DOCUMENT RESUME

ED 184 873 SE 030 534

AUTHOR Bellman, Richard E.; And Others
TITLE Project SOUL: Computer Training Program for High School Students from Disadvantaged Areas. Part I, General Report. Technical Report.
INSTITUTION University of Southern California, Los Angeles. Dept. of Electrical Engineering.
SPONS AGENCY National Science Foundation, Washington, D.C.
REPORT NO USCEE-402-A
PUB DATE Apr 71
GRANT NSF-GJ-0981
NOTE 112p.; For related documents, see SE 030 535 and ED 180 765. Appendix K (Publicity - Newspaper Clippings) removed due to copyright restrictions.

EDRS PRICE MF01/PC05 Plus Postage.
DESCRIPTORS Compensatory Education; *Computer Oriented Programs; Computers; *Computer Science Education; Curriculum Development; *Disadvantaged Youth; Educationally Disadvantaged; Experimental Programs; *Program Descriptions; Programing; *Secondary Education; Secondary School Mathematics; Secondary School Science; Summer Programs
IDENTIFIERS Project SOUL

ABSTRACT

This report presents a general description of the 1970 "Project SOUL" summer program. The computer training course, intended for disadvantaged high school students, provides both training for immediate jobs and fundamental mathematical and scientific training for students going on to college. Included in this document are a description of the selection of participants, the data processing program, the scientific programing course, and a project overview. (MK)

April 1971 USCEE 402A

UNIVERSITY OF SOUTHERN CALIFORNIA

Technical Report

PROJECT SOUL:

Computer Training Program for High School Students from Disadvantaged Areas

PART I - General Report

Richard E. Bellman
Carlos Ford-Livene
M. Virginia Zoitl

Supported by the National Science Foundation
under Grant No. GJ-0981

ELECTRONIC SCIENCES LABORATORY

USC

Engineering

April 1971 USCEE 402A

Technical Report

PROJECT SOUL:

Computer Training Program for High School Students from Disadvantaged Areas

PART I - General Report

Richard E. Bellman

Departments of Mathematics, Electrical Engineering, and Medicine

Carlos Ford-Livene

Departments of Mathematics and Electrical Engineering

M. Virginia Zoitl

Department of Electrical Engineering

University of Southern California

Supported by the National Science Foundation under Grant No. GJ-0981

PART I CONTENTS

APPENDICES

INTRODUCTION

In the contemporary world, education is essential for economic survival, particularly technical education. Yet a considerable part of our population is denied any opportunity for this education by accident of birth. Millions of children grow up in areas of the city where school facilities are both inadequate and overcrowded and where the teaching staff is overworked and underprepared.

This de facto segregation of the poor, black, brown, and white poor, must be eliminated if we want a democratic and prosperous America. With the best of will and resources it will require considerable time and effort to bring the schools in deprived areas up to the levels of those in more prosperous areas, say in the Los Angeles area to have the schools in Watts and Venice on par with those in Brentwood and Pacific Palisades.

While the general effort is being made, there are certain special measures we can take to accelerate certain parts of the program. We can help the high school graduate looking for a job and the high school student hoping to go to college. Although it is obviously impossible in a summer program to repair the deficiencies of twelve years of underprivileged education, we can hope to provide some vital training in a few crucial areas.

One of these areas is the use of digital computers. In an age of automation the average person has a choice of learning how to use the machine, or of being displaced by it. Fortunately, we are at the beginning of this age, at a time when it is not too difficult to teach the high school student with the rudiments of mathematics the fundamental properties of the contemporary digital computer.

Once this training has enabled him to get a job in American industry, the cultural deficiencies of his education can be remedied in various ways. Once this training has allowed him to compete in college in a number of important areas, mathematics, engineering, and science, the student can concentrate on the other areas that have been neglected in his schooling. Given confidence and motivation, the human mind has fantastic abilities to learn quickly.

In 1968, our program started with a volunteer staff from the USC Departments of Electrical Engineering and Mathematics. Funds for thirty student stipends were received one day before the high school year ended. We contacted six high schools by telephone, and obtained the names of 200 interested students. From these names, thirty students were randomly selected. The Neighborhood Youth Corps sent us an additional twenty students.

In 1969, we received support from the National Science Foundation, a grant of $27,400, Grant No. GJ 423, which enabled us to continue the program in a systematic fashion. This grant provided funds to train one hundred inner-city youths from seventeen high schools. Eighty science-major students were selected to receive instruction in computer programming, while twenty non-science majors were chosen to learn data processing skills.

In view of the success of these programs, we expanded our efforts in 1970. We enlarged the program to accommodate as many underprivileged youths as funds, facilities, and personnel allowed.

On the basis of our experience, we feel strongly that a program of this nature can, and should, be carried out in every medium size city that contains a college or university. We hope that our program will serve as a prototype for similar programs throughout the country.

This training program has the following advantages:

1. It provides training for immediate jobs.
2. It provides fundamental mathematical and scientific programming training for students going on to college.
3. It makes good use of college and university facilities that are not used to capacity during the summer.
4. It furnishes valuable teaching experience to graduate students and research assistants.
5. It allows these young people to obtain the satisfaction of using their intelligence and training for social and idealistic purposes.
6. It provides a method of financial support during the summer for graduate students and research assistants in the area of computers and mathematics.
7. It provides an important example of the way that the college and university, business interests, and the community can work together as a team for the benefit of each and all.

In the following pages we present a general description of the 1970 program. A more detailed description of the courses taught, as well as techniques used, may be found in the supplementary reports by James Woolever and Art Lew, curriculum coordinators of the business data processing and scientific programming courses, respectively.

SELECTION OF PARTICIPANTS

Letters of "invitation to participate" and copies of the report on the 1969 program were sent by Richard Bellman to the principals of eighteen high schools in the general area around the University of Southern California. Members of the Project Soul staff met soon thereafter with a representative from each school to further explain the purposes of the project, answer questions regarding it, and set guidelines to be followed in selecting candidates.

Each school was requested to furnish the names and addresses of twenty-two candidates for the three courses, to be nominated on the following bases:

1. Keypunch Operations.
   Two seniors with non-scientific background who could type at least 40 words per minute.

2. Fundamentals of Business Data Processing and COBOL Programming.
   Five seniors with non-scientific background. This group was required to take a programming aptitude test.

3. Scientific Programming.
   Fifteen candidates (7 juniors and 8 seniors), each with a background in mathematics and science. At least one year of high school algebra with a grade of C or better was a prerequisite. This group was required to take the programming aptitude test. Students with more than 20 hours of previous training in FORTRAN IV were <u>not</u> eligible to participate.

An outstanding scholastic record was not necessarily a prerequisite for eligibility as a candidate for any phase of the program. In fact, we specifically requested that each school nominate two candidates for the scientific programming course who have demonstrated aptitude in mathematics and science, but were inconsistent in their performance, as reflected by their classroom records.

To assist in the selection of participants in the data processing and scientific programming courses, a programming aptitude test was administered by the Project Soul staff to over four hundred candidates, Saturday, June 6, 1970, on the campus of the University of Southern California. On that same date each candidate for the keypunch operations course was interviewed.

The candidiates were informed by mail Monday, June 8, 1970, regarding the results of their application. Successful applicants were allowed seven days to accept or decline an offer. All vacancies as of June 16 were filled by alternates. On June 19, class schedules and related information were mailed to each participant.

THE DATA PROCESSING PROGRAM

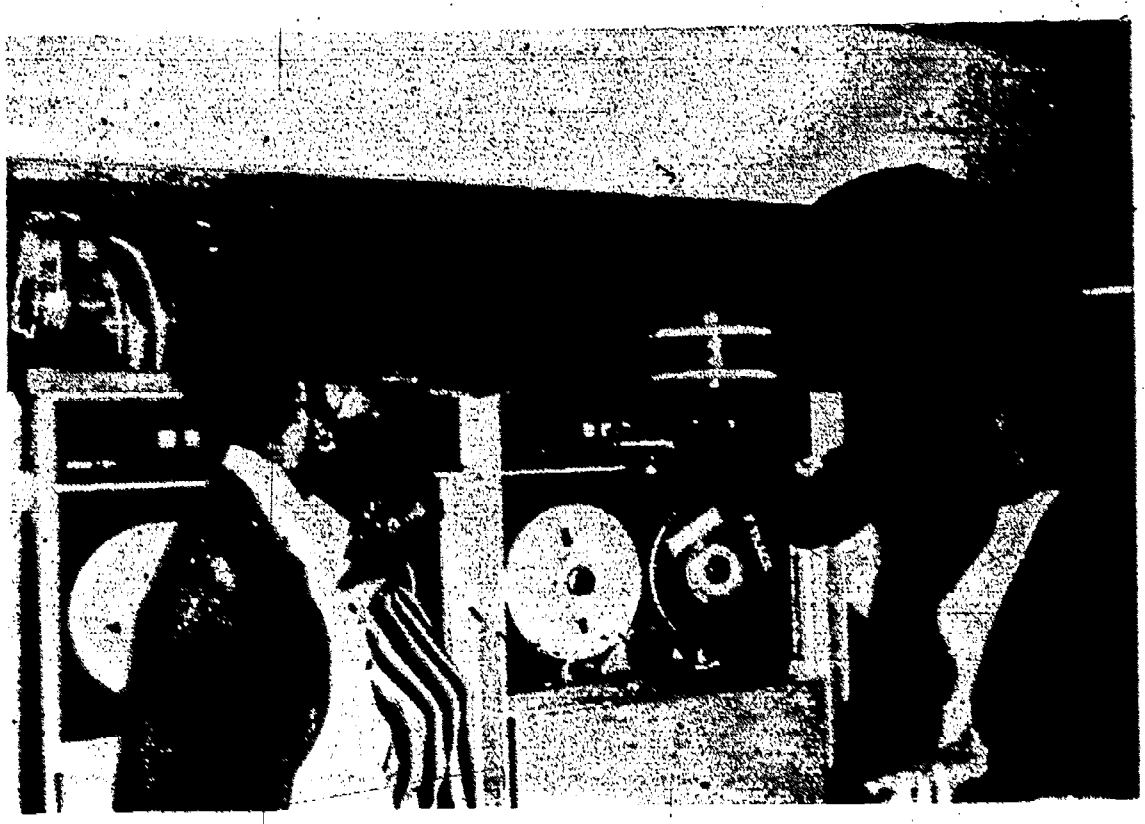
1. Introduction

A total of fifty high school juniors and seniors with non-scientific backgrounds were invited to participate in the Data Processing Program. In general terms, the primary goal was to provide these young persons sound basic training in commercial data processing. The training, practical in its form, was designed to expand the horizons of the students in the educational and occupational frames of reference and, thereby, to motivate their interest in continuing education. Secondary goals were to prepare them for employment in the areas of keypunch operations and data processing, and to establish and maintain a file of competent data processing personnel for recruitment by firms in the Los Angeles area which have indicated an interest in our trained people.

The keypunch course accommodated ten students and lasted four weeks. Two hours per day were devoted to lecture and two to laboratory work.

Below, students are shown receiving individual instructions from John Kelder, keypunch instructor.

Forty students in groups of twenty took part in the six-week business data processing course. The first two weeks were spent on the fundamentals of business data processing, and the remaining four were devoted to an intensive course in COBOL programming utilizing the Honeywell H-200 computer system. Again, two hours per day were spent in lecture and two in the laboratory; at least one of the latter involved hands-on experience with the hardware.



A hands-on training session.

A detailed description of the courses content as well as teaching methods and techniques used can be found in Part II, Report on the Data Processing Program, by James Woolever, curriculum coordinator for that segment of the 1970 Project Soul.

2. Section I - Keypunch Operation

a. Course Description and Objectives

This course has been designed to acquaint the student with the various processes of Key Punching in a business environment. A prerequisite was the ability to type 40 words per minute.

Upon successful completion of this four-week course, the student would be able to:

1. State from memory at least six functions which a punched hole in a card can perform.
2. Describe correctly a unit record card in terms of the number of columns, and the names of the various punch positions.
3. Define a card-field and calculate the number of columns required to punch a given field.
4. When questioned, locate and explain the use of:

Main Line Switch	Card Hopper
Punching Station	Reading Station
Card Stacker	Back Space Key
Keyboard	Chip Box
Program Control Unit	

5. Set up a program control card, using the automatic functions available with this machine.
6. Correct and manually duplicate error cards.
7. Define the following program codes:

   Automatic Duplication
   Automatic Skip
   Alpha Shift
   Field Definition

8. State the purpose of the following:

   Program Drum
   Column Indicator
   Program Card

9. Remove and replace a program card.

10. Correctly operate the card punch under program control.

b. Course Outline

I. Class Opening
II. Principles of Unit Record Accounting
III. The Unit Record Card
IV. Features of the Punched Card
V. Manual Punching Exercises
VI. Program Control Unit
VII. Additional Features of the Card Punch
VIII. Verifying Procedures
IX. Examination

Text: International Business Machines Reference Manual 024-026 Keypunch.

c. Assessment and Recommendations

With reference to achieving its immediate goal, the keypunch course was generally successful. After the first sixty hours of formal training the students were found to be very efficient in most phases of card punching. This suggests the feasibility of a three-week (sixty-hour) course. We, nevertheless, recommend that the fourth week be retained for the purpose of acquiring "on-the-job" experience on an internship basis. This experience can be acquired either on the college campus, where the course is taught, or in the local community. In either case the student's performance would be supervised and evaluated by the instructor of the course.

A typical keypunch laboratory session.

The successful performance of the keypunch class of ten students suggests that the program should be expanded to accommodate as many underprivileged youths as funds, facilities, and personnel will allow. We further recommend that keypunch courses be made available at the high schools across the nation on a similar basis as the long-standing courses in typing and stenography.

3. Section II - Fundamentals of Business Data Processing and COBOL Programming

a. Course Description

This course was divided into two parts, the first of which covered the fundamentals of Business Data Processing. In Part One the student learned the basic operation of the following unit record machines: IBM keypunch, IBM interpreter, and IBM card sorter.

The second part utilized the most frequently used procedural language for Business Application-COBOL. COBOL is a "Near-English" language which provides for thorough documentation of the program, and enables programmers to be able to program on all major manufacturer's equipment with a minimum of adjustment. Business application was analyzed, flowcharted, coded, and debugged using the computer.

Prerequisite for the course was satisfactory completion of programmer aptitude test.

b. Course Objectives

1. Introduce the students to the fundamentals of data processing and its application in business today.
2. Discuss principles of the recording function through the use of the keypunch machine.
3. Discuss the principles of sorting in both numeric and alphabetic sequence.
4. Introduce the student to the operation and control panel wiring of the card interpreter.
5. Discuss and use the current methods of computer problem flowcharting.
6. Develop the student's ability to work effectively with any modern third generation computer system.
7. Develop in the student confidence that he can work and communicate in any computer environment by applying the general principles of programming to the specific language that he may encounter.
8. Provide the challenge of applying and extending the student's ability in problem-solving situations.
9. Further extend the student's knowledge of current data processing techniques.

c. Course Outline

Part One

1. Class Opening
2. Fundamentals of Data Processing
3. The Unit Record Card
4. The Card Sorter
5. The Interpreter

Part Two

A. Electronic Data Processing and COBOL
   1. Data Characteristics and Organization
   2. What is a Procedure?
   3. What is COBOL?

B. The Procedure Division
   1. The Parts of a COBOL Source Program
   2. Basic Procedure Division Elements
   3. The Move, Add, Subtract, Multiply, and Divide Verbs
   4. The GO TO, perform and Stop Verb
   5. The IF Statement
   6. Input and Output

C. The Data Division
   1. The File Description
   2. Level Structure
   3. Record Description
   4. The Picture Clause
   5. Condition Name Value Entries
   6. The Working Storage Section
   7. The Constant Section

D. The Environment Division

E. The Identification Division

Text and References:

1. Text: Spitzbarth, Laurel M., Basic COBOL Programming, Menlo Park, California, Addison-Wesley Publishing Co.

2. References: International Business Machine:

026-024	Keypunch
029	Keypunch
557	Interpreter
083	Card Sorter

Methods of Instruction

1. Instructor explanation and demonstration of course content.
2. Use of overheads and computer listings to provide actual models of language and techniques discussed.
3. Assignments requiring students to write programs that utilize all aspects of course.
4. Class discussion on projects and assignments.

Methods of Evaluation

1. Regular attendance
2. Laboratory projects
3. Formal testing

d. Assessment and Recommendations

In general, the students fulfilled the basic requirements of the data processing course quite satisfactorily. Their intellectual curiosity and growth during the six weeks became quite evident as they became involved with solving increasingly complicated commercial problems via COBOL programming. In fact, their rapid progress and ever-increasing enthusiasm far exceeded expectations. Soon they were producing and submitting programs to be run on the Honeywell H-200 at a faster rate than the system we had established was prepared to handle. This posed a significant problem, since COBOL programs require a relatively large amount of compiling time. Consequently, the one-hour limit per day of computer time (part of which was to be devoted to hands-on training on the hardware)

was insufficient. We now know, and recommend hereby, that unless the system (we) used is suitably altered, no less than two hours daily of computer time should be allowed. Precise statements of justification for additional computer time for program check-out and for hands-on training are given below by data processing instructor Norman Gale. We trust the experience cited above and the statements of Mr. Gale will assist in convincing financial aid committees of the need for adequate funding of computer time.

Instructor Norman Gale explains a program to three of his students.

There are many reasons supporting hands-on training:

1. Hands-on training develops confidence and a psychological sense of mastery of the equipment.
2. Operation of the equipment helps the student to understand the system more thoroughly. Being present when the operator messages are displayed will help the student improve his ability to function under system control.
3. Hands-on training provides immediate reinforcement of correct behavior and discouragement of incorrect behavior--the student gets immediate results of his programming and operating efforts. Time lags between performance and evaluation in educational situations are crucial.
4. Mistakes that students will undoubtedly make in machine handling are better made while at school in a training situation than after completing the course when on-the-job mistakes can be very costly. Furthermore such mistakes, after completing the course, can reflect poorly on the program.
5. The time lag between execution and results is eliminated and this facilitates more effective learning.

Program check-out is a process of determining whether or not the program just written will produce correct results. Usually, program check-out consists of two steps:

1. The first step, known as desk-checking, consists of reviewing the program starting with the problem definition, checking the sequence for logical errors using the flow chart and coding sheet.

2. The second step of the program check-out involves testing the program on the computer itself. Thus, on the programmer's first attempt to translate the program into machine language, more likely than not, a list of errors will be presented to the programmer.

The computer motivates excellently classroom discussions and program writing of their assigned case studies. The students, enthusiastic and motivated, express often the desire for more computer time. The students, some with three or more programs wait two days for a single compiling of their program. The lack of compiling time results in less time not only for writing new programs, but also increasing the difficulty of debugging current programs as a result of the time lag.

Because of the overall success of the data processing course this year and of the previous year, we feel the course has proven its value. Therefore, we again recommend an expansion of the program to accommodate as many underprivileged youths as funds, facilities and personnel will allow. We further recommend that similar programs be made available across the nation, wherever there are students who, because of inadequate facilities and lack of community concern, find themselves at an increasing disadvantage. We note that comparable programs are presently made available to students at the more privileged high schools. There is no sound reason for neglecting the capable youth, who may not have been so fortunate as others, but who would take advantage and benefit from whatever opportunities he may at long last encounter.

THE SCIENTIFIC PROGRAMMING COURSE

1. Introduction

The Scientific Programming Course was designed for mathematics- or science-oriented students. The scholastic requirement imposed, however, was minimal: a year of high school algebra with average-or above grades. Our rationale for this is that programming aptitude is not sufficiently well-correlated with the usual academic subjects to warrant limitation to above average students. It is more important that the student have an interest in the subject and, of course, an aptitude for the peculiar rigors of programming. The interest is important, for without motivation little can be learned.

The aptitude for programming is reflected in the abilities to reason logically, to solve simple mathematical problems and, especially, to carry out tedious algorithms with precision. An aptitude test was designed to detect the foregoing abilities. (See Appendix E.) It was used as part of the admission process, and also for class placement. A prime concern was that we attempt to minimize the chances of (relative) failure and its concomitant discouragement by excluding those students who in all likelihood would not profit from the course, and by keeping the level of students within each class fairly uniform. It was also decided to exclude students having more than 20 hours of prior training in computer programming, since the course was designed for the novice.

The Scientific Programming Course was an 80-hour course. It was offered in two consecutive sessions in order to keep class sizes relatively small while accommodating 150 students. The length of each session was four weeks, consisting of five days a week of lectures, laboratory work, and individual instruction as necessary, totaling four hours a day, half in the

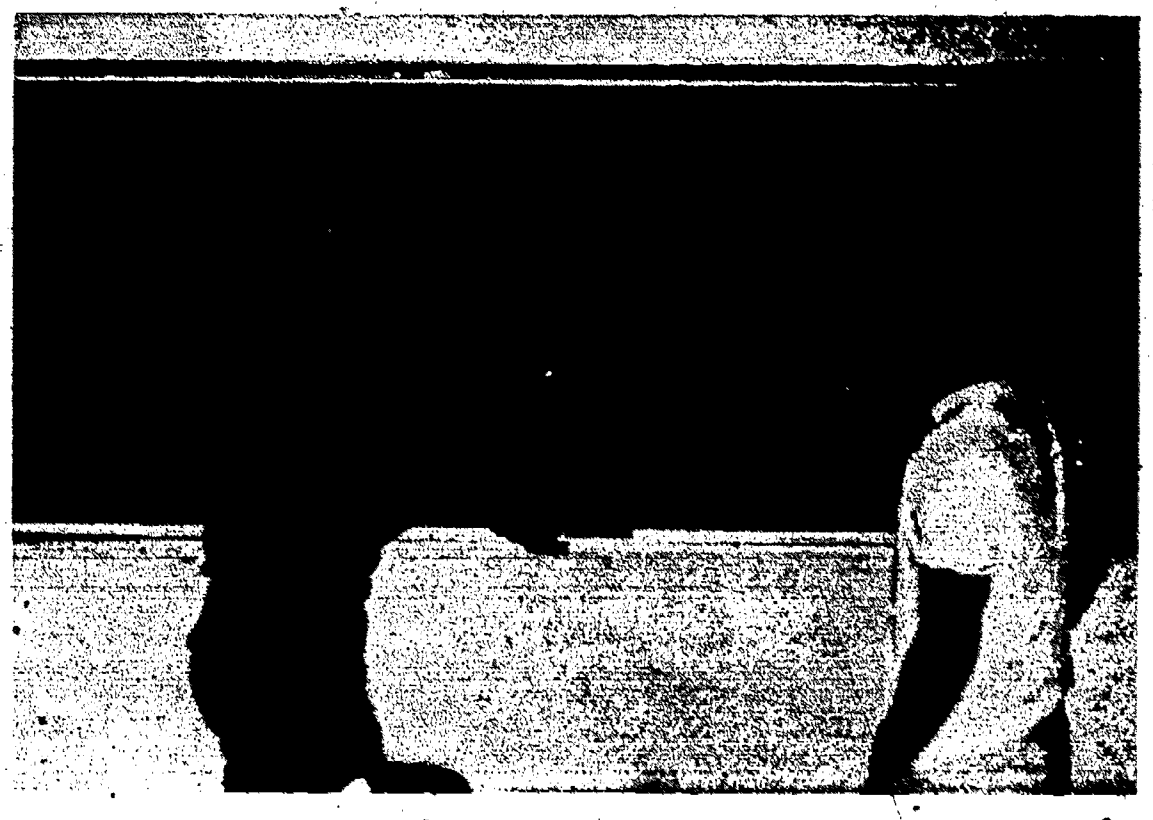
morning and half in the afternoon. The enrollment in each session of 75 students was divided into three classes of 25 students each. A schedule of classes is given in Appendix D.

Amid the glare of the TV camera, students in a scientific programming class take notes.

The placement of students was based in large measure on the results of the aptitude test. The test was given three weeks prior to the start of the first session. Applicants were given the opportunity to express their preferences for sessions at that time. A course calendar is appended (Appendix A).

A typical day for the students was comprised of two hours of classroom instruction (lectures and discussion) and two hours of laboratory work, in either order. In the laboratory the students were provided with practical programming experience. There

were daily assignments which kept each student busy coding, debugging; or keypunching. Laboratory assistants were present at all times to answer questions and to provide whatever other assistance the students required. Tutors were also available for pvivate consultation.



Gary Bloom (left), scientific programming tutor, tries to "reach" a student.

a. Course Description and Objectives

In developing the curriculum, our primary goal was that, upon successful completion of the course, the student should be able to write simple computer programs to solve whatever numerical problems he might encounter later in college or on the job. Our rule of thumb is that, if the student can solve a problem on paper, he should be able to solve it using a computer. To this end, the course provides training in the use of digital computers to solve

problems in mathematics and science. Actual experience in programming an IBM System/360 computer using the FORTRAN IV language was provided. Most importantly, general concepts basic to the understanding of any computer or programming language were discussed. Knowledge of FORTRAN and basic computer concepts should thus enable the student to readily learn other computer languages as the need arises. The preparation provided by this course should also accelerate the rate at which the student can learn to handle the more complex problems with which he is sure to be confronted.

Andy Yakush (seated), scientific programming instructor, assists a student in debugging his program

b. Course Material

The text adopted for this course was Computer Science: FORTRAN Language Programming, co-authored by A. I. Forsythe, T. A. Keenan, E. I. Organick, and W. Stenberg, which was published by John Wiley and Sons, Inc., New York, 1970. Its companion volume, Computer Science: A Primer, was used as a reference but not required. Supplementary notes written especially for the course include:

1. "An Introduction to Computers and Programming," by A. Lew,
2. "Basic Computer Concepts," by A. Lew,
3. "Numbers and Their Representations," by E. Angel and A. Lew.

These are appended. (Appendix D, Scientific Programming Course)

Practical programming experience was provided by means of a series of laboratory assignments. (See Appendix E, Scientific Programming Course, for a listing of the lab problems.) These assignments required each student to write actual FORTRAN programs to be run. Keypunches were available for the students to punch their own programs onto cards. The programs were then run on an IBM System/360 Model 44 computer, housed in the Systems Simulation Laboratory of the University of Southern California School of Engineering.

c. Course Outline

In the following pages, we present two outlines of the Scientific Programming Course: (1) a topical outline, (2) a chronological outline. The topical outline is a listing of the topics covered in the course. The topics are grouped under three headings: (I) Basic Concepts, (II) FORTRAN IV, and (III) Mathematical Topics. The chronological outline is listed by lessons. The subject matter for each of sixteen lessons is briefly noted. A more

detailed description of the course as well as teaching methods and techniques used can be found in Part III, Report on the Scientific Programming Course, by Dr. Art Lew, curriculum coordinator for that segment of the 1970 Project Soul.

COURSE OUTLINE (TOPICAL)

I. BASIC CONCEPTS
   a. The stored-program concept
   b. Basic computer operations
   c. Compilers and operating systems
   d. Algorithms and flow charts
   e. Programs and languages
   f. Tracing a program
   g. Representation of numbers

II. FORTRAN IV
   a. FORTRAN elements
   b. Input and output
   c. Assignment statements
   d. Conditional branching
   e. Subscripting
   f. Iteration
   g. Subprograms

III. MATHEMATICAL TOPICS
   a. Number systems
   b. Roundoff
   c. Quadratic equations
   d. Euclidean algorithm
   e. Square root
   f. Sine function
   g. Roots of equations
   h. Averages and deviations
   i. Areas under curves
   j. Gaussian elimination

COURSE OUTLINE (CHRONOLOGICAL)

LESSON	SUBJECTS
1	A general introduction. Sample programs.
2	The stored-program concept. Basic computer operations. Compilers and operating systems.
3	Algorithms and flow charts. Tracing a program. FORTRAN as an algorithmic language.
4	FORTRAN elements. Input-output. Carriage control.
5	Number systems. Integer numbers. Real numbers. Computer storage of numbers. Roundoff.
6	Assignment statements and other FORTRAN elements. Integer and real representation of numbers. Real arithmetic.
7	Decisions, conditions, and branching. Logical expressions. Quadratic equations.
8	Literals and spacing. Euclidean algorithm. REVIEW.
9	Subscripted variables. Memory allocation. Input and output of arrays. Applications.
10	Iteration and DO loops. Input-output and nested DO loops. The iteration box.
11	Square root. Sine function. Associated mathematics.
12	Subprograms. Arguments. Examples.
13	Roots of equations. (The Newton-Raphson Method.)
14	Averages and deviations. (Statistics.)
15	Areas under curves. (Integration.)
16	Gaussian elimination. (Matrix algebra.)

d. Assessment

The concensus of opinion was that in general the course was a fruitful and rewarding experience for staff and students alike. There was no doubt that non-honor high school students could be taught scientific programming in as little as four weeks. This was borne out again in the course this summer, as it had been the previous two summers in predecessor courses, as well as in courses across the country.

Most of the remarks made in last year's report remain valid. Academically oriented students, many of whom will use their newly acquired knowledge as a tool in their college work, became more enthusiastic about pursuing scientific research after having been exposed to the use of computers in solving problems. Less academically inclined students showed much more interest than they had in their high school science and math courses, which they claimed were often too theoretical and irrelevant to their needs. There were, unfortunately, a number of students who lacked either sufficient motivation or perseverance or aptitude, and consequently did not derive much benefit from the course. A major question is whether a more careful pre-evaluation and admission policy might have or should have eliminated these students from the start.

A question raised by one instructor was that of whether our students should be precisely those who have found no motivation in their traditional educational experience. In pre-session discussions, it was decided that there was very little we, having no formal pedagogical or psychological training, could do with such students. In any case, it was concluded that the problems these students face are so deep as to be impervious to a limited four-week attack. Our course was seen as providing an opportunity for those who are motivated and have the aptitude (albeit not reflected in their performances in high school classes), rather than as attempting to provide motivation to those who lack same.

While it was our objective to provide preparatory rather than remedial training, we must still be concerned with the motivation of the students, at least to the extent of not discouraging anyone. This is a demanding requirement and necessitates a conscientious effort on the part of the entire staff. The extent to which the staff was successful in this effort is difficult to measure.

Other criticisms made related to the content of the course and to the text. The text was not found to be very helpful. This did not, however, present a serious problem. The course had been designed with most of the burden placed on the instructors. Furthermore programming, more so than other subjects, is learned by doing. Hence, although the text should be chosen wisely, it is more important to concentrate on improving the quality and efforts of the instructional staff. It should be noted that no book examined was found to be entirely suitable for this course.

The course content was generally found to be satisfactory. Some instructors expressed a preference for a different ordering of subjects and a slight shift in emphases. The DO statement was thought to be too closely linked with subscripted variables, and a simplification of input-output by use of standardized formats was suggested. Some students also had difficulty with various mathematical concepts, but a deemphasized exposure to the concepts is of sufficient value to make their complete elimination a dubious alternative.

Finally, the desire for the inclusion of more exercises and laboratory problems in the lesson plans was expressed. More could be easily found in the multitude of FORTRAN texts that abound, and consequently there was no real difficulty in leaving additional assignments to the discretions of the instructors. One other criticism was that the laboratory problems provided were too easy at the start while relatively difficult and time-consuming near the end

of the course. One suggestion for remedying this is discussed in the concluding section.

e. Recommendations

We conclude this report on the Scientific Programming Course with a list of recommendations for improving the operation and content of the course. We first should emphasize that the course has been found to be extremely valuable and should be offered again, perhaps on an annual basis. Furthermore, we recommend that the course, or a facsimile, be offered across the country for there is need everywhere. Recommendations for future courses include the following.

A. ADMISSION. The more effort made in pre-evaluation of applicants, the more successful the course is likely to be. There is of course a practical limit to the amount of screening that can be performed, but some attempt should be made to limit enrollment to those showing motivation and aptitude. (We discuss this further below.)

B. SCHEDULE. Because of the nature of the subject, it is recommended that the course be taught with a shifting schedule. For example, during the first week(s) of the class, three-fourths of the day should be lecture, one-fourth laboratory; during the final week(s), this would be reversed, with one-fourth lecture and three-fourths laboratory. The middle week(s) may be split evenly between lecture and laboratory.

C. STAFF. Great care should be taken in the recruitment of the instructional staff. Desirable qualities for instructors, laboratory assistants, and tutors include knowledgeability, enthusiasm, and a sincere desire to help. Of course, some prior experience in teaching would be preferable, but this has been found to be unnecessary. Of greater importance is the ability to interact well.

D. INTERACTION. It is of great importance to establish rapport with each student. This has the benefits of increasing the motivation of the students, aiding thereby in their learning process, and as a consequence rewarding the members of the instructional staff, stimulating thereby their enthusiasm, whence also their effectiveness. The instructors should see each student individually as often as feasible and should encourage class discussion by all. The laboratory assistants should take the initiative in soliciting questions from the students, for those who need the most help generally will not seek it. It is recommended that this be done by reviewing each student's progress periodically each day in his presence. This would encourage the students to make better use of their time in the laboratory, and would benefit the more reserved or shy students who would otherwise remain withdrawn. The instructors and laboratory assistants should not hesitate to refer students to the tutors; again this should not be voluntary, but should also not be an ordeal nor a stigma. The tutors, for their part, should be well versed in the Socratic method for best effectiveness. Ideally, the students should regard each member of the instructional staff as a friend to whom they may readily and naturally turn.

E. CONTENT. The content of the course is basically sound. However, for future courses with different emphases, some modifications would be appropriate. A more concrete job-oriented course would result by eliminating most of the basic introductory material and starting with FORTRAN elements. (FORTRAN texts by and large are of this nature.) A less mathematical course would result by deleting Lesson 5 and treating the other mathematical topics in a cookbook fashion. A course for honor students would result by discussing each subject more deeply, providing more motivation and background for the various concepts, and also by strengthening the mathematical content. A simplified course would result by eliminating most of the discussion of input-output problems and utilizing standardized formats.

F. DURATION. While the progress made by most students in the eighty hours of the course was most gratifying, there is little doubt that a longer course would be of great benefit. EDP school programming courses, for example, provide five to ten times the hours of instruction. College courses, on the other hand, generally provide fewer hours (around 40 to 50) of instruction. (In college, however, laboratory work is not usually included in the above hour count as the students are expected to do most of their work on their own outside the classroom.) In any case, it is recommended that the duration of the course be increased.

G. OBJECTIVES. The limitation to motivated and apt students is justifiable in relation to our stated objectives and capabilities. Regrettably, many students, and in particular those who have a substantial need, cannot meet our admission criteria. This leads us to the question of whether or not our objectives are too narrow, and whether a similar course should be designed for those who have the aptitude but lack motivation or for those who have the motivation but lack the aptitude. It should be emphasized that a course for these neglected students must necessarily differ in content from the present course. Furthermore, to handle such students would require an instructional staff well-trained in educational theory and psychology. It is felt that efforts along these lines would be of avail and value, and therefore it is recommended that such remedial courses be designed and implemented.

H. SCOPE. Because of the success of the program this year, and of the programs of the previous two years, we feel that the Scientific Programming Course has proven its value. Therefore, we again recommend an expansion of the program to accommodate as many underprivileged youths as funds, facilities, and personnel will allow. Furthermore, we recommend that similar programs be made available across the country, wherever there are students, who because of inadequate schools and a lack of community concern

find themselves increasingly at a disadvantage. We note that programming courses are presently made easily available to students at middle or higher-class schools, and a multitude of courses for honor students abound. There is no sound reason for neglecting the average but capable high school youth, who may not have been as fortunate as others, but who would take advantage and benefit from whatever opportunities he may at long last encounter.

PROJECT SOUL 1969 AND 1970--AN OVERVIEW

As part of our project, an evaluation was undertaken of all participants in the 1969 and 1970 Project Soul. The purpose of this evaluation was to find out some information about the attitudes of the high school students who participated in Project Soul during the summers of 1969 and 1970. The 1970 participants were required to fill out a questionnaire on the first day of classes and a second questionnaire on the final day of classes. One hundred seventy-two questionnaires were completed on both of these occasions. One hundred questionnaires were sent out to students who participated in Project Soul in 1969 and 39 were completed and returned, a high number considering the questionees had not been contacted for approximately one year. Copies of the questionnaires are found in Appendix F.

The purpose of these questionnaires was to determine the students' reaction to their experience in Project Soul. An attempt was made to find out if the program had affected their attitudes towards their plans for education and careers, to find out if they have a realistic view of the capabilities and potentialities of the computer, both as a tool in a technological society and as a possible factor in their own futures, and to find out how they benefited from, and their reactions to Project Soul.

The data presented for the 1970-session students represents the combined scores for 172 students in nine different classes: six classes in scientific programming with an emphasis on FORTRAN (132 students), two classes in data processing with an emphasis on COBOL (30 students), and one class in keypunch operation (10 students). The data for the 1969-session students represents the combined scores of the 39 questionnaires that were returned. The 1969 classes consisted of two classes in scientific programming (80

students, 34 questionnaires returned) and one class in data processing (20 students, 5 questionnaires returned).

The data presented is in the form of percentages, representing the percent of the total number of students in any group (1969 or 1970) giving a particular answer. It should be noted that for certain questions, the percentages given for individual answers may sum to greater than 100 percent. This was possible because certain questions lent themselves to several answers. For example, if a student was asked how he benefited from his experience in Project Soul, he might say that it increased his job opportunities and he will be helped in school. In this case, he would be given credit for both answers.

All students were asked how they will benefit or how they have benefited from their participation in Project Soul. The following answers were given:

	1970 Questionnaire 1	1970 Questionnaire 2	1969 Follow-up Questionnaire
Increased job opportunities	59%	32%	13%
Increased general knowledge	36%	59%	69%
Help in school	29%	9%	23%
Personal improvement	2%	63%	68%
Financial gains	0%	6%	0%
No benefit	2%	1%	5%
No answer	3%	5%	0%

TYPICAL RESPONSES:

1970-QUESTIONNAIRE 1

1. How do you feel you will benefit from taking a course in computer programming?

I feel I will benefit from this course because I would like to go into this field in college. I think it will also help me get a job because I can put this down as a reference. If I do get a job in computer programming, I'll have this much knowledge.

1970-QUESTIONNAIRE 2

1. How have you benefited from your participation in Project Soul? Please include practical benefits such as improved job opportunities and also personal benefits such as new friends or increased confidence.

Project Soul can help me, maybe, get a job. It has helped me understand and appreciate computers a little more. I think it was a good experience going to a college, and meeting students from other schools.

1969

1. How have you benefited from your participation in PROJECT SOUL?

Through my participation in "Project Soul", I feel that I have gained such an exciting experience in this particular field of computer technology. I have also gained a more or less intellectual feeling of working and competing against various other students who some have had a deeper and broader knowledge of mathematics in dealings with computers. And, I have more than ever enjoyed working on the University level.

Although a majority of the students originally felt that their participation in Project Soul would increase their opportunities to get jobs, by the end of the program they were less enthusiastic about their employment potential. At the end of one year, very few felt that they still had greater job opportunities as a result of the project. It should be noted that very few of the students were actually intending to get full-time jobs during the year following their participation (see below) and therefore very few had actually tested the marketability of the skills they had learned.

The decrease in the percentage of students indicating that their participation in Project Soul would be of particular help in school probably reflects the fact that most students would not be using computers in their immediate academic futures. The 1969 scores indicate that be the end of one year, 23% had actually used the knowledge that they had gained.

There was a rise in the percentage of students indicating they felt an increase in general knowledge, especially about computers and math, would be a benefit derived from their participation. This probably reflects the fact that when they originally started the program, they expected to learn skills that had specific application to school and the job market. When they found that this was not completely true, they put more emphasis on the general nature of their educational experience. This pattern is seen in the rise in the percentage of students indicating that they received such personal benefits as increased confidence, increased ability to meet and work with people, etc. Since they had not found the training specifically applicable to jobs and school, their emphasis shifted to more personal benefits.

Very few students felt that the expense allowance had been a significant benefit of the program, or felt that they had received no benefits from the program.

The 1970 students were asked to describe their plans for the year following their participation in Project Soul and the 1969 students were asked to describe what they had actually done during the year following their participation in Project Soul:

	1970-Ques. 1	1970-Ques. 2	1969
Start college	44%	47%	51%
Finish high school	37%	37%	46%
Find job	7%	6%	0%
Start commercial training	2%	4%	0%
Start computer training	4%	2%	0%
Undecided	3%	2%	3%
No answer	2%	2%	0%

They were then asked how much education they would like to receive if they did not have to worry about money, academic record, family responsibilities, etc. Their responses were as follows:

	1970-Ques. 1	1970-Ques. 2	1969
As much as possible	4%	18%	3%
College education	51%	31%	6%
Professional or graduate school	24%	32%	71%
2 year college	5%	2%	0%
Other (trade schools, etc.)	3%	12%	17%
No answer	13%	5%	3%

In general, the Project Soul participants are oriented towards continuing their education, at least to the college level. These results are not unexpected as they all had volunteered to take part in a summer educational program that would not afford them great monetary compensation and would not fully train them for a job.

When the students were asked what type of jobs they would look for if they had to find a full-time job immediately, the students gave the following answers:

	1970-Ques. 1	1970-Ques. 2	1969
Computers and related	7%	41%	46%
Clerical-secretarial	35%	22%	18%
Unskilled or manual labor	16%	9%	13%
Business training program-sales	24%	6%	2%
Definitely not computers	0%	3%	13%
Other	0%	4%	5%
Undecided	24%	12%	2%
No answer	7%	9%	0%

TYPICAL RESPONSES:

1970-QUESTIONNAIRE 1

12. If it were necessary for you to get a full-time job immediately, what kind of work would you look for?

Clerk-Typist. Most of my classes in high school were business courses.

1970-QUESTIONNAIRE 2

9. If it were necessary for you to get a full-time job immediately, what kind of work would you look for? Has your participation in Project Soul affected your choice?

I WOULD LOOK FOR KEYPUNCH OR PATH PROCESSING JOB first, AND then WHATEVER "odd" jobs I could find. Yes, it has given ME enough KNOWLEDGE to HANDLE MORE TRAINING IN KEYPUNCH & DATA PROCESSING EASIER.

The following responses were given to the question as to whether or not participation in Project Soul had an affect on their choice:

	1970-Ques. 1	1970-Ques. 2	1969
Affect	--	30%	46%
No affect	--	25%	21%
No answer	--	45%	33%

There is an obvious increase in the number of students who would be interested in a job related to computers, either as a programmer, operator, or keypuncher. The relative drop in interest in clerical-secretarial jobs, as indicated by the students' comments on their questionnaires, appears to be caused by their exposure to, the computer field and to the fact that certain positions in the computer field such as operations and keypunching require a few more skills than most secretaries have.

TYPICAL RESPONSE:

1970-QUESTIONNAIRE 2

9. If it were necessary for you to get a full-time job immediately, what kind of work would you look for? Has your participation in Project Soul affected your choice?

Clerical type in a company that have Computer. Yes, the keypunch course have given me experience Which most company wants

Many of the students who chose unskilled or manual labor felt that their training was not sufficient to qualify them for any particular type of job in the computer field or any other field requiring training.

TYPICAL RESPONSE:

1970-QUESTIONNAIRE 2

9. If it were necessary for you to get a full-time job immediately, what kind of work would you look for? Has your participation in Project Soul affected your choice?

Stock clerk, warehouseman but if it weren't for my lack of programing knowledge, I wouldn't mind the challenge and it's reward that computer science offers

The decrease in interest in business training programs or sales jobs probably also reflects an increased awareness and interest in computers on the part of many of the students.

In response to the question as to what they felt their chances of actually obtaining the job of their choice and whether those chances were affected by their participation in Project Soul, the following answers were given:

	1970-Ques. 1	1970-Ques. 2	1969
Excellent chances	11%	5%	10%
Good chances	35%	43%	33%
Fair chances	35%	35%	38%
Poor chances	18%	13%	10%
No answer	1%	4%	9%
Project Soul had an affect	--	41%	41%
Project Soul had no affect	--	41%	38%
No answer	--	18%	21%

The students' own estimation of their chances of obtaining the job of their choice did not change greatly, although there was some increase in the percent of students that felt their chances were good and some decrease in the percent that felt their chances were poor. The fact that these results do not appear to reflect the large percentage of students who felt that Project Soul had affected their scores is probably due to the increase in those who would try to find a job in the computer field (see above). Many of these students felt their increased training and their certificates of completion would help them find computer jobs but they realized that these factors did not guarantee their success.

TYPICAL RESPONSE:

1970-QUESTIONNAIRE 2

10. Which of the following best describes your chances of getting the job you would want if you had to get a full-time job immediately?

_____ Excellent
_____ Good
_____ Fair
_____ Poor

Has your participation in Project Soul affected your chance's?
Please explain your answer.

a little. It hasn't as far as a job in a marked, but it has for my choice in a computer outfit. Because if you know at least a little bit of what you are going to do, you have a better chance of getting the job

When requested to describe their present career objectives or goals for their lives and whether or not these objectives and goals had been affected by their participation in Project Soul, the students gave the following descriptions:

	1970-Ques. 1	1970-Ques. 2	1969
No answer	13%	9%	0%
Undecided	0%	20%	21%
Medicine and related fields	19%	9%	15%
Computers and related fields	19%	21%	15%
Art or music	5%	2%	3%
Clerical-secretarial	5%	2%	0%
Math, science or engineering	34%	22%	46%
Social sciences and services	8%	2%	5%
Professional - lawyer, historian, etc.	6%	5%	3%
Business	8%	3%	0%
Teaching	13%	5%	0%
Will not work with computers	0%	5%	5%
Will try to make much money	4%	0%	0%
Will try to be happy and successful	10%	2%	0%
Project Soul has affected goals	--	44%	28%
Project Soul has not affected goals	--	35%	12%
No answer	--	21%	60%

As can be seen from the data, the Project Soul participants form a heterogeneous group with respect to their career and life objective. The modal response in the science-math-engineering group is to be expected because of the technical nature of the subject being studied in this program. There was little apparent change in the percentage of students interested in a career in computers but this does not necessarily mean that Project Soul had no affect on students' objectives. It is possible that a large number of students were oriented towards a career in the computer field as a result of the program, but an equally large number of students were oriented away from the.

field. This would not be reflected in the data. It is obvious that participation in Project Soul did have an affect, as approximately 1/2 to 2/3 of the students that answered the question indicated that their goals had been affected.

The students gave the following estimates of their opportunity for individuality and creative work in the computer field:

	1970-Ques. 1	1970-Ques. 2
Excellent opportunity	16%	9%
Good opportunity	51%	46%
Fair opportunity	23%	33%
Poor opportunity	4%	9%
No answer	6%	3%

There was a slight but observable drop in their estimate of their own opportunity for individuality and creative work in the computer field. According to comments on the questionnaires, students realized how demanding computer programming could be and some felt that they did not have the background and abilities in math and related areas necessary for success in the field.

TYPICAL RESPONSE:

1970-QUESTIONNAIRE 2

12. Which of the following best describes the ~~opportunity you think~~ you will have for individuality and creative work in the computer field:

_____ Excellent Opportunity _____ Good Opportunity

_____ Fair Opportunity _____ Poor Opportunity

Please give reasons for your answers.

My background is limited and I didn't appear to grasp the basic concepts as well as others

The 1970 students felt that the following degrees were necessary for success in the computer field:

	1970-Ques. 1	1970-Ques. 2
Engineering degree	48%	26%
Mathematics degree	33%	33%
Either mathematics or engineering	3%	27%
Neither degree	28%	19%
Neither is necessary but one would be helpful	10%	24%
No answer	8%	8%

At the beginning of the program, 48% of the students felt that a degree in engineering was necessary for success in the computer field while at the end of the program, only 26% felt it was necessary. There was no change in the percent of students that felt a degree in mathematics was necessary. This probably reflects an increased understanding by the students of the requirements of computer programming. Many of the students came to understand that one could work with computers without knowing the electronic and mechanical aspects of the machines themselves. The above data indicates that the importance of a degree, or at least advanced training, is evident to the students, but the lack of a degree does not make it impossible to work in the computer field.

When asked to estimate the average salary of a computer programmer, the students from the 1970 session answered as follows:

	1970-Ques. 1	1970-Ques. 2
$20,000/year or more	7%	6%
$17,000-$19,999/year	1%	1%
$14,000-$16,999/year	8%	13%
$11,000-$13,999/year	8%	16%
$8,000-$10,999/year	29%	27%
$6,000-$7,999/year	17%	11%
$5,999/year or less	0%	5%
Very little money	0%	1%
Very much money	5%	3%
No answer	25%	17%

The median estimate fell in the category $8,000-$10,999, which is the most accurate estimate of the starting salary of a beginning computer programmer with a college degree. Many of the students made estimates that were slightly higher than realistic considering the job market at the time the questionnaires were filled out. In the future, care should be taken to make sure that all participants in Project Soul have a sufficient understanding of the economics of the computer industry to allow them to make a wise judgment of their ability and desire to succeed in this field.

In order to determine where the students had learned about computers prior to their participation in Project Soul, the 1970 participants were required to indicate on the first questionnaire which of the following sources they had used:

Books	46%
Newspapers	33%
Magazines	83%
Television	55%
Radio	22%
School	72%
Family	31%
Friends	42%
No answer	6%

In an attempt to determine if Project Soul had given the students an idea of the broader applications of computers, they were asked if the computer could be useful in helping to solve some of the major social problems that society faces. Their replies follow:

	1970-Ques. 1	1970-Ques. 2	1960
Yes, it can help	33%	54%	69%
No, it cannot help	41%	15%	26%
No answer	26%	31%	5%

TYPICAL "YES" RESPONSE:

1970-QUESTIONNAIRE 2

16. Take one social problem, such as pollution, and briefly describe how the computer could aid in the solution of that problem.

First give the computer the problem as it is and give the causes which helped in this problem. I'm sure it'll give an answer.

TYPICAL "NO" RESPONSE:

1970-QUESTIONNAIRE 2

16. Take one social problem, such as pollution, and briefly describe how the computer could aid in the solution of that problem.

It couldn't. How can an impersonal machine that can't even think dictate to people why they shouldn't litter. Social problems, such as pollution, are based on emotions, personal beliefs and environment stimulus. The computer couldn't solve this problem with a word solution

These answers are typical of the explanations given on all three questionnaires. It appears that while participation in Project Soul made the students more aware that computers could be useful in finding solutions to major social problems, it did not leave them with a particularly clear understanding of how they could be used. Several students had requested that more emphasis be given to applications and these results point to this comment as a necessity.

When asked if the expense allowance was sufficient, the following responses were given by the 1970 (Questionnaire 2) students:

Yes	74%
No	15%
No answer	11%

Many of those students who felt that the allowance was not sufficient were comparing the amount they received with what they felt they could have made if they had taken summer jobs. Others noted that they had relatively large travel expenses because they lived a long distance from the University of Southern California. Neither of these objections are completely valid as the students were informed in advance that they would have to make their own provisions for getting to classes and that the program was being run for educational purposes and was not to be considered a full-time job.

When asked if they would have been able to participate in Project Soul if the allowance for expenses had not been available, the students from the 1970 session gave the following answers:

	1970-Ques. 2
Yes	61%
No	24%
Yes, but the program would not have been as attractive	10%
No answer	5%

For 24 percent of the students, participation in Project Soul would have been impossible without the expense allowance, and at least another 10 percent felt that the allowance made the program more attractive. A vast majority of the students felt that receiving some compensation for their time and effort increased their enthusiasm to participate.

The 1969 students were asked if their certificates of completion had been useful in obtaining jobs or furthering their education. Their responses follow:

Yes, it helped to get a job	16%
Yes, it helped in school	18%
Never tried to use it	50%
No, it did not help	16%

Several of those who stated that the certificates had not helped had tried to get a computer programming job and were turned away because they lacked experience. They rarely showed the certificates to their prospective employers. Students should be given a clear explanation of the present employment conditions in the computer industry in terms of what to expect if they do try to find jobs in this field.

TYPICAL RESPONSE:

1969

14. Was your certificate of completion from Project Soul useful to you in getting employment or in furthering your education. Please explain your answer.

No. In most (if not all) places I sought employment I filled out an application. In all cases I was rejected because of lack of experience. In no circumstance did I show my certificate to an employer.

Several suggestions were made for changes in the program. While none of these changes was considered necessary by more than 10 percent of the students, all should be considered in the planning of future programs:

1. Make the program longer because four to six weeks was not sufficient time to learn more than the most basic skills.
2. Help should be given in finding jobs that require the skills that have been taught. Prospective employers and employment agencies should be made aware of the program and the market value of the skills taught.

3. Less time should be spent on lectures and more time should be allocated for actual programming and running of programs on the machine.

4. Machine operations should be taught as well as programming and keypunching.

5. Have an advanced class for students who learn more quickly than the majority of the group.

6. Provide more individual instruction, with an emphasis on mathematics.

In general, the students were enthusiastic with their experience in Project Soul and the vast majority felt that the program should be continued.

APPENDIX A

Calendar of Events

"PROJECT SOUL" 1970

April 23, Thursday	Letters to high school principals
May 12-22	Meetings to select candidates (Project Soul Staff/High School Rep.)
May 25, Monday	Deadline for submitting names, addresses and telephone number of candidates
June 6, Saturday	Aptitude Test - USC Campus
June 8, Monday	Candidates will be informed regarding the results of their applications
June 15, Monday	Deadline for accepting or declining an offer to participate
June 16, Tuesday	Selection of alternates to fill vacancies
June 19, Friday	Class schedule will be mailed to each participant
June 29, Monday	Keypunch course, Commerical D.P. course, and First Session Scientific Programming course begin
July 24, Friday	Keypunch and First Session Scientific Programming courses end
July 27, Monday	Second Session Scientific Programming course begins
August 7, Friday	Commercial Data Processing course ends
August 21, Friday	Second Session Scientific Programming course ends

APPENDIX B

Organizational Chart

"PROJECT SOUL" 1970

Principal Investigator:
Richard Bellman

Project Director:
Carlos Ford-Livene

Technical Administrator:
M. Virginia Zoitl

Secretary-Placement Aide:
B. Gloster

Scientific Programming Curriculum
Coordinator: Art Lew

Session I:
June 29-July 24

Instructors:
1. Andy Yakush
2. Dan Tuey
3. M. Wasserman

Lab. Assistants:
1. A. Leon
2. P. Kumar
3. G. Nageo
4. R. Schein
5. R. Salas

Tutor:
1. C. Shoemaker

Session II:
July 27-August 21

Instructors:
1. Andy Yakush
2. Dan Tuey
3. B. Kashef

Lab. Assistants:
1. A. Leon
2. P. Kumar
3. M. Wasserman
4. R. Schein
5. R. Salas

Tutor:
1. G. Bloom

Data Processing Curriculum
Coordinator: James Woolever

Keypunch Operator Course

Instructors:
1. John Kelder

Business Data Processing Course

Instructors:
1. N. Gale
2. J. King

Lab. Assistant:
1. Richard Avery

APPENDIX C

a. Responsibilities of the Curriculum Coordinator
b. Responsibilities of the Instructors
c. Responsibilities of the Laboratory Assistants, Tutors, and Special Aides

a. The responsibilities of the Curriculum Coordinator were to:

1. Develop the curriculum for the course.
2. Prepare a general teaching outline, including daily lesson plans, reading assignments, and laboratory problems.
3. Design a programming aptitude test to assist in the selection of students.
4. Conduct orientation and briefing sessions for the instructional staff.
5. Coordinate and supervise the work of the instructional staff, and assist in the implementation of the curriculum as the need arises.
6. Submit a detailed personal evaluation of the course in general and to give recommendations for future projects.

b. The responsibilities of the Instructors were to:

1. Present daily lectures according to the lesson plans provided. (Variations at the discretion of the Instructors were permitted.)
2. Assign reading, laboratory problems, and other homework.
3. Design, administer, and grade examinations.
4. Supervise the activities of the Labbratory Assistants.
5. Refer students to the Tutors for individual instruction as the need arises.
6. Monitor the progress and make a final evaluation of each student.

c. The responsibilities of the Laboratory Assistants were to:

1. Do the laboratory problems (before assignment to students).
2. Give students instructions on the use of the keypunch.

3. Check laboratory and other homework assignments.
4. Assist students in writing and debugging their programs, and in interpreting error messages.
5. Maintain accurate records of each student's progress in their laboratory work.
6. Recommend tutorial assistance.
7. Provide feedback to the Instructors.

The responsibilities of the Tutors were to:

1. Answer any questions posed by the students.
2. Provide individual tutoring to students as the need arises.
3. Provide feedback to the Instructors.

The Special Aides consisted of (i) job carriers, who were responsible for the submittal and return of student jobs from the computer center, and (ii) secretarial help, who typed, ran dittos, collated, and sundry other tasks.

APPENDIX D

Schedule of Classes

COURSE	DATE	CLASS SIZE	TIME	LOCATION	INSTRUCTOR	KEYPUNCH DISTRIBUTION
Keypunch	June 29-July 24	10	10-12 1-3	OHE 134	John Kelder	OHE 132-5kps
Business Data Processing, Section A	June 29-August 7	20	10-12 Lab. 1-3 Lect.	CSL 113 CSL 129	Norman Gale R. Avery (Lab. Assistant)	CSL-5kps
Business Data Processing, Section B	June 29-August 7	20	10-12 Lect. 1-3 Lab.	CSL 129 CSL 113	Jules King R. Avery (Lab. Assistant)	CSL-5kps
Scientific Programming, Section A	June 29-July 24	25	10-12 Lab. 1-3 Lect.	OHE 210 VHE 310	Andy Yakush	OHE 210-5kps
Scientific Programming, Section B	June 29-July 24	25	10-12 Lect. 1-3 Lab.	VHE 310 OHE 210	Dan Tuey	OHE 210-5kps
Scientific Programming, Section C	June 29-July 24	25	10-12 Lect. 1-3 Lab.	VHE 314 OHE 212	Michael Wasserman	OHE 212-5kps
Scientific Programming, Section D	July 27-August 21	25	10-12 Lab. 1-3 Lect.	OHE 210 VHE 310	Andy Yakush	OHE 210-5kps
Scientific Programming Section E	July 27-August 21	25	10-12 Lect. 1-3 Lab.	VHE 310 OHE 210	Dan Tuey	OHE 210-5kps
Scientific Programming Section F	July 27-August 21	25	10-12 Lect. 1-3 Lab.	VHE 314 OHE 212	Bayesteh Kashef	OHE 212-5kps

OHE denotes Olin Hall of Engineering
VHE denotes Vivian Hall of Engineering
CSL denotes Computer Science Laboratory

APPENDIX E

Aptitude Test

INSTRUCTIONS FOR PART I

In Part I you will be given some problems like those on this page. The letters in each series follow a certain rule. For each series of letters you are to find the correct rule and complete the series. One of the letters at the right side of the page is the correct answer. Look at the example below.

		(1)	(2)	(3)	(4)	(5)
W.	a b a b a b a b	a	b	c	d	e

For this problem, the series goes: ab ab ab ab
The next letter in the series is a. Choice 1 is the correct answer.

		(1)	(2)	(3)	(4)	(5)
X.	a a b b c c d d	a	b	c	d	e

In Example X above, the series goes like this: aa bb cc dd. The next letter in the series is e. Choice 5 is the correct answer.

Now do Example Y below.

		(1)	(2)	(3)	(4)	(5)
Y.	c a d a e a f a	d	e	f	g	h

In Example Y, the series goes: ca da ea fa. Therefore, the correct answer is g, Choice 4.

Now do Example Z.

		(1)	(2)	(3)	(4)	(5)
Z.	a x b y a x b y a x b	a	b	c	x	y

In Example Z, the series goes like this: axby axby axb. Therefore, the correct answer is y, Choice 5.

In the problems on the following page, you are to select the correct letter on the right-hand side of the page which belongs next in the series. Indicate the correct answer on the answer sheet.

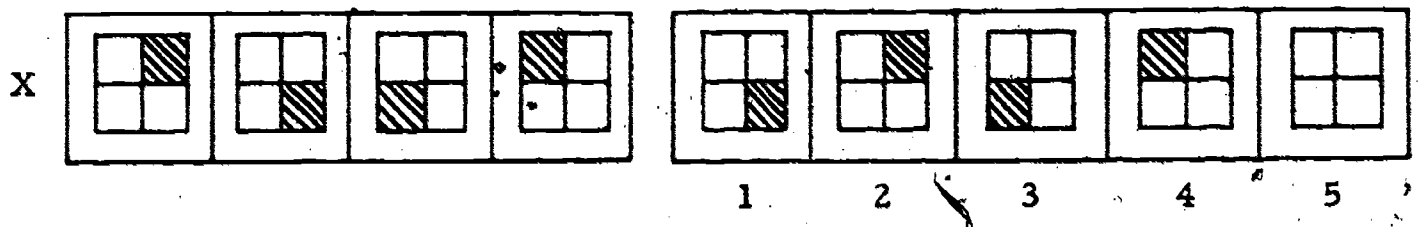
PART I

1.	c d e x y z f g h x y z	(1) i	(2) j	(3) k	(4) l	(5) m
2.	t s r t s r t s	(1) r	(2) s	(3) t	(4) v	(5) w
3.	a b c c d e f f g	(1) e	(2) f	(3) g	(4) h	(5) i
4.	m n m n k l o p o p k l	(1) k	(2) o	(3) p	(4) q	(5) r
5.	a b c i j d e f i j	(1) g	(2) h	(3) i	(4) j	(5) k
6.	a i b c i d e f	(1) e	(2) f	(3) g	(4) h	(5) i
7.	a g b h c	(1) d	(2) f	(3) g	(4) h	(5) i
8.	a e d h g	(1) h	(2) i	(3) j	(4) k	(5) l
9.	k s j t i u h	(1) v	(2) w	(3) x	(4) y	(5) z
10.	n j f m i e l	(1) d	(2) h	(3) i	(4) j	(5) m

INSTRUCTIONS FOR PART II

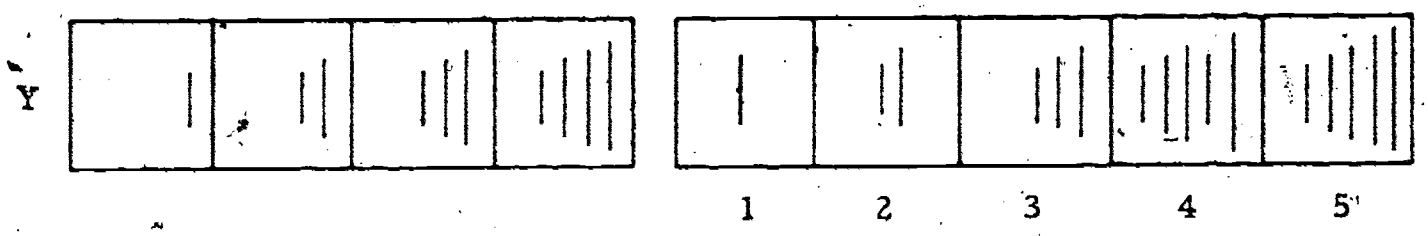
In Part II you will be given some problems like those on this page. Each row is a problem. Each row consists of four figures on the left-hand side of the page and five figures on the right-hand side of the page. The four figures on the left make a series. You are to find out which one of the figures on the right-hand side would be the next or the fifth one in the series. Now look at Example X.

Example



In Example X there is a clockwise movement of the striped square: upper right, lower right, lower left, upper left. The next or fifth position in this clockwise movement would thus be upper right, and so Choice 2 is the correct answer.

Now look at Example Y.

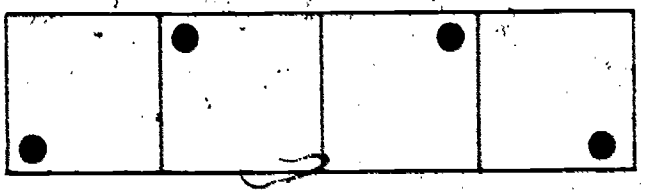


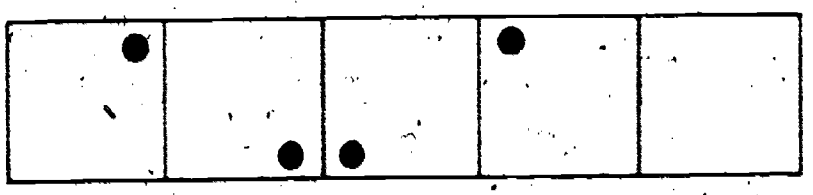
In the series of figures on the left, there is one more line in each box and these lines increase in length. Now look at the five choices on the right-hand side of the page and determine the correct answer.

You should have selected Choice 5 which has five lines, one more than the last box on the left with the fifth line slightly longer than the last line in Box 4.

PART II

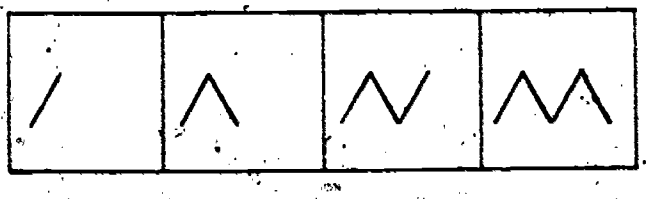
11.

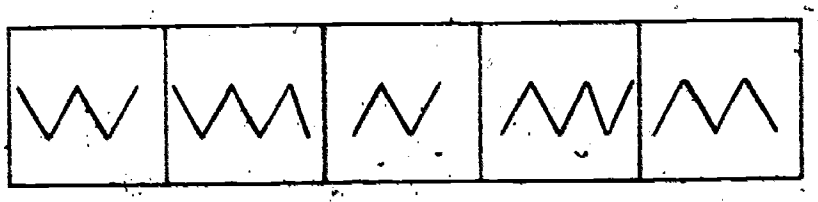




1 2 3 4 5

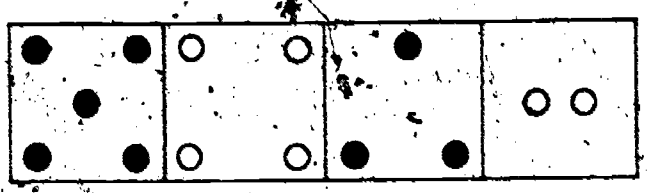
12.

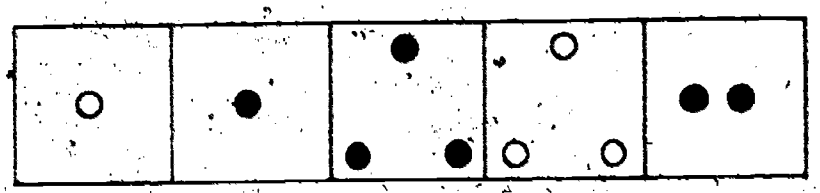




1 2 3 4 5

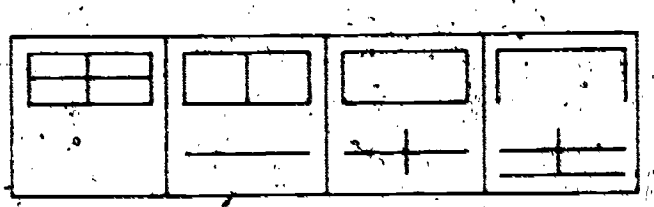
13.

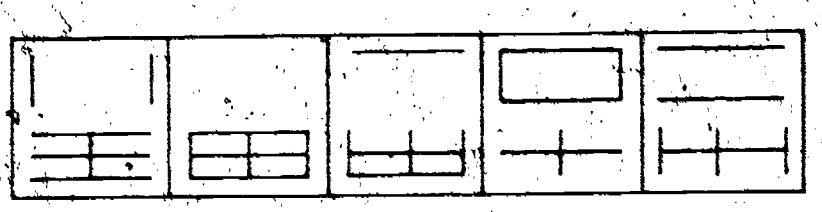




1 2 3 4 5

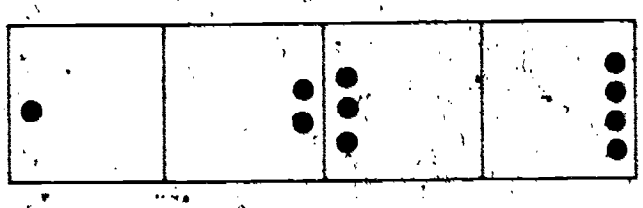
14.

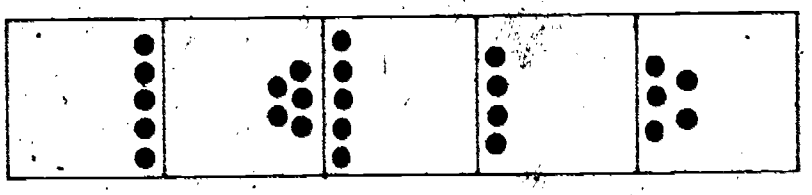




1 2 3 4 5

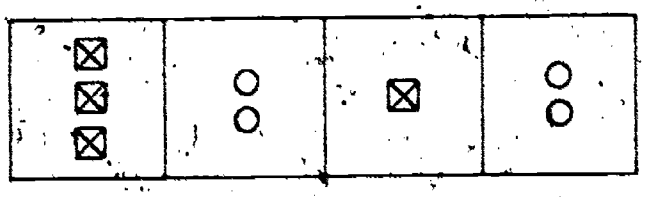
15.

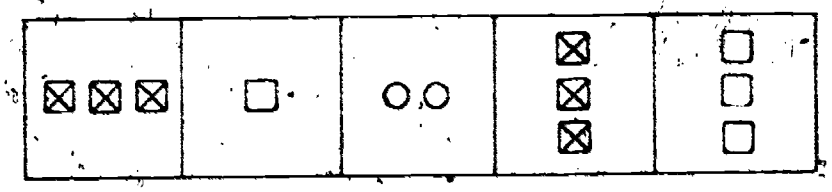




1 2 3 4 5

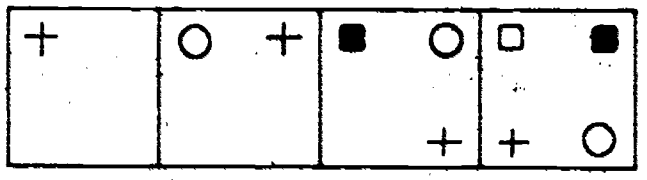
16.

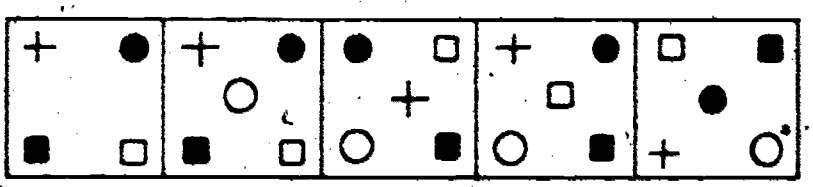




1 2 3 4 5

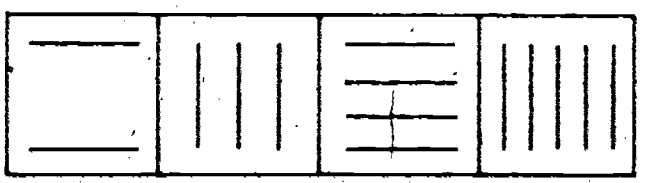
17.

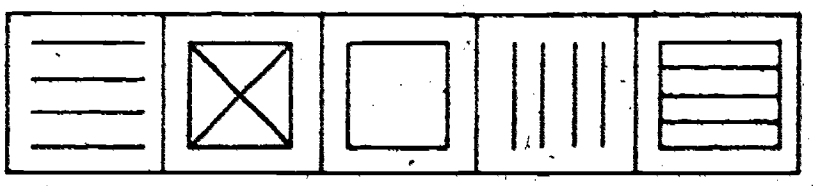




1 2 3 4 5

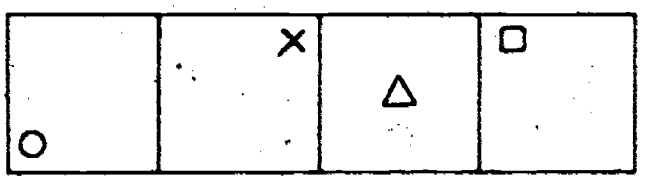
18.

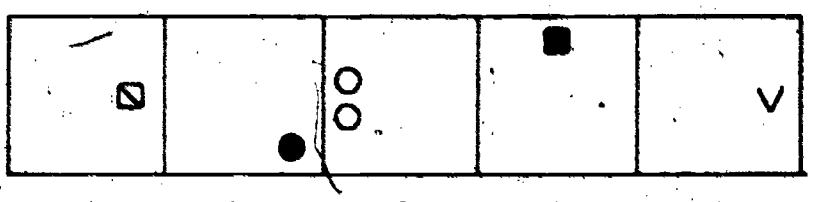




1 2 3 4 5

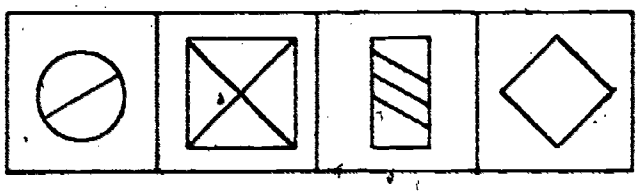
19.

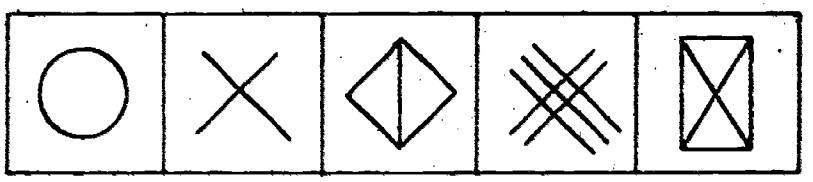




1 2 3 4 5

20.





1 2 3 4 5

INSTRUCTIONS FOR PART III

In Part III you will be given some problems in arithmetical reasoning. After each problem there are five answers, but only one of them is the correct answer. You are to solve each problem and indicate the correct answer on the answer sheet. The following problems have been done correctly. Study them carefully.

Example X: How many apples can you buy for 80 cents at the rate of 3 for 10 cents?

(1) 6 (2) 12 (3) 18 (4) 24 (5) 30

The correct answer to the problem is 24, which is Choice (4).

Example Y: In 4 weeks John has saved $2.80. What have his average weekly savings been?

(1) 35¢ (2) 40¢ (3) 50¢ (4) 70¢ (5) 80¢

The correct answer to the above problem is 70¢, Choice (4).

PART III

21. A clerk multiplied a number by ten when it should have been divided by ten. The answer he got was 100. What should the answer have been?

(a) 1 (b) 10 (c) 100

(d) 1000 (e) 10,000

22. The average salary of three programmers is $95 per week. If one programmer earns $115, a second earns $65, how much is the salary of the third programmer?

(a) $95 (b) $105 (c) $115

(d) $160 (e) $180

23. If a card punch operator can process 80 cards in half an hour, how many cards can she process in a seven and one-half hour day?

(a) 560 (b) 600 (c) 800

(d) 1120 (e) 1200

24. In a programming team of 12 persons, 1/3 are women and 2/3 are men. To obtain a team with 20% women, how many men should be hired?

(a) 4 (b) 6 (c) 8

(d) 12 (e) 20

25. It cost a college 70 cents a copy to produce the program for the homecoming football game. If $15,000 was received for advertisements in the program, how many copies at 50 cents a copy must be sold to make a profit of $8000?

(a) 14,000 (b) 35,000 (c) 46,000

(d) 75,000 (e) 115,000

CANDIDATES FOR THE DATA PROCESSING COURSE STOP HERE.

PART III

26. Express the number 1921.02 in scientific notation.

(a) 19.2102×10^2 (b) 1.92102×10^3

(c) $.192102 \times 10^4$ (d) $192102. \times 10^{-2}$ (e) 1921.02

27. If $f(x) = \dfrac{3x - 1}{2x^2 - 3x - 2}$, then $f(1/3)$ is:

(a) 1 (b) 1/3 (c) 0

(d) $-\frac{25}{9}$ (e) ∞

28. Solve for x, given that $3x + 1 = 7$

(a) 7 (b) $\frac{4}{3}$ (c) 3

(d) 2 (e) none of these

29. Solve the system of equations $x + y = 12$
$x - y = 4$

(a) $x = 8, y = 4$ (b) $x = 16, y = 8$ (c) $x = 9, y = 3$

(d) $x = 12, y = 8$ (e) none of these

30. Solve for x, given that $|3x - 4| \leq 7$.

(a) $-1 \leq x \leq 11$ (b) $\frac{1}{3} \leq x \leq \frac{11}{3}$ (c) $-1 \leq x \leq \frac{11}{3}$

(d) $-11 < x < 3$ (e) none of these

APPENDIX F

a. Application Form

b. Evaluation Questionnaires
#1, Beginning of Session
#2, End of Session
#3, Follow-up on 1969 Participants

University of Southern California

Department of Electrical Engineering

Computer Training Program for High School Students

("PROJECT SOUL")

PARTICIPANT APPLICATION

1. Name: _____
(Please Print) Last First Middle

2. Date of Birth: _____ 3. Soc. Sec. # _____

4. Home Address: _____ Tel. # _____
(Please Print)

5. Name of Guardian: _____
(Please Print)

6. Guardian's address and telephone number if different:

_____

7. High School Attending: _____

8. Classification: ☐ Soph. ☐ Junior ☐ Senior

9. Date you expect to graduate from high school: _____

10. Area(s) of Major interest:
(Please Circle)

Biology, Chem., Engineering, Math., Physics,

Computer Programming, Data Processing,

Agriculture, Architecture, Social Sciences, Other.

Math Courses Taken	Number of Semesters	Grade Received

Science Courses	Number of Semesters	Grade Received

12. Have you taken a course in FORTRAN IV, COBOL PROGRAMMING and/or Keypunch Operation? If so, indicate how long the course lasted (for example, 2 hours per week for 12 weeks).

13. If you have had any of the three courses in No. 12, state the name of the school where you took to course (s).

14. Which of the following is close to your over-all high school grade? (Please circle one):

A, A-, B^+, B, B^-, C^+, C, C^-, D^+, D, D^-.

15. Do you plan to attend college after graduating from high school? If not, what are your aspirations for the future?

16. If you are admitted to "PROJECT SOUL" will you be commuting by car?

17. If you are admitted to "PROJECT SOUL" which of the following two sessions would you prefer to attend?:

SESSION I	SESSION II
☐ June 29 - July 24	☐ July 27 - Aug. 21

NOTE: Whereve= Possible we will try to enroll you in the sessicn of your choice.

QUESTIONNAIRE #1 - BEGINNING OF SESSION

1. How do you feel you will benefit from taking a course in computer programming?

2. When you think about your participation in PROJECT SOUL, what feelings do you have about yourself? Are you proud, scared, indifferent, happy, or what? Please think about your own feelings before answering.

3. What are your greatest personal strengths?

4. What potentialities (talents, skills, abilities) do you think you have which are as yet hidden or undeveloped?

5. What are your greatest personal weaknesses?

6. Briefly describe yourself in two or three paragraphs. You may want to include your personal background, present interests, or important experiences in your life.

7. Do you have in mind a career or goal(s) for your life? If so, please describe.

8. Leaving aside problems of money, your academic record, family responsibilities, etc., what further education would you like to receive and what would you like to study?

9. At the present time, what are your actual plans for the coming year (after you leave this program)?

10. If you were not limited by such things as lack of money and family responsibilities, what would you <u>like</u> to be doing during the coming year?

11. If you were not limited by such things as lack of money and family responsibilities, what would you like to be doing five years from now?

12. If it were necessary for you to get a full-time job immediately, what kind of work would you look for?

13. How would you describe your chances of getting the job you would want if you had to get a full-time job immediately?

_____ Excellent

_____ Good

_____ Fair

_____ Poor

Please explain the reason for your answer.

14. Assuming that you are able to get the necessary education and training, what kind of job, profession or career is of most interest to you? If there is more than one, list them in order of decreasing importance.

15. Which of the following best describes the opportunity you think you will have for individuality and creative work in the computer field?

_____ Excellent opportunity

_____ Good opportunity

_____ Fair opportunity

_____ Poor opportunity

Please give reasons for your answer.

16. What do you think the average salary of a computer programmer is?

17. What are the occupations or occupational skills of your

Mother: _____

Father: _____

Adult brothers: _____

_____

_____

Adult sisters: _____

_____

_____

18. Briefly describe what you now know about computers. For instance, how do they operate, what can they be used for, what are their limitations, etc.?

19. From which of the following have you learned about computers?

_____ Books
_____ Newspapers
_____ Magazines
_____ Television
_____ Radio
_____ School
_____ Family
_____ Friends

Briefly describe your most important source of information about computers.

20. In order to be a good computer programmer (or key punch operator, if applicable), do you feel it is necessary to have

a. a college degree in mathematics?, or

b. a college degree in engineering?

21. Is COBOL generally used for scientific or business purposes?

22. What are "IBM cards" used for?

23. Name as many computer companies as you can.

24. Briefly describe how your daily life would be different if the computer had never been invented.

25. Do you feel that computers should play a larger or smaller role in society?

26. Do you feel that computers in particular and technology in general can be of help in solving some of the major social problems we face? Please explain your answer.

27. What suggestions do you have for improving the process by which students are selected for this program?

28. If you were not receiving an allowance for expenses, would you still be able to attend this program?

29. What is your reaction to this questionnaire?

QUESTIONNAIRE #2 - END OF SESSION

NAME: _____

ADDRESS: _____

TELEPHONE NUMBER: _____

SOCIAL SECURITY NUMBER: _____

BIRTH DATE: _____

GROUP NUMBER: _____

PLEASE FILL IN THE ABOVE INFORMATION AND SEPARATE THIS PAGE FROM THE REST OF THE QUESTIONNAIRE AS IT WILL BE HANDED IN SEPARATELY. PUT YOUR SOCIAL SECURITY NUMBER AND GROUP NUMBER ON THE TOP PAGE OF THE QUESTIONNAIRE. IF YOU DO NOT HAVE A SOCIAL SECURITY NUMBER, PUT YOUR BIRTH DATE ON THE TOP OF EACH PAGE.

THE INFORMATION GIVEN IN INDIVIDUAL QUESTIONNAIRES WILL NOT BE MADE AVAILABLE TO ANY STAFF MEMBERS OF PROJECT SOUL.

The purpose of this questionnaire is to help the PROJECT SOUL staff understand you and your fellow students better. This is the second in a series of questionnaires that you will be requested to complete, both during the summer and after the summer. This questionnaire will provide information to aid the staff in organizing the program to suit your needs and in evaluating the program as a basis for improving it.

In answering the questions, it is important that you express your present opinions and feelings as directly and clearly as possible. Do not write what you think the staff wants to hear, but "tell it like it is." If any of the questions are not clear, feel free to ask the staff for clarification.

1. How have you benefited from your participation in Project Soul? Please include practical benefits such as improved job opportunities and also personal benefits such as new friends or increased confidence.

2. Briefly describe your experience in Project Soul.

3. What previously hidden or undeveloped talents or skills has your participation in Project Soul helped you develop?

4. Please describe your present career objectives or goals for your life. Have they been affected by your participation in Project Soul?

5. If you were able to get as much education as you wanted without having to worry about money, your present academic record, etc., how much further education would you like to receive and what would you like to study?

6. What are your plans for the coming year?

7. If you could ignore lack of money, family responsibilities, etc., what would you like to be doing during the coming year?

8. If you could ignore lack of money, family responsibilities, etc., what would you like to be doing five years from now?

9. If it were necessary for you to get a full-time job immediately, what kind of work would you look for? Has your participation in Project Soul affected your choice?

10. Which of the following best describes your chances of getting the job you would want if you had to get a full-time job immediately?

_____ Excellent

_____ Good

_____ Fair

_____ Poor

Has your participation in Project Soul affected your chances? Please explain your answer.

11. Assuming that you are able to get the necessary education and training, what kind of job, profession or career is of most interest to you? If there is more than one, list them in order of decreasing importance. Has your participation in Project Soul affected your choice?

12. Which of the following best describes the opportunity you think you will have for individuality and creative work in the computer field?

_____ Excellent Opportunity _____ Good Opportunity

_____ Fair Opportunity _____ Poor Opportunity

Please give reasons for your answers.

13. What do you think the average salary of a computer programmer is?

14. In order to be successful in the field of computer science, do you feel that it is necessary to have a college degree in mathematics, science or engineering?

15. Name as many companies as you can that are involved in the field of computer science.

16. Take one social problem, such as pollution and briefly describe how the computer could aid in the solution of that problem.

17. Do you feel that computers should play a larger or smaller role in society? Please explain your answer.

18. Do you feel that the allowance you received for expenses was sufficient?

19. What is your reaction to this questionnaire?

QUESTIONNAIRE #3 - FOLLOW-UP ON 1969 PARTICIPANTS

1. How have you benefited from your participation in PROJECT SOUL?

2. Please describe your present career objectives or goals for your life. How have they been affected by your participation in PROJECT SOUL?

3. What will you be doing in September, 1970? For instance, what school will you be going to and what will you be studying, where will you be working and what will your job be, etc.?

4. What have you actually been doing during the past year (since you completed PROJECT SOUL)?

5. If it were necessary for you to get a full-time job immediately, what kind of work would you look for? How has your choice been affected by your participation in PROJECT SOUL?

6. Which of the following best describes your chances of getting the job you would want if you had to get a full-time job immediately?

   _____ Excellent

   _____ Good

   _____ Fair

   _____ Poor

   Has your participation in PROJECT SOUL affected your chances? Please explain your answer.

7. What are your actual plans for the coming year?

8. What do you actually plan to be doing four years from now? (If you have made no specific plans, indicate what you will most likely be doing.)

9. If you could ignore lack of money, family responsibilities, etc., what would you like to be doing during the coming year?

10. If you were not limited by such things as lack of money and family responsibilities, what would you like to be doing four years from now?

11. Assuming that you are able to get the necessary education and training, what kind of job, profession or career is of most interest to you? If there is more than one, list them in order to decreasing importance. Has your participation in PROJECT SOUL affected your choice?

12. Leaving aside problems of money, your academic record, family responsibilities, etc., what further education would you like to receive and what would you like to study?

13. Do you feel that computers in particular and technology in general can be of help in solving some of the major social problems we face? Please explain your answer.

14. Was your certificate of completion from PROJECT SOUL useful to you in getting employment or in furthering your education. Please explain your answer.

15. What suggestions do you have for improving PROJECT SOUL?

APPENDIX G

Sample Certificates of Completion

UNIVERSITY OF SOUTHERN CALIFORNIA

UNIVERSITY PARK • LOS ANGELES, CALIFORNIA 90007

SCHOOL OF ENGINEERING
DEPARTMENT OF ELECTRICAL ENGINEERING
"PROJECT SOUL"

This is to certify that

has satisfactorily completed the Key Punch Operation Course of the 1970 Computer Science Program for High School Students. This program was sponsored by the National Science Foundation.

Duration: Four weeks (29 June - 24 July)

Carlos Ford-Livene
Program Director

Richard Bellman
Professor of Mathematics,
Electrical Engineering,
and Medicine

UNIVERSITY OF SOUTHERN CALIFORNIA

UNIVERSITY PARK • LOS ANGELES, CALIFORNIA 90007

SCHOOL OF ENGINEERING
DEPARTMENT OF ELECTRICAL ENGINEERING
"PROJECT SOUL"

This is to certify that

has satisfactorily completed the 1970 Summer Computer Science Program for High School Students. This program, sponsored by the National Science Foundation, utilized the FORTRAN IV language on an IBM System/360.

Course: Scientific Programming

Date: July 24, 1970

Carlos Ford-Livene **Program Director**	**Richard Bellman** **Professor of Mathematics, Electrical Engineering and Medicine**

UNIVERSITY OF SOUTHERN CALIFORNIA

UNIVERSITY PARK • LOS ANGELES, CALIFORNIA 90007

SCHOOL OF ENGINEERING
DEPARTMENT OF ELECTRICAL ENGINEERING
"PROJECT SOUL"

This is to certify that

has completed - with honors - the 1970 Summer Computer Science Program for High School Students. This program, sponsored by the National Science Foundation, utilized the FORTRAN IV language on an IBM System/360.

Course: Scientific Programming
Date July 24, 1970

Carlos Ford-Livene
Program Director

Richard Bellman
Professor of Mathematics, Electrical Engineering and Medicine

APPENDIX H

The Work of a Student

```
C       TALLY GRADES AND PERCENTAGES
0001        DOUBLE PRECISION R1,R2
0002        DIMENSION J(600), KIN(600)
0003        REAL MEAN, MED, MED1, MED2
0004        INTEGER HIGH
0005        LAND=0
0006     24 READ (5,16) R1,R2
0007     16 FORMAT (2A6)
0008        WRITE (6,15) R1,R2
0009     15 FORMAT (1H-,2A6)
0010        READ (5,11) M, (J(I), I=1,M)
0011     11 FORMAT (18I4)
0012        IF (M.GT.200) GO TO 82
0013    576 N=6
0014        HIGH=0
0015        MID=0
0016        LOW=0
0017        DO 30 I=1,M
0018        IF (J(I).GT.30) GO TO 66
0019        IF (J(I).LT.20) GO TO 6
0020        HIGH=HIGH+1
0021        GO TO 9
0022      6 IF (J(I).LT.10) GO TO 8
0023        MID=MID+1
0024        GO TO 9
0025      8 LOW=LOW+1
0026      9 N=N+1
0027     30 CONTINUE
0028     66 WRITE (6,12) HIGH, MID, LOW, N
0029     12 FORMAT ('0HIGH', I6, 2X, 'MID', I6, 2X, 'LOW', I6, 2X, 'TOTAL', I6
           1)
0030        PHIGH=FLOAT(HIGH)/FLOAT(N)*100
0031        PMID=FLOAT(MID)/FLOAT(N)*100
0032        PLOW=FLOAT(LOW)/FLOAT(N)*100
0033        WRITE (6,22) PHIGH, PMID, PLOW
0034     22 FORMAT ('0PERCENTAGE HIGH SCORES', F16.9, 4X, 'PERCENTAGE MIDDLE S
           1CORES', F16.9, 4X, 'PERCENTAGE LOW SCORES', F16.9)
0035        CALL SORT (J,N)
0036        WRITE (6,11) (J(I), I=1,N)
0037        TOTAL1=0
0038        TOTAL2=0
0039        DO 36 I=1,N
0040        TOTAL1=TOTAL1+J(I)
0041        TOTAL2=TOTAL2+(J(I))**2
0042     36 CONTINUE
0043        MEAN=TOTAL1/N
0044        STDEV=SQRT(TOTAL2/N-MEAN*MEAN)
0045        WRITE (6,27) MEAN, STDEV
0046     27 FORMAT ('0THE MEAN=',F15.8,5X, 'THE STANDARD DEVIATION=',F15.8)
0047        IF (FLOAT (N)/2.EQ.FLOAT(N/2)) GO TO 4
0048        K=(N+1)/2
0049        MED=J(K)
0050        WRITE (6,100) MED
01       100 FORMAT ('0ODD NUMBER OF TERMS, THE MEDIAN=', F16.9)
0           GO TO 55
```

```
0053            4 K=N/2
0054              MED1=J(K)
0055              MED2=J(K+1)
0056              WRITE (6,103) MED1, MED2
0057          103 FORMAT ('0EVEN NUMBER OF TERMS, ONE MEDIAN=', F15.8, 5X,' 'THE OTH
                 1ER =', F15.8)
0058           55 DO 25 MD=1,N
0059              LAFF=MD+LAND
0060              KIN(LAFF)=J(MD)
0061           25 CONTINUE
0062              LAND=LAND+N
0063              GO TO 24
0064           82 DO 67 MAC=1,LAND
0065              J(MAC)=KIN(MAC)
0066           67 CONTINUE
0067              JAF=LAND+1
0068              J(JAF)=99
0069              GO TO 576
0070              END
```

```
                C     SUBROUTINE SUBPROGRAM FOR SORT
0001                  SUBROUTINE SORT (A,N)
0002                  DIMENSION A(600)
0003                  INTEGER A,T
0004                  K=N-1
0005                  DO 40 I=1,K
0006                  M=I+1
0007                  DO 30 J=M,N
0008                  IF(A(I)-A(J))30,30,4
0009               4  T=A(J)
0010                  A(J)=A(I)
0011                  A(I)=T
0012              30  CONTINUE
0013              40  CONTINUE
0014                  RETURN
0015                  END
```

```
   II     D

HIGH    25  MID     1  LOW     0  TOTAL    26

PERCENTAGE HIGH SCORES    96.15383911    PERCENTAGE MIDDLE SCORES    3.846153259    PERCENTAGE LOW SCORES    0.0
 13  23  24  24  24  24  24  24  24  24  25  25  25  25  25  25  25
 25  25  25  25  26  27  27  28

THE MEAN=    24.42306519     THE STANDARD DEVIATION=    2.52951241

EVEN NUMBER OF TERMS, ONE MEDIAN=    25.00000000     THE OTH ER =    25.00000000
```

II E

HIGH 25 MID 5 LOW 0 TOTAL 30

PERCENTAGE HIGH SCORES 83.33328247 PERCENTAGE MIDDLE SCORES 16.66656494 PERCENTAGE LOW SCORES 0.0
14 19 19 19 19 20 20 21 21 21 21 21 22 22 22 23 23 23
23 24 25 26 26 26 26 26 26 26 28 28

THE MEAN= 22.66665649 THE STANDARD DEVIATION= 3.16586494

EVEN NUMBER OF TERMS, ONE MEDIAN= 22.00000000 THE OTH ER = 23.00000000

II F

HIGH 8 MID 17 LOW 0 TOTAL 25

PERCENTAGE HIGH SCORES 31.999984741 PERCENTAGE MIDDLE SCORES 67.999984741 PERCENTAGE LOW SCORES 0.0
14 15 15 15 16 16 17 18 18 18 18 18 18 18 19 19 19 20
21 23 23 23 23 24 24

THE MEAN= 18.87998962 THE STANDARD DEVIATION= 2.97084618

ODD NUMBER OF TERMS, THE MEDIAN= 18.00000000

I A

HIGH 14 MID 3 LOW 1 TOTAL 18

PERCENTAGE HIGH SCORES 77.777770996 PERCENTAGE MIDDLE SCORES 16.666656494 PERCENTAGE LOW SCORES 5.555554390
0 12 14 16 22 22 24 24 25 25 26 26 26 27 27 28 28 28

THE MEAN= 22.22221375 THE STANDARD DEVIATION= 7.13020611

EVEN NUMBER OF TERMS, ONE MEDIAN= 25.00000000 THE OTH ER = 25.00000000

I B

HIGH 20 MID 5 LOW 0 TOTAL 25

PERCENTAGE HIGH SCORES 79.999984741 PERCENTAGE MIDDLE SCORES 19.999984741 PERCENTAGE LOW SCORES 0.0
16 16 17 17 19 21 21 21 22 22 22 22 22 22 23 23 23 24
24 24 24 24 24 24 26

THE MEAN= 21.71998596 THE STANDARD DEVIATION= 2.67625713

ODD NUMBER OF TERMS, THE MEDIAN= 22.000000000

I C

HIGH 19 MID 5 LOW 0 TOTAL 24

PERCENTAGE HIGH SCORES 79.166656494 PERCENTAGE MIDDLE SCORES 20.833328247 PERCENTAGE LOW SCORES 0.0
18 19 19 19 19 20 20 21 21 21 21 22 23 23 23 24 24 24
25 26 26 26 26 26

THE MEAN= 22.33332825 THE STANDARD DEVIATION= 2.62472057

EVEN NUMBER OF TERMS, CAE MEDIAN= 22.00000000 THE OTH ER = 23.00000000

APPENDIX I

Rosters

KEYPUNCH CLASS

Name	High School
Scharlyn Deloris Boykin	Los Angeles
Gladys Edna Cargo	Locke
Dennis El Ghezounia Drissi	Los Angeles
Donna Marilyn Fujishima	Washington
June M. Harris	Dorsey
Joyce McNeal	Crenshaw
Patricia Ann Oliver	Washington
Rosita Ramon	Lincoln
Sheryl Redonda Rodgers	Washington
Mary Pin Yep	Belmont

BUSINESS DATA PROCESSING
SECTION A

Name	High School
Albert Allegue	Venice
Thelma Susann Arellano	Garfield
Eloy Aristides Arencibia	Roosevelt
Gar Jiunn Chan	Los Angeles
Louise Y. Ito	Los Angeles
Gwendolyn Jenkins	Venice
Caroletta Johnson	Locke
Harvey Wong Jue	Washington
Elsa Lara	Roosevelt
Elia de los Angeles Lopez	Roosevelt
Georgianna Martinez	Lincoln
Adrene Elizabeth Mason	Washington
Diane Miller	Fremont
Manuel Quezada	Roosevelt
Elizabeth Reed	Jefferson
Maggie A. Sotomayor	Lincoln
Craig Sugimoto	Venice
Steve C. Terakami	Venice
Stephanie Timms	Lincoln
Lancy Luvenia Ware	Washington

BUSINESS DATA PROCESSING
SECTION B

Name	High School
Irma Cindy Banales	Lincoln
Deborah Ann Blackburn	Dorsey
Irene Chris Chavez	Roosevelt
Beverly G. Chu	Dorsey
Ruth Lilia Garcia	Lincoln
Melvin Hightower, Jr.	Fremont
Mike T. Kato	Venice
Beverly Ann Kelly	Dorsey
Veronica Lee Kohn	Jefferson
Pearl L. Lew	Crenshaw
Larry Edwin Marcuse	Venice
Russell Marshall	Venice
Perry Y. Miyake	Venice
Leilani R. Morris	Washington
Jeanny Mei Moy	Manual Arts
Robin Yasuichi Nishimura	Los Angeles
Gary Orme	Centennial
William Peter Pew	Venice
Gloria Ann Stinson	Washington
Anna Vega	Lincoln

SCIENTIFIC PROGRAMMING
SESSION I, SECTION A

Name	High School
Maria Antonia Borges	Roosevelt
David Lee Broughton	Crenshaw
Fred Callis II	Centennial
Jorge Manuel Diaz	Inglewood
Gary Bernard Foster	Jefferson
Sam Masujiro Hamamoto	Roosevelt
Kevin Kenichi Hirata	Roosevelt
Gary William Jackson	Los Angeles
Bennett Thungman Jeong	Roosevelt
Karen Ann Johnson	Hamilton
Octavia Coreen Johnson	Fremont
James Larry King	Centennial
Mark Kuwaye	Dorsey
Vernon Mark Lee	Los Angeles
Joanne Aiko Matsubara	Los Angeles
Mamoru Nakatsui	Roosevelt
Ki Sung Park	Los Angeles
Edward Lee Persley	Centennial
Gail Pon	Crenshaw
Catherine Alice Thombs	Venice
Patricia Jean Townsley	Venice
Joslyn Kelm Wong	Dorsey
Arthur Woo	Los Angeles
Teresa Miu Woo	Los Angeles
Naomi Norma Yamane	Los Angeles

SCIENTIFIC PROGRAMMING
SESSION I, SECTION B

Name	High School
Juan R. Aguilar	Venice
Benton Jun Au	Belmont
Shirley Jean Bates	Crenshaw
Patricia Ann Blackwell	Locke
Mike Joseph Blessing	Venice
Hector Nahum De La Roche	Roosevelt
Gilmore Alberto Gutierrez	Morningside
Alvin E. Johnson	Jefferson
Robert Ray Johnson, Jr.	Centennial
Joseph Lawrence Jones	Washington
Richard Wade Kagawa	Venice
Mike Lee	Lincoln
Darrell Gregory Logan	Fremont
Manuel J. Lopez	Roosevelt
Cynthia Gail McCall	Locke
Maria C. Moreno	Roosevelt
David Alan Munroe	Venice
Mike Ryan Palacio	Hamilton
Edward James Perkins	Locke
Keith Bryant Powe	Jefferson
Lai Tan Quon	Crenshaw
Wai Ching Quon	Roosevelt
Alfonso A. Tautimez	Roosevelt
Miggle Matthew Tugmon, Jr.	Jefferson
Byron Louis Young	Centennial

SCIENTIFIC PROGRAMMING
SESSION I, SECTION C

Name	High School
Janice Yvonne Anthony	Morningside
Stanley Edward Brown	Centennial
Willard Anthony Brown	Jefferson
Marlein Carballosa	Belmont
Manette Dennis	Hamilton
Kirk Raymond Ellis	Centennial
Robert Frank Gay	Jefferson
Debra Lynn Grant	Washington
Marvin Ellis Hollis	Jordan
Richard Hultman	Roosevelt
Richard Osamu Kato	Lincoln
Iannice Maria Lee	Jefferson
Alvita Selene McCray-Holmes	Dorsey
Denise Evonne McDuel	Fremont
Marsha Lynn Munemura	Crenshaw
Jesus Munoz	Lincoln
Rosie Maria Nunez	Roosevelt
Robert Olive	Centennial
Michael Ben Panado	Washington
Rosa Lee Pecot	Centennial
Ted Lee Pendleton	Venice
Lydia Quezada	Roosevelt
Theresa Kimi Saisho	Garfield
Donald R. Tahara	Washington
Darlin Williams	Locke

SCIENTIFIC PROGRAMMING
SESSION II, SECTION D

Name	High School
Ron Ayres	Los Angeles
William Henry Bow	Los Angeles
Allen Wallace Chow	Washington
John Howard Fawcett	Venice
Margaret Rose Freedland	Hamilton
Sylvia Yvonne Gee	Hamilton
Armando Antonio Guzman	Roosevelt
Reggie Euart Johns	Crenshaw
Clifford B. Jones	Belmont
Steven Masami Kamo	Garfield
F. Michael Kocik	Venice
Jimmy T. Koda	Crenshaw
Daryl Lawton	Hamilton
Rebecca Seung-Hye Lim	Los Angeles
Andrew Montealegre	Los Angeles
Leo M. Munoz	Roosevelt
Calvin C. Ng	Lincoln
Dan L. Padgett	Venice
Laura Elaine Singerman	Hamilton
Wayne Siu	Lincoln
Terry Soo Hoo	Los Angeles
Juan Benito Lopez Tienda	Garfield
Phillip Brian Tor	Hamilton
Gina Vestal	Venice
Raymond Chek Woo	Roosevelt

SCIENTIFIC PROGRAMMING
SESSION II, SECTION E

Name	High School
Michael Phillip Brody	Morningside
Rick Allan Brown	Inglewood
Andrea Denel Buchanan	Dorsey
Juanita Cooper	Crenshaw
Leon Mark Felburg	Venice
Kelvin Wellman Gee	Washington
Dina Goldstein	Hamilton
Stacy K. Ino	Venice
Diane Ruth Jinks	Fremont
Larry Vernon Johnson	Crenshaw
Dean Leroy Jones	Washington
Marilyn Jones	Crenshaw
Anita Jean Lofton	Fremont
Fabiola C. Lopez	Belmont
Mark Morales	Roosevelt
Edward Gordon Pell	Hamilton
Harold Vernon Peters	Locke
Danny Roger Revay	Morningside
Eloise Caridad Rivero	Belmont
Emmanuel Dwight Robert	Morningside
Mark Elliott Rutta	Venice
Jose Antonio Salazar	Garfield
Jeffrey Lawrence Soo Hoo	Belmont
Randy Lewis Takemoto	Roosevelt
Marc Edward Vinicky	Morningside
Gary Young	Belmont

SCIENTIFIC PROGRAMMING
SESSION II, SECTION F

Name	High School
Patricia Ann Burnett	Jefferson
Charlesetta Calhoun	Jordan
Robert Lee Cowan	Dorsey
Wilda Dear	Manual Arts
Melva Denman	Jefferson
Petra Olivos Garcia	Belmont
Victoria Ellen Garrett	Washington
Christine Gwendolyn Gridiron	Washington
Ramon Francisco Hernandez	Roosevelt
Gail Marks	Washington
Greenwood T. May	Crenshaw
Brenda Ann Meekins	Jefferson
Rosalba Merida	Roosevelt
Mark Leslie Miller	Venice
James Darrell Mitchell	Locke
Freddie Moore, Jr.	Jefferson
Yolanda Ramirez	Roosevelt
Debbie D. Smith	Dorsey
Yvonne Loyrane Sneed	Crenshaw
Norma Alice Tapia	Roosevelt
Jeffrey Steven Terry	Morningside
Margaret Annette Thomas	Dorsey

APPENDIX J

The Staff

Richard Avery, undergraduate, Laboratory Assistant

Richard Bellman, Ph.D., Principal Investigator

Gary Bloom, M.S., Tutor, Scientific Programming

Carlos Ford-Livene, M.A., Director

Norman Gale, M.S., Instructor, Data Processing

Betsy Gloster, M.A., Secretary-Placement Aide

Bayesteh Kashef, Ph.D., Instructor, Scientific Programming

John Kelder, M.A., Instructor, Keypunch

Jules King, M.A., Instructor, Data Processing

Pradeep Kumar, M.A., Laboratory Assistant

Alberto Leon, undergraduate, Laboratory Assistant

Art Lew, Ph.D., Curriculum Coordinator, Scientific Programming.

Larry Marks, B.S., Evaluation Coordinator

Gideon Nagel, undergraduate, Laboratory Assistant

Raúl Salas, undergraduate, Laboratory Assistant

Richard Schein, undergraduate, Laboratory Assistant

Christine Shoemaker, M.A., Tutor, Scientific Programming

Dan Tuey, M.S., Instructor, Scientific Programming

Michael Wasserman, B.A., Instructor, Scientific Programming

James Woolever, Curriculum Coordinator, Data Processing

Andrew Yakush, M.A., Instructor, Scientific Programming

M. Virginia Zoitl, M.S., Technical Administrator

APPENDIX K

Publicity (Newspaper Clippings)

REMOVED PRIOR TO BEING SHIPPED TO EDRS
FOR FILMING DUE TO COPYRIGHT
RESTRICTIONS.