole activity of the majority of the people in the computer field today. These kinds of activities, similarly, will represent the major involvement of people during the projection period.

Why not, then, direct our educational policies and courses towards the more specialized needs of this group? The arguments come, in essence, to the notion that the data processing community benefits from students educated in the fundamentals of Computer Science.

Costs of Educational Computing in Computer Science

(Except for the cost of providing computer services), the cost of educating the Computer Science student is not significantly different from that of educating the Physics or Chemisty student (~~again without~~ laboratory expenses). ~~However, an important,~~ but costly part of the student's training is the provision of experience with a computer (or with computer services). This necessary laboratory experience is estimated to cost:

$ 20 per man per year	for the non-Computer Science student
$1,000 per man per year	for the Computer Science BS years
$2,000 per man per year	for the Computer Science MS years
$4,000 per man per year	for the Computer Science PhD years

This represents a cost of about $90 million per year for Computer Science education.

2. RECOMMENDATIONS

1. At the present time there exists a recognized shortage of professionals trained in Computer Science. This shortage is felt in all areas where computers are used. To remedy this shortage, at a fairly expert level, we recommend the establishment of strong master's programs in Computer Science in degree granting institutions.

By a strong master's program we mean a program that will provide a sufficient education for those professionals who are going to fill the need for trained practitioners of Computer Science in industry and government, and who ~~will~~ improve the efficiency and scope of computer operations.

Furthermore, we recommend that master's programs in Computer Science contain strong elements of laboratory training in the development and utilization of computer systems.

2. To remedy the ~~above mentioned~~ shortage at a less expert level, we recommend the establishment of strong bachelor's programs in Computer Science in degree granting institutions.

By a strong bachelor's program, we mean a program which will prepare students for employment as working computer professionals, and for advanced education in ~~either or both~~ master's and PhD programs in Computer Science.

Furthermore, we recommend that bachelor's programs in Computer Science contain strong elements of laboratory training in the development and utilization of computer systems.

3. We recommend that the development of doctoral programs in Computer Science continue at its present rate. While these programs have ~~done~~ well, it is recommended that they continue to be supported by (1) graduate teaching and research fellowships, (2) post-doctoral teaching fellowships ~~to~~ aid in acquisition of new faculty, and (3) support of new and different computer facilities. Examples of these new and different facilities are satellite computers, processors for film and TV animation for instructional purposes, hybrid computers, converters to and from other systems, and advanced equipment ~~such~~ as is developed ~~as a result of~~ the investment of national resources in research and development programs for defense, space and other sciences.

4. It will be essential for the universities and colleges to greatly expand the~~ir students~~' opportunities to learn the essentials and principles of all elements from problem formulation to computing realization ~~and for~~ these institutions ~~to be~~ aware of the part that Computer Science can play in this expansion. It is recommended that ~~support be provided to implement~~ cooperation between computer scientists and individuals of other departments, ~~to the end that~~ individuals from other departments be encouraged and supported in providing opportunities for students to gain insight and knowledge in part or all of Computer Science. All reasonable efforts should be made to encourage interdepartmental cooperation in this area. Finally, both research in the general area of application and materials preparation directed toward teaching deserves support, especially when each is planned to support the other.

5. It is recognized that the need for professionals in Computer Science is a national one and, therefore, all effort should be made to ~~provide~~ encourage ~~support for~~ the development of bachelor's and master's programs with the widest possible geographical distribution.

6. In the rapidly changing field of Computer Science and computer related activities, up-to-date information on research is needed, ~~and~~ but it is hard to get. Under NSF sponsorship, the Southern Regional Education Board has prepared surveys of college and university educational activity in the computing sciences, but ~~apparently~~ there appears to be no agency that is supporting a similar effort ~~is doing anything similar~~ for research in this field. At the same time, graduate departments ~~have a great~~ suffer from a lack of ~~need for, but possess very little~~ information on what research in computing sciences is being sponsored; who does the research, who sponsors it, and at what levels.

Even in a relatively old and stable field like mathematics, a strong need has been felt for up-to-date information about the nature of education and research in the field, and the amounts and sources of its funding. These needs resulted in the NSF-sponsored Survey of Research Potential and Training in the Mathematical Sciences (c. 1957), and the reports of the Ford Foundation-sponsored Survey Committee of the Conference Board of the Mathematical Sciences (c. 1967). The latter committee apparently will maintain a continuous inventory from now on.

We recommend that similar support be provided for a continuing research and manpower committee whose mission would be to maintain a continuing national inventory of research activity and manpower needs in Computer Science.

7. In the current environment, there exists a large and growing number of highly trained and competent PhD's from related fields. There is evidence that Many of these people would like to redirect their talents to Computer Science. We, therefore, recommend the development of special programs that are directed to ~~that special attention be paid to~~ the invaluable opportunity that exists for creating applications programmers, systems programmers, and Computer Science faculty and research persons of ~~by retreading~~ recent PhD's in other fields in order to meet). In particular, we recommend the institution of transdoctural programs that will provide for the appropriate ~~complete the~~ training of such people in one to two years.

area of design and development of large computer programs for such large systems that there appears to be a lack of organized instruction in higher education, here or anywhere, at the present time.

<u>Systems Laboratories</u>. We consider the laboratory-experimental aspect of the training of students in Computer Science to be vital to their development. We therefore believe that ~~the establishment of~~ computer systems laboratories experience is an important part of the curriculum of both undergraduates and graduates in Computer Science.

There are many substitute plans that could conceivably serve to fulfill the same purpose as the computer systems research laboratories, e.g., summer employment in industry, cooperative work projects with industry, or part-time employment in a computation center on campus.

We believe that a team of six students can be given a very significant experience for $1,000.00 per student or $6,000.00 for the whole team for a one-quarter laboratory.

<u>The Master's Degree Program</u>. The nation has need for people to do a variety of jobs connected with computers. A substantial number of these people will be involved with the design and implementation of large computer systems each consisting of an assembledge of equipment (hardware) and a complementary collection of systems and library programs (software).* Those people involved in such design and implementation activities are carrying on a profession which is in a very real sense similar to that pursued by professional engineers. It is our belief that professional educational programs

---

* time-sharing systems, traffic control systems, command and control systems, management information systems, etc.

should be available which are specifically planned to provide the knowledge necessary to carry out these computer system design activities. By analogy with the engineering situation, it seems clear that these educational programs should be at the graduate level, leading to a master's degree; that they should build on a relevant bachelor's level education; they should be specifically planned as terminal, professional master's programs; and that they should consist of courses at the level of scientific generality offered to beginning doctoral candidates, i.e., should not be vocational type courses. Initially, the bachelor's level education of those entering this master's program will probably consist of a degree in engineering, physics, mathematics, etc. with a minor in Computer Science. As the number of Computer Science baccalaureates increases, a larger proportion of the students entering the master's program will have a deeper preparation in Computer Science that will bring about improvements in the quality of the program.

It is recognized that there may be a master's program in other academic areas which accent computers and their applications. It is felt that these programs should be designed by and largely be manned from within these academic areas. The Computer Science faculty should be used to teach what are appropriately the Computer Science courses included in these applied programs.

For the Master of Science program, a figure of 5 million dollars per year in hardware costs was obtained.

The total cost in hardware is 29 million dollars per year. One of the figures that we used was that the EDP industry would be taking in about 100,000 people per year. What percentage of these should be PhD's? Figuring that one percent should be PhD's we get a desirability of producing a thousand PhD's a year. Our feeling on the matter was that by 1975 we might be able to produce 1000 PhD's in Computer Science, but that we would not be able to produce 1000 PhD's per year by 1975. If you can get up to about 300 by 1975 this would be about what we could expect. It seems to double about every two years.

From whence comes this figure of 15,000 BS students per year? Is it attainable? At the present time in engineering and mathematics the output per year is of the order of 50,000. Now assuming there is no major change in size of total undergraduate enrollment in engineering and science schools but that quality Computer Science undergraduate programs do come into being, how many of the 50,000 per year could we expect to prefer an education in Computer Science? We believe that without a great deal of heavy advertising or pressure of any sort, 20-30% of the undergraduate enrollment in mathematics and engineering programs would shift into Computer Science programs, if there were existing quality undergraduate programs in Computer Science. Furthermore, the percentage is probably conservative. That means that of the 100,000 per year that are required in the EDP area, 85,000 are probably going to have to remain or be non-Computer Science baccalaureates. We also made an estimate of Computer

Computer Science at the University of Waterloo * (J.W. Graham)

History and Philosophy. Computer Science courses have been taught at the University of Waterloo since the academic year 1959-60. They were actually taught before the university installed its first computer. It was not until 1964 that a program in Computer Science and a philosophy of operation of that program became evident in any formal sense.

At the present it is felt that ~~although~~ Computer Science is starting to ~~get as~~ develop a coherent body of knowledge, but it is still not at the stage where it can be taught as a separate and complete undergraduate discipline, ~~and be considered as~~ that is, it has not yet developed a basic body of knowledge, such as is the case in mathematics. It is felt by the Faculty at Waterloo and by many others in the field [4] that

---

* Numbers in square brackets, e.g. [6] , refer to references found at the end of this subsection.

analysis and programming courses and courses in allied areas such as probability and statistics and logic. Any curriculum that was developed for the co-operative program was applicable to the regular program and vice-versa.

The honours mathematics program with Computer Science option has now been established on both a regular and co-operative basis since 1964, although it is still under development and will continue to change in order to remain current. The present program is based on a solid foundation of mathematics with the optional courses primarily in later years. There are three ways a student may pursue an option in Computer Science. He may enroll in the Co-operative Honours Mathematics Program, the Regular Honours Mathematics Program or the Regular General Mathematics Program. In an honours program a student attends university for four years and completes 17 or 18 mathematics courses (including Computer Science) and 9 elective courses. The difference between the regular and co-operative program has been explained previously. In a general program he attends university for three years and completes 9 mathematics courses and 7 elective courses. The elective courses may be chosen from the sciences, humanities or engineering and the student may study such courses as philosophy, psychology, economics, chemistry, physics, etc.

A description of the undergraduate program on a year-by-year basis is presented next. Where a choice of Mathematics courses is indicated some typical examples are shown; elective courses are indicated where taken, but no course titles will be given.

There is no real need for course descriptions — I recommend that pp 41–52 be eliminated

Beely