software age

APRIL 1969


2
5
4
1
3
10
7
9
3
2

software age

APRIL, 1969 Vol. 3–No. 4

CONTENTS **PAGE**

S/A's Resume Center ---------- 6
A New Method for Representing Hexidecimal Numbers ---------- Vartan Khosharian 9
Executive Routine ---------- 13
Checkmate ---------- 17
Software Modulization ---------- D. V. Mathusz 19
Trouble Tran's Adventures in Fortran ---------- 28
Visiting Boston During SJCC ---------- 32
Computers in the Classroom ---------- Charles H. McCoach 42
New Products ---------- 50
Better Books ---------- 52
The Marketplace ---------- 58
S/A Confidential Resume Form ---------- 61
Index to Advertisers ---------- 62

Publisher ---------- David W. French
Managing Editor ---------- Jean S. Logan
Advertising Manager ---------- John F. Neiman
Traffic Manager ---------- Betsy Pavkovich
Business and Circulation ---------- Elspeth Daly
Resume Control ---------- Anne Wildeman

Circulation of this issue more than **125,000**

SOFTWARE AGE is published monthly by PRESS–TECH, Inc.
2211 Fordem Avenue, Madison, Wisconsin 53701
Phone: (608) 249–0128 or (800) 356–8141

Subscription free to qualified readers. Others, $10/yr. Individual copies, $1. Foreign subscriptions, $15/yr. For change of address please return your address label from the magazine.

Main Sales Office: David W. French or John F. Neiman, 2211 Fordem Avenue, *Madison, Wis.*—Telephone (608) 249–0128 or (800) 356–8141.

New York, N.Y.: Telephone (212) 697–5356.

Los Angeles, Calif.: Richard Faust, 9800 S. Sepulveda Boulevard—Telephone (213) 776–0100.

San Francisco, Calif.: Ernesto Montano, 595 Buckingham Way, Suite 226—Telephone (415) 731–8240.

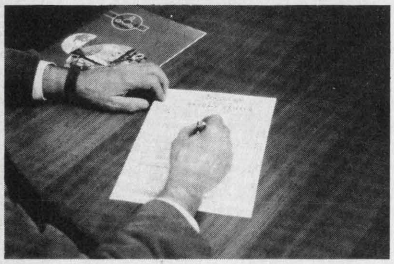
Philadelphia, Pa.: Richard D. Clemmer, 27 Acoma Lane, Collegeville, Pa.—Telephone (215) 489–9141.

Boston, Mass.: Telephone (617) 542–1466.

Controlled circulation postage paid at Madison, Wis.

Can an EDP man attending SJCC find that better job without letting the world know he's looking? SURE HE CAN!

SOFTWARE AGE Magazine's FREE Resume Center protects your privacy . . . speeds your search . . . costs you nothing!



1. Fill out the special resume form from magazine or at Center.

2. Our Dennison operator will photocopy the resume free and return *all* copies to you.

3. You place your resume in the company lock boxes of your choice.

Your job is simple. You follow the same procedure at the center that you do in S/A—but on a highly accelerated basis. Fill out the special resume form at the S/A RESUME CENTER. Our Dennison operators will provide you with as many *free* photo copies of your resume as you require.

You personally deposit them in the lock boxes of those companies which interest you. The resume form includes blanks where you can enter your hotel, telephone number and room number. Only the participating company representative has access to the lockbox. At his convenience, he can contact you at your hotel to arrange an interview.

The entire operation of the S/A RESUME CENTER is designed to provide speed, convenience, simplicity and maximum confidence with no middlemen involved. You determine exactly the companies you want—no others will see your resume or know of your interests.

The S/A RESUME CENTER is FREE. There is no charge to you or the company for this valuable service. The resume center is sponsored solely by the advertisers in SOFTWARE AGE magazine.

software age
resume center

Boston, Statler Hilton Hotel, Park Square

Noon to 7 P.M., Wednesday, May 14

9 A.M. to 7 P.M., Thursday, May 15 — 9 A.M. to 3 P.M., Friday, May 16

SOFTWARE AGE Magazine, 2211 Fordem Ave., Madison, Wis. 53701 John F. Neiman, Advertising Manager

S/A RESUME CENTER

Special Resume Form

For Use at

The Statler–Hilton Hotel, Boston, May 14–16, 1969

Do not use this form to answer advertisements in this issue. *See page 61*

BE SURE TO CHECK YOUR HOTEL FOR MESSAGES

NAME ____________

HOTEL ____________

TELEPHONE NUMBER ____________

ROOM NUMBER ____________

BEST TIME TO CALL ____________

JOB DESIRED: ____________

List computer hardware knowledge (names of systems, tape, disk, terminals, etc.): ____________

Programming specialties and years of experience (commercial, scientific, theoretical, experimental, analog, etc.): ____________

Systems programming on which you have had development experience (compilers, assemblers, executives, monitors, O.S., etc. Indicate for what computer): ____________

Programming languages used and extent of experience (COBOL, FORTRAN, etc.): ____________

Applications programmed (aerospace, banking, insurance, math subroutines, compilers, etc.): ____________

Systems analysis experience (card design, flow charting, operation analysis, etc.): ____________

EDP management experience (include years and number of people reporting to you): ____________

SALARY: ____________ (current) ____________ (desired) **DATE OF AVAILABILITY:** ____________

EDUCATION: Indicate major as well as degree unless self-explanatory.

Degrees			
Years			
Schools			

EMPLOYMENT: Indicate present employment and previous jobs below.

Employer			
City			
Years	to	to	to
Title or Function			

Name ____________ Age ____________

Home Address ____________ Home Phone ____________

____________ (city) ____________ (state) ____________ (ZIP code) U.S. Citizen? ____________

Security Clearance ____________ Location Preference ____________

Marital Status ____________

Military Status ____________

BE SURE YOU HAVE NOTED YOUR HOTEL, ROOM AND TELEPHONE NUMBER IN THE BOX—UPPER RIGHT CORNER OF THIS PAGE

software age

MAGAZINE

2211 FORDEM AVE., MADISON, WIS. 53701

TEAR OUT THIS RESUME PAGE AND BRING IT TO THE S/A RESUME CENTER

The Statler–Hilton Hotel—Boston

This is a special resume form to be used only if you plan on being in Boston during the SJCC Show. Fill out the reverse side and bring it with you to the S/A RESUME CENTER, May 14–16, The Statler–Hilton Hotel, Boston, Mass.

COMPANIES PARTICIPATING IN THE S/A RESUME CENTER

	Page
☐ **Auerbach Corp.** (Philadelphia, Pennsylvania)	3rd Cover
☐ **Blue Cross–Blue Shield** (Chicago, Illinois)	16
☐ **Callahan Center for Computer Personnel** (Philadelphia, Pennsylvania)	
☐ **Career Center** (New York, New York)	47
☐ **Collins Radio Co.** (Dallas, Texas)	40
☐ **Computer Personnel Consultants, Inc.** (Chicago, Illinois)	45
☐ **Comress** (Washington, D. C.)	51
☐ **Conductron Corp.** (St. Charles, Missouri)	22
☐ **Data Tech** (Cherry Hill, New Jersey)	May Issue
☐ **Electronic Associates, Inc.** (West Long Branch, New Jersey)	39
☐ **First National Bank of Chicago** (Chicago, Illinois)	12
☐ **Foxboro Co.** (Foxboro, Massachusetts)	May Issue
☐ **General Electric Co., Information Service Dept.** (Bethesda, Maryland)	37
☐ **General Electric Co., Process Computer Dept.** (Phoenix, Arizona)	May Issue
☐ **Hartford Insurance Group** (Hartford, Connecticut)	25
☐ **Honeywell, Inc., Electronic Data Processing Div.** (Waltham, Massachusetts)	18
☐ **Hughes Aircraft Co., Ground Systems Group** (Fullerton, California)	May Issue
☐ **IBM Corp.** (Gaithersburg, Maryland)	29
☐ **Interstate Staffing, Inc.** (Bala Cynwyd, Pennsylvania)	57
☐ **Itek Corp.** (Burlington, Massachusetts)	May Issue
☐ **Leeds & Northrop Co.** (North Wales, Pennsylvania)	
☐ **Litton Industries—Mellonics Systems Development Div.** (Sunnyvale, California)	60
☐ **Lockheed Missiles & Space Co.** (Sunnyvale, California)	30
☐ **Montgomery Ward Data Center** (Chicago, Illinois)	May Issue
☐ **National Cash Register Co.** (Dayton, Ohio)	May Issue
☐ **National Manpower Register** (New York, New York)	2nd Cover
☐ **Raytheon Co., Equipment Div.** (Waltham, Massachusetts)	4th Cover
☐ **RCA Corp.** (Camden, New Jersey)	3
☐ **Scientific Data Systems** (El Segundo, California)	36
☐ **Sikorsky Aircraft** (Stratford, Connecticut)	15
☐ **Singer–General Precision, Inc.** (Sunnyvale, California)	5
☐ **Source EDP** (Chicago, Illinois)	59
☐ **U. S. Army Aviation Systems Command** (St. Louis, Missouri)	53
☐ **Univac Corp.** (Philadelphia, Pennsylvania)	26 & 27
☐ **Westinghouse Information Systems Laboratory** (Pittsburgh, Pennsylvania)	41
☐ **Xerox Corp.** (Rochester, New York)	31

TOTAL for duplicating ___________

A New Method of Representing Hexidecimal Numbers

by Vartan Khosharian

Khosharian is a specialist in command and control computer systems with Lockheed Electronics Company, Houston Aerospace Division, Manned Spacecraft Center. He is a project leader for system programming of the computer aided communication analysis system used for the Apollo 8 Mission.

He has programmed ten different computers over the past twelve years, and holds a B.A. degree in mathematics from Arizona State University, Tempe, and has done graduate work at U.C.L.A.

The basic numeration system employed in modern day digital computers is binary. Due to the alienation of the binary system with respect to the decimal system, the octal numeration system is utilized to impose a direct relation to binary numbers and yet allow a comfort level for the decimal numbers. In computer operation the binary digits are grouped in sets, called "words". It is convenient to reduce word size by employing the relation that exists between binary and octal notation. Two to the third power, for instance, may be represented in the octal system by eight to the first power. Thus the direct octal representation of a computer word is established and word size is reduced. As to the exact number of binary digits that should be used in a computer word, the computer designers are influenced by many factors such as cost, speed, processor logic, etc.

There are two basic types of digital computers, BCD and binary. The BCD (Binary Coded Decimal) has always been considered as the business problem solving orientated machine, whereas the binary computer has been considered as the specialist for scientific problem solving. The basic differences between these two computers are in the word organization and the processing logic, such as the adder unit. The BCD computer represents decimal digits by logically grouping a minimum of four binary digits in a computer word. The processing logic performs arithmetic operations in serial order by each decimal digit binary representation. The binary computer represents the exact value expressed in the number of digits in a computer word. The processing logic performs arithmetic operations in parallel order on each binary digit.

At the beginning of the computer era, computer manufacturers produced BCD and binary computers as two separate and distinct machines. But as computers became more popular in the business world, with the rapid increase in information processing, the practice of manufacturing two different types of computers was abandoned. Manufacturers developed a computer to handle both business and scientific problems. This meant that the computer word organization would have to lean more toward the BCD representation. As a result, the numbers of binary digits in a computer word is divisible by four and the external representation of these four binary digits is the hexadecimal numeration system. Unfortunately, the decimal system provides only ten symbols with which to express numbers. The Roman letters A, B, C, D, E, F are used to express the additional six symbols needed for a hexadecimal system.

It is very cumbersome to perform arithmetic operations utilizing the letters A, B, C, D, E and F. A more practical approach is to use a uniform system of pure numbers, thus eliminating the necessity for conversion to decimal and/or octal.

According to the scales of notation, any given hexadecimal number is expressed by the following algorithm:

$$N_{16} = a_0 16^0 + a_1 16^1 + a_2 16^2 + a_3 16^3 + \ldots\ldots + a_n 16^n$$

where $a_i < 16$ for $i = 0, 1, 2, \ldots n$

The octal number is expressed as follows:

$$N_8 = b_0 8^0 + b_1 8^1 + b_2 8^2 + b_3 8^3 + \ldots\ldots + b_n 8^n$$

where $b_i < 8$ for $i = 0, 1, 2, \ldots n$

Given the following bi-octal number algorithm,

$$N_{2\times 8} = (C_0 8^0 + d_0 8^1)\ 2^0 + (C_1 8^1 + d_1 8^2\ )\ 2^1 + d_2 8^3)\ 2^2 + (C_3 8^3 + d_3 8^4)\ 2^3 + \ldots\ldots + (C_n 8^n + d_n 8^{n+1})\ 2^n$$

where $c_i < 8$ and $d_i = 0$ or 1 for $i = 0, 1, 2, \ldots n$

if $d_i = 1$ then $N_{2\times 8} = N_{16}$ where $8 \leqq a_i < 16$,

and

if $d_i = 0$ then $N_{2\times 8} = N_{16}$ where $0 \leqq a_i < 8$,

it can be seen that the bi-octal representation expresses any hexadecimal number based on the value of the coefficient d_i.

In order to implement the bi-octal system, the following scheme is suggested for representing a hexadecimal digit:

0, 1, 2, 3, 4, 5, 6, 7, 0, $\bar{1}$, $\bar{2}$, $\bar{3}$, $\bar{4}$, $\bar{5}$, $\bar{6}$, $\bar{7}$
where the bar ($^-$) denotes the value of the coefficient, $d_i = 1$ and the absence of the bar denotes $d_i = 0$. The correlation between the presently used hexadecimal number representation and proposed bi-octal number representation is given in the following table:

Hexadecimal	*Bi-octal*
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	$\bar{0}$
9	$\bar{1}$
A	$\bar{2}$
B	$\bar{3}$
C	$\bar{4}$
D	$\bar{5}$
E	$\bar{6}$
F	$\bar{7}$

It should be remembered that place value in the bi-octal system is identical to that of the hexadecimal system. For example, the bi-octal symbol $\bar{2}$ in the second place, as in the numeral "$\bar{2}$0", indicates ten sets of sixteen or the number 160 when converted to decimal notation. The bi-octal numerals are considerably more convenient to use than the conventional hexadecimal notation with its six letter symbols. Certain rules have been established for the operations of addition, subtraction and multiplication, in the bi-octal system. These operations are described below. The methodology outlined is believed to offer a simpler means of performing these computations than does the hexadecimal system, particularly for those people who are not well versed in computer arithmetic.

The operation of addition is performed in the bio-octal system by adding, as in the decimal system, and expressing the sum modulo 8. The "carry value", if any, represents sets of sixteen, as in the hexadecimal system.

The following rules are established for addition in the bi-octal system:

1. If the two digits that are to be added do not have the bar symbol and the resultant produces a value equal to or greater than eight, the resultant is the value of the remainder with a bar. (i.e. $5 + 4 = \bar{1}$)

2. If one of the two digits to be added has the bar symbol and the resultant produces a value less than eight, the resultant is the value of the two added digits with a bar. (i.e., $\bar{3} + 3 = \bar{6}$)

3. If the two digits to be added both have bars and the resultant produces a value equal to or greater than eight, the resultant is the value of the remainder with a bar and a one carry for the next digit. (i.e., $\bar{6} + \bar{4} = 1\bar{2}$)

The following conditions produce a one carry:

1. If one of the two digits to be added has the bar symbol and the resultant produces a value equal to or greater than eight, the resultant is the remainder with a one carry. (i.e., $\bar{6} + 4 = 12$)

2. If the two digits to be added both have bars and the resultant produces a value less than eight, the resultant is the sum, Modulo 8, with a one carry. (i.e., $\bar{4} + \bar{3} = 17$)

3. If the two digits to be added both have bars and the resultant produces a value equal to or greater than eight, the resultant is the sum, Modulo 8, with a bar and a one carry for the next digit. (i.e., $\bar{6} + \bar{4} = 1\bar{2}$)

To multiply two digits in the bi-octal system, express the product Modulo 8. The carry value is determined by:

1. If the multiple of 8 is even, take one half of that number.

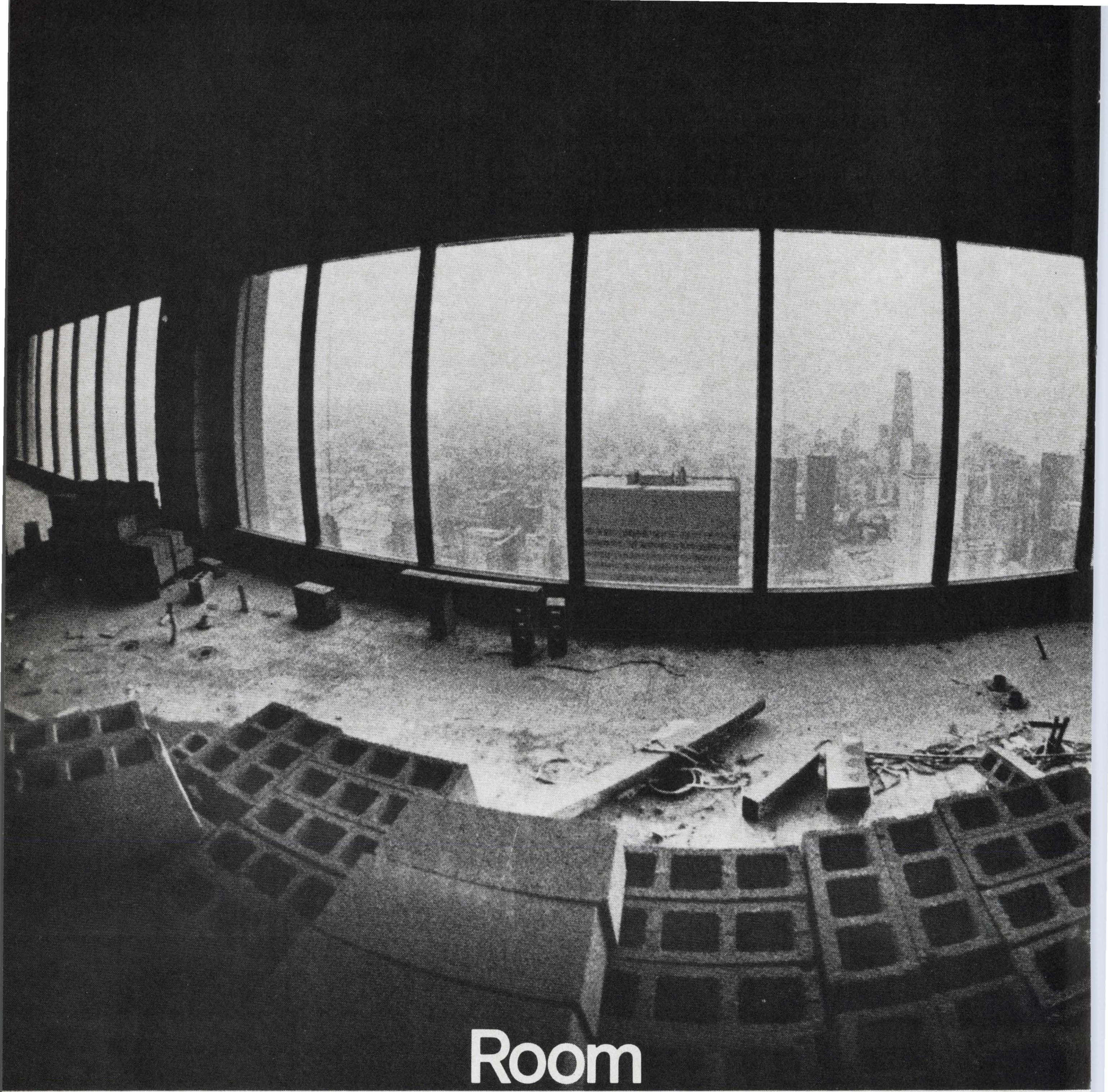
2. If the multiple of 8 is odd, take one half of one-less-than the multiple for the carry value.

If the number of Modulo 8's is odd, the resultant digit has a bar.

$$(5 \times 5) = 1\bar{1}$$
$$(7 \times 7) = 31$$

3. If the two digits to be multiplied have bars, multiply the two digits modulo 8, and take half the number of mod 8's, as above, for the carry. If the number of mod 8's is odd, the resultant has a bar. Now, multiply the bar value of one digit by the number value of the other digit, taking half the resulting value as the carry. If the number is odd,

(Continued on page 24)



executive routine

Herbert E. Martenson

Mr. Martenson, in 18 years of industry experience, has acquired a broad exposure to a variety of manufacturing, distribution, personnel and accounting situations. These experiences, coupled with an Industrial Engineering education, cover the spectrum of computer, administrative and managerial activities. He is presently associated with Martenson Associates, a software firm with offices in Crystal Beach, Florida, and Park Forest, Illinois.

He has served as Director of Computer Services for the Signode Corporation and held a similar position at the corporate level with Greyhound. He also served as systems consultant to the smaller subsidiaries of these companies. In addition, he has been associated with R. R. Donnelley and R.C.A. in managerial capacities doing systems work.

In the course of these activities, he has had first hand knowledge of the points raised in this article. He offers a view of the computer field as seen by a practitioner.

It's curious how some managements react to the man in charge of a computer project. I have known cases of genuine fear develop on the part of executives and managers as they find the people who developed the computer program more knowledgeable about a system than they are. The change in dependency status can become a real problem since human emotions are involved. Capable computer men are often side tracked or "resigned" to keep them in line. Subsequently, new ideas emanating from the computer group are greeted with antagonism and suspicion.

As a computer pro, it's vital to keep your line associates informed of what you're doing and why. It takes time, it's often tedious to explain things over and over again but it's the only way to gain true acceptance in many organizations. The man who is a line manager of a horde of people doing simple things, finds little solace in the fact that HIS employees are to be displaced and that those people remaining will need retraining—including himself. The problem is how to motivate people to work themselves out of their job. This level of altruism is difficult to attain, if you will permit an understatement.

I recall one situation involving Accounts Receivable. The company was in trouble. Collections were slow. Unearned discounts were not detected when taken. Trial balances were a physical impossibility due to the paperwork volume. Mechanization was clearly an economic necessity. A rush systems design project was authorized. Programmable control methods were developed with the "cooperation" of the line supervisors. And then the squabbling began. One year later, the project was incomplete. Everyone was unhappy about the inordinate delay. The basic A/R problem was more critical than ever and despite many opportunities for meaningful progress, discussions were still going on regarding the adequacy of the approach to be used.

An imaginary story? No, it's true. It happened even though top management "wanted" a different result. The scapegoat was Data Processing. The culprit was top management (not middle management). The line managers acted as normal human beings within the limits of the authority vested in them. Dedication to (absent) goals was missing. Also missing was attention and education by upper management to overcome the fears of the line managers and the clerks involved. It probably was a toss-up where the most money was lost—in an old, inadequate A/R system or in the cost of spinning wheels to get a better one.

How can situations such as the foregoing be avoided? In some cases it's impossible, but the following guides can be of some help.

1. Get agreement on exactly what is to be accomplished first. Don't ever discuss a time-table until the objectives are set down in an unambiguous form. In the statement of the objective, outline auxiliary problems in need of resolution.
2. Don't accept edicts (unless you have a computer that responds to them). You can't win. If you are told to get something done by a deadline, be certain it's feasible before acceding to the demand. Often there are factors far beyond your control that are involved in making a time-table effective. Unless you have commitments from everyone involved to get their part done, you will be left holding the bag.
3. Use some form of periodic project control technique consistent with the scope of the project. Manual methods are OK for small jobs. Mechanized systems are available for use on larger projects. (Look into the system used by Bell & Howell, Skokie, Ill. It works and it's simple to use.)
4. Report in writing, but avoid pointing fingers or finding fault with others involved in the project. Be objective! Repeat yourself. People often forget the salient parts of a previous report.
5. Discuss the future schedule with those involved in executing it to be sure no previously unforeseen problems have developed. If a change in schedule is required, document the reasons and change the targets appropriately. Don't keep hoping for a miracle. Avoid surprises.

Despite the fact that computers are old hat to the EDP practitioners, the mystique surrounding them remains with lay people. Fear of the unknown and untried is a normal human reaction that must be expected of others. Myths and misconceptions are also pitfalls to be dealt with. Remember to build time into your project completion schedule for these types of things, and you will be better able to satisfy the EDP needs of your organization. ■

CHECKMATE

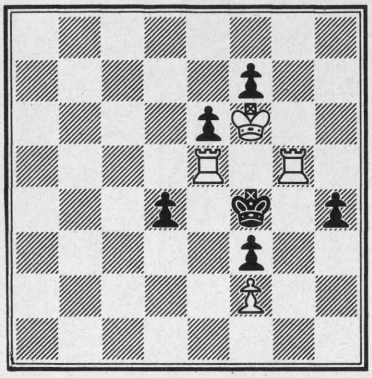
by
GEORGE N. VASSILAKIS
TRW SYSTEMS GROUP

Problem 7

William A. Shinkman

The White Rooks

1910



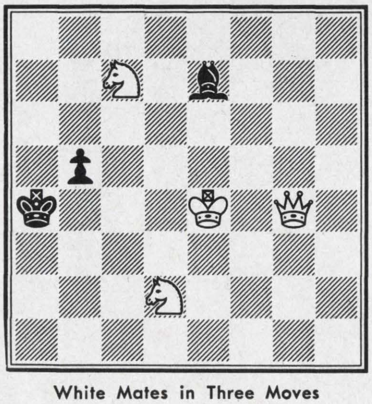
White Mates in Three Moves

Problem 8

Otto Wurzburg

American Chess Bulletin

May–June, 1955



White Mates in Three Moves

Solutions:

Problem 5: 1Q–Q2 threat 2Q–R2; if 1 . . . K–Q3 then 2 Q–R2 mate; if 1 . . . QxQ then 2 N–B3 mate.

Problem 6: 1 R–B6, N–R6; 2 RxNP, PxR; 3 RxN mate; if 2 . . . N–N4; 3 R–N8 mate; if 1 . . . P–R4; 2 KRxPch, PxR; 3 R–R6 mate.

COMPUTERIZED CHESS

If your field is Computer Science and you have difficulty choosing a topic for a thesis why not try "Computerized Chess." It is a very interesting and challenging topic. The following letter from Mr. T. A. Marsland may give you the inspiration you need.

"Although it is nearly two months since I read your item about chess-playing computer programs in the Checkmate column of Software Age, it is only now that I have stirred myself to bring my own program to your attention. This basic program was begun in March 1968, as an extra activity during my post-doctoral year at the University of Washington, and has continued on a part-time basis since that time. The program is written in Burroughs Extended ALGOL for execution on a B-5500. Currently the system is implemented on a commercial timesharing computer operated by COMNET Inc. in Orange, New Jersey.

"The program is accessed through a standard teletypewriter (Model 33 or 35). A set of dynamic control options can be used to direct such information as board position, legal move list and debugging data to the terminal. At any position a static analysis is performed to select up to seven moves for more detailed examination; that list may be further pruned before submitting the remaining plausible moves to the look-ahead procedure.

"The enclosed game was played against another machine, at the University of California, Davis, in a telephone match, it was my program's first victory. The opposition had a simpler scoring function but were using 3 ply (1 full move) look-ahead. At that time the B-5500 program did not have the look-ahead feature implemented, but the static analysis was fairly detailed. When playing against itself, my basic scoring function plays at an average rate of 8 secs. (CPU) per move.

"I realize that this system is not of the caliber of the MIT program, however, I expect great improvements during the year. Meanwhile, I am interested in arranging matches against other programs and hope that you will be able to put me in contact with the authors of those programs that you have heard about."—**T. A. Marsland**

COMPUTER VS. COMPUTER

Marsland's computer was a Burroughs B-5500 at Bell Telephone Laboratories in Holmdel, New Jersey. His opponent at the University of California used an IBM 7044.

As you see, white played an interesting opening and a very impressive end game. Both computers made blunders in the middle game. White missed a king–queen fork on the sixth move but found it on the 7th, and after winning his opponent's queen on the 8th used his own queen to clean up the board. After loosing his queen, black behaved like a 10-year-old who knows he is loosing and starts giving away pieces to end the game quicker.

The important thing, of course, is that computers are learning how to play chess and before long we may have chess masters with names such as IBM 360 mod 101, CDC 8600, SDS Sigma 10, RCA Spectra 90, etc.

	B-5500 (White)	IBM 7044 (Black)		B-5500 (White)	IBM 7044 (Black)
1	P–K4	P–Q4	17	P–K5	NxR
2	B–N5	B–Q2	18	PxPch	K–K1
3	N–QB3	P–Q5	19	RxN	PxP
4	BxB	QxB	20	PxP	N–Q4
5	N–Q5	Q–K3?	21	Q–B6ch	K–B2
6	Q–B3?	P–QB3	22	QxBP	NxB
7	N–QB7ch	K–Q2	23	NxN	P–QR3
8	NxQ	PxN?	24	P–KN3	P–R3
9	QxB	P–KN4	25	Q–B6	P–QR4
10	N–K2	P–B4	26	QxPch	K–N2
11	P–Q3	P–N5	27	QxNPch	K–B2
12	B–B4	N–KB3	28	Q–K6ch	K–N2
13	QxR	N–B3	29	QxKPch	K–R1
14	QxR	N–N5	30	N–N6ch	K–N1
15	P–QB3	NxQPch	31	Q–K6ch	K–R2
16	K–Q2	NxBP	32	Q–KB7	checkmate

As the size of systems grows, the implementation time grows and is laden with revisions. As the complexity of the system grows, the processing algorithms frequently become analysis and research efforts. The system design then becomes highly fluid and dependent upon these R&D outcomes. Management and the working analysts need some unit of systems design effort that can be scheduled, designed, and documented for independent evaluation. The Software Module can be designed and documented in just this fashion. In this form, the module can be the means of coordinating the various programs at an installation into something like a cybernetic whole. This paper addresses the problem of module design in a systems design context, including documentation and the management problems of visible progress, political priority and education.

Software Modulization

Part I

By Donald Ventner Mathusz

1.0 Overview: The Ad Hoc Systems Design Syndrome

As long as one small group can do the analysis, design and programming in a short period of time, system environment shifts and their system revisions can be avoided and the implementation job remains relatively simple. In this situation documentation will probably be left to slack periods (almost never). Critical data and timing interfaces between packages, and programs will be left to the memory of a few key people. Unfortunately, as systems groups evolve, slack man-hours grcw less and systems grow in size and complexity while manpower turnover increases.

As the size and complexity of software systems grow, the basic job of achieving full implementation grows beyond bounds. All manner of necessary and desired revisions are proposed to the pieces of the system implemented, in process and under study. The management of implementation becomes one of great effort merely just to keep the present state of implementation updated revisionwise, while the total system implementation progress is diluted.

Under these circumstances, the management problem can easily become ensnarled in such vast volumes of details, that the resource allocation of directed man-hours in analysis, design and programming becomes an almost impossible job. (This has been proven at a number of computer groups.) Some of the developments contributing to this problem are listed below:

(a) Changes in hardware, machine language, core size and arrangement that occur over long time periods.

(b) Documentation: non-existent, deficient, or non-standardized.

(c) Interface programs and/or information flows not explicitly documented.

(d) System shifts in time, i.e., changes in needs before the original system is complete such that major revisions occur before the initial project completion.

(e) Long-complex programs whose logic is highly interdependent, therefore difficult to follow and more economic to rewrite than revise. (Program probably includes "trick" statements that only operate to the programmer's unique logic on particular compilers.)

(f) Lack of input data quality control, e.g., range of acceptable values, negative not allowed checks et al. not found in operation or documentation.

(g) Visible progress (lack of): for political reasons it is wise to be able to display progress and for project control purposes we need meaningful milestones to monitor.

(h) Projects become increasingly complex and algorithms must be researched. The final system design is now dependent upon the results of these studies.

2.0 Management Control

Management must concern itself with the problems of implementation progress, resource allocation to this end, and the political problems of funding, payoff rate, and priority of goals. This entails defining independent unit work packages and their resource and funding input re-

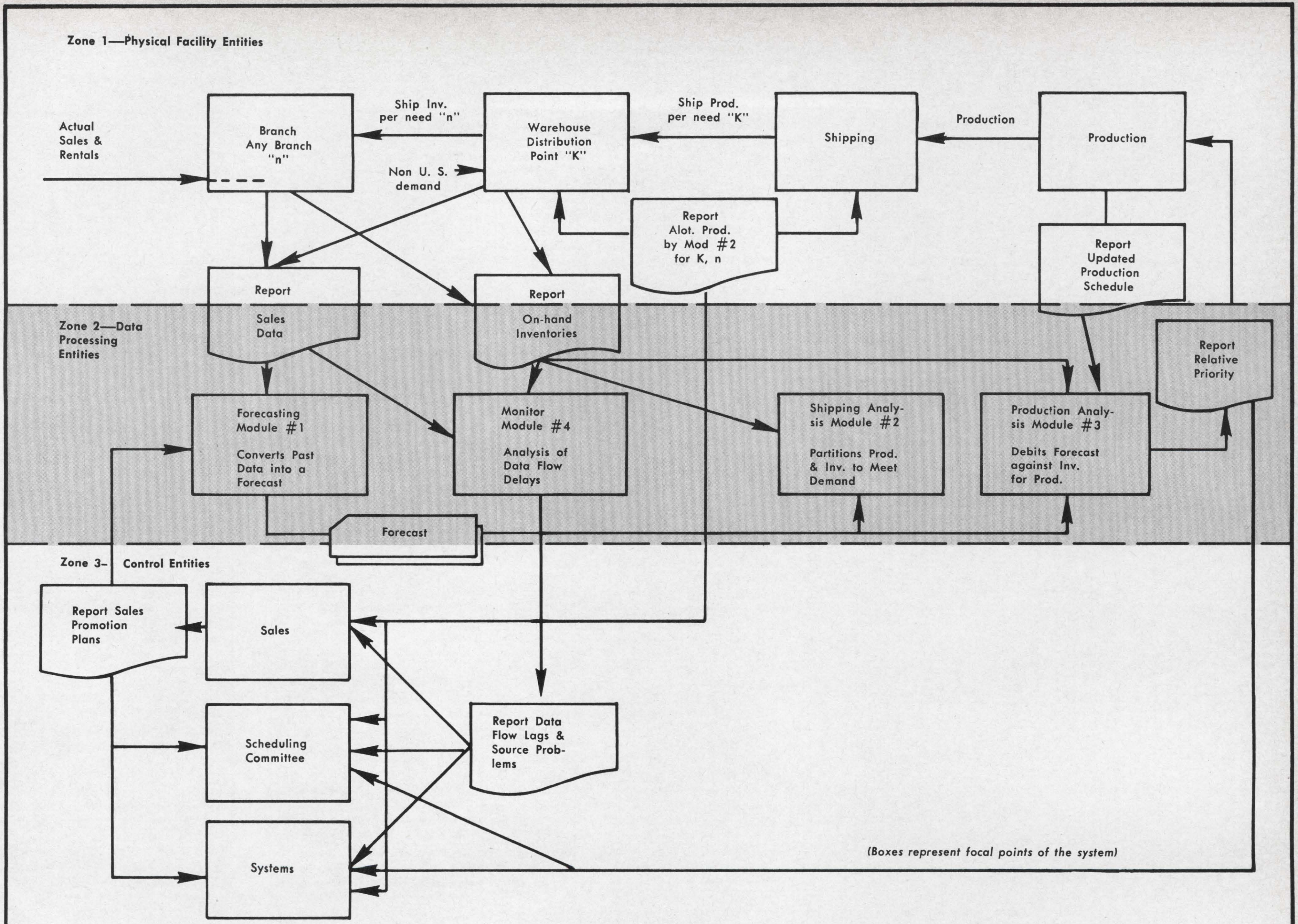

Zone 1—Physical Facility Entities
Actual Sales & Rentals
Branch Any Branch "n"
Ship Inv. per need "n"
Warehouse Distribution Point "K"
Non U. S. demand
Ship Prod. per need "K"
Shipping
Production
Production
Report Alot. Prod. by Mod #2 for K, n
Report Updated Production Schedule
Report Sales Data
Report On-hand Inventories
Zone 2—Data Processing Entities
Report Relative Priority
Forecasting Module #1 Converts Past Data into a Forecast
Monitor Module #4 Analysis of Data Flow Delays
Shipping Analysis Module #2 Partitions Prod. & Inv. to Meet Demand
Production Analysis Module #3 Debits Forecast against Inv. for Prod.
Forecast
Zone 3- Control Entities
Report Sales Promotion Plans
Sales
Scheduling Committee
Systems
Report Data Flow Lags & Source Problems
(Boxes represent focal points of the system)

Figure 1

quirements and relating these to the goal achievements. The definition of the work package is the key to the problem of scheduling, costing and achievement measurement. It must be done such that it is, in fact, an independent entity that can be designed, programmed, and operated independently of other work in progress. This independent entity is defined as the software module. Its processing function(s) input and output I/O are defined at the system design stage in general. In particular, the I/O is defined rigorously such that its exact data needs and products are known for use in a multiprogramming team effort. In this manner, several programming teams can be working on different modules of the program efficiently.

Scheduling of the effort, using the modules as major milestones or PERT net nodes, becomes effective. To accomplish the system, there will appear some necessary prerequisite relationships between set of modules. But also, there will be some choice within module sets as to implementation order. The effects of smoothing resource expenditures will mean that the theoretical maximum effort/shortest time elapse path will not be taken. Rather the project will be lengthened such that man-hours and funding costs are reasonably uniform. The net effect of this upon choice among modules is to increase the number of independent choices greatly. As an example consider the case where three activities have only one common prerequisite event in the fundamental scheduling net, i.e., these three and basically parallel activities. If we have resources to do only one at a time, the effective schedule net (resource restrained) puts these three activities in series. But note that now the activity order is independent in the series of three and our independent choices increase from one to six. Paradoxically, choice increases as the scheduling net is resource restrained.

A few minor points remain. Debugging is an important part of the total effort. Just as subroutines ease this problem considerably in the domain of programming, their larger modularization units help at the systems level. Documentation which should be complete, standardized and act as an indexing device for revision and implementation, is usually a neglected item. Part of the blame rests with a lack of standardization which results in unnecessary documentation work for the writers and undue analysis time for the revision programmer. Modularization yields a ready-made documentation format that diminishes both problems while at the same time allowing documentation to progress in bits as the module is completed.

Donald V. Mathusz is currently an Operations Research Analyst with a Department of Navy Laboratory. His government experience has included work in reliability systems, threat analysis and logistics systems simulation. His current work involves directing cost-benefit analysis utilizing simulation as both a costing and benefit evaluation tool.

His industrial experience ranges from toolmaking through to the design and management of computerized sales forecasting and production control systems. He has taught statistics and computer systems courses and presently holds the BIE, 1960, from Syracuse University; MMgmtE, 1963, from Rensselaer Polytechnic Institute, and is now attending the University of Maryland.

This should lessen the boredom of the last big write-up while eliminating the management temptation to put off the documentation effort because of new program demands.

2.1 Visible Progress, Political Priority and Education

Whatever the choices are for a project's given implementation rate, we can use them effectively to accomplish two necessary political goals. In a sense, the first is defensive and I call it visible progress. It matters little to higher management what real progress is—if they can't see and evaluate it (barring an unusual total confidence). The uniform display of progress is a political necessity in a department's life. The visibility of progress is served well by the module concept in both presentation and the opportunities for partial implementation.

The order of priority of partial implementation is a potent weapon in visual progress and a second aspect: political priority and education. The political priority refers to the desires of key people as to what they want immediately from the system. By education, we here refer to the problem of teaching management what they can and cannot have from an information system. This is sometimes referred to as growing a system about the management in command and control systems.

The system output modules are very sensitive to this aspect, and little conflict should exist in and between political priority and educational priorities of implementation. Where it does exist, however, it becomes hypercritical and requires the finest diplomacy and management possible. Our purposes here, however, is not to examine this problem, but merely to pinpoint the place of modularization in it.

3.0 Modulization in Systems Designs

While we are not basically concerned with systems design as such, in order to clarify modularization's place in it we are forced to outline some phases of it. This is especially so, since the literature tends to deal with either the highly theoretical or the specific case study. In general, we might say that the relationship of the module to systems design is analogous to the relationship of the subroutine to the program.

3.1 Outline of the Analysis-Design Cycle

Systems analysis typically precedes systems design since it is by definition a predesign process. Actually the systems analysis involves two separate analyses, or at least one would expect to assign different kinds of analysts to them. We would call the first endogenous analysis. It is concerned with the formal or informal information systems already in existence within the relevant organization. The other is exogenous analysis, and it is concerned with all relevant external information sources both specific and general, such as economic indexes.

It should be apparent that the system design interacts with the analysis to produce the final system. This interaction involves goal, economic and technical trade-offs. We prefer to begin the systems design as an independent activity by designing an "ideal" cybernetic system. The project schedule is then tentatively drawn up to verify, find or generate the need design parts.

A critical element in this process is in what manner the vast volumes of detail will be organized. The module serves as the vehicle for organizing this detail into parts that can be evaluated independently.

3.2 The Ideal Cybernetic System Design

The system design may be thought of as a structure comprised of intermodule flows and the functional modules in three classes—physical information, processing and control. The system net may be drawn in a variety of ways, but the net flow using nodes as processing functions and flows as internode directed lines is compact and describes all the needed static relations. A basic cybernetic design for the typical inventory problem of the manufacturing concern is shown *in Figure 1.* It should be kept in mind that this design was made only detailed enough to display and isolate the various kinds of systems design problems. Remember too, that the complex problem of phasing a new system design into an existing prior system will not show here, but on the implementation schedule net. So we have not presented the entire story, but only enough to show the basic place of the software module.

(to be continued next month)

HEXIDECIMALS

(Continued from page 11)

the first digit is 0. Repeat this step multiplying the bar of the other digit by the digit value. The bar multiplied by the bar produces a carry of 4.

$$\begin{array}{r} \bar{7} \\ \bar{6} \\ \hline 2\bar{2} \\ 30 \\ 3\bar{0} \\ 40 \\ \hline \bar{5}2 \end{array}$$

4. If one of the two digits to be multiplied has a bar, multiply the two digits modulo 8. Determine the carry number by the prescribed method. If the number of mod 8's is odd, the resultant has a bar. Now multiply the bar value of the bar-digit by the number value of the other digit, taking half the resulting value as the carry. If the number is odd, the carry value is half of one-less-than the number and the first digit is 0.

$$\begin{array}{r} \bar{7} \\ 5 \\ \hline 23 \\ 2\bar{0} \\ \hline 4\bar{3} \end{array}$$

The bi-octal subtraction method involves the use of subtracting two digits as in decimal whenever the minuend is larger than the subtrahend. If the subtrahend is larger than the minuend, a borrow of one from the preceding digit of the minuend is taken and the two digits are subtracted as in decimal, with the resultant digit producing a value minus two. The resultant digit will have a bar symbol under the following conditions.

1. If the two digits that are to be subtracted do not have the bar symbol, the resultant digit will have a bar value.

$$\begin{array}{l} 34 \\ -5 \\ \hline 2\bar{9} - 2 = 2\bar{7} \end{array}$$

2. If the subtrahend digit has a bar and the minuend does not, the resultant digit will not have a bar value.

$$\begin{array}{l} 23 \\ -\bar{6} \\ \hline 17 - 2 = 15 \end{array}$$

3. If the two digits that are to be subtracted both have a bar symbol, the resultant digit will have a bar value.

$$
\begin{array}{r}
5\bar{2} \\
-7 \\
\hline
4\bar{5} - 2 = 4\bar{3}
\end{array}
$$

The addition and multiplication table on page 27 shows the consistency and simplicity of the above rules governing bi-octal representation and arithmetic operation.

The division scheme of bi-octal numbers involves taking the divisor into the dividend. As in decimal division, bi-octal division will employ a rigorous trial and error procedure to obtain the proper quotient.

$$
\begin{array}{r}
13\ 3\surd\bar{3} \\
\bar{3}\surd\overline{\bar{5}4} \\
\bar{3} \\
\hline
24 \\
21 \\
\hline
3
\end{array}
$$

The bi-octal number may be converted to decimal integers by repetitive division of the value 2 into the quotient. The remainders make up the decimal value.

$$
\begin{array}{ll}
\bar{2}\ /\ \bar{2}3\bar{4} & \\
\bar{2}\ /\ 106 & 0 \text{ Ans. } 2620_{10} \\
\bar{2}\ /\ 1\bar{2} & 2 \\
\quad 2 & 6
\end{array}
$$

(Continued on page 27)


Programmers and systems analysts...
Univac: Where you can help make the world a little bit better
An American astronaut hasn't walked on the surface of the moon . . . yet.
But now America's manned Apollo 8 space shot, with the help of UNIVAC® computers, has successfully orbited the moon. Which brings an American lunar landing that much closer to a reality.
The people who are proudest of this Apollo flight aren't just NASA people. They are also the Univac programmers and systems analysts who spent months writing the programs that made the use of telemetering of this flight possible.
You see, we understand that nothing bothers a programmer or systems analyst more than doing busy work. So we keep our people busy with meaningful work like the Apollo program.
What's more, we feel that the important problem is a problem that still hasn't been solved: the problem of keeping America strong. Because when America is strong she can better fulfill her obligations to underprivileged countries looking to her for help and direction. This is where you come in.
Univac needs people who are old-fashioned enough to care about making the world a little better. Who are interested in making a real contribution, not just marking time. And who want to be rewarded on the basis of the excellence of their service, instead of just on the length of their service.
If you're this kind of person, we here at Univac can't offer you a job. But we can offer you a fine career. Think it over and then check the facing page for a complete list of current Univac job opportunities.
SPERRY RAND
UNIVAC
An Equal Opportunity Employer M/F

BI-OCTAL ADDITION AND SUBTRACTION TABLE

	1	2	3	4	5	6	7	$\bar{0}$	$\bar{1}$	$\bar{2}$	$\bar{3}$	$\bar{4}$	$\bar{5}$	$\bar{6}$	$\bar{7}$
1	2	3	4	5	6	7	$\bar{0}$	$\bar{1}$	$\bar{2}$	$\bar{3}$	$\bar{4}$	$\bar{5}$	$\bar{6}$	$\bar{7}$	10
2	3	4	5	6	7	$\bar{0}$	$\bar{1}$	$\bar{2}$	$\bar{3}$	$\bar{4}$	$\bar{5}$	$\bar{6}$	$\bar{7}$	10	11
3	4	5	6	7	$\bar{0}$	$\bar{1}$	$\bar{2}$	$\bar{3}$	$\bar{4}$	$\bar{5}$	$\bar{6}$	$\bar{7}$	10	11	12
4	5	6	7	$\bar{0}$	$\bar{1}$	$\bar{2}$	$\bar{3}$	$\bar{4}$	$\bar{5}$	$\bar{6}$	$\bar{7}$	10	11	12	13
5	6	7	$\bar{0}$	$\bar{1}$	$\bar{2}$	$\bar{3}$	$\bar{4}$	$\bar{5}$	$\bar{6}$	$\bar{7}$	10	11	12	13	14
6	7	$\bar{0}$	$\bar{1}$	$\bar{2}$	$\bar{3}$	$\bar{4}$	$\bar{5}$	$\bar{6}$	$\bar{7}$	10	11	12	13	14	15
7	$\bar{0}$	$\bar{1}$	$\bar{2}$	$\bar{3}$	$\bar{4}$	$\bar{5}$	$\bar{6}$	$\bar{7}$	10	11	12	13	14	15	16
$\bar{0}$	$\bar{1}$	$\bar{2}$	$\bar{3}$	$\bar{4}$	$\bar{5}$	$\bar{6}$	$\bar{7}$	10	11	12	13	14	15	16	17
$\bar{1}$	$\bar{2}$	$\bar{3}$	$\bar{4}$	$\bar{5}$	$\bar{6}$	$\bar{7}$	10	11	12	13	14	15	16	17	$1\bar{0}$
$\bar{2}$	$\bar{3}$	$\bar{4}$	$\bar{5}$	$\bar{6}$	$\bar{7}$	10	11	12	13	14	15	16	17	$1\bar{0}$	$1\bar{1}$
$\bar{3}$	$\bar{4}$	$\bar{5}$	$\bar{6}$	$\bar{7}$	10	11	12	13	14	15	16	17	$1\bar{0}$	$1\bar{1}$	$1\bar{2}$
$\bar{4}$	$\bar{5}$	$\bar{6}$	$\bar{7}$	10	11	12	13	14	15	16	17	$1\bar{0}$	$1\bar{1}$	$1\bar{2}$	$1\bar{3}$
$\bar{5}$	$\bar{6}$	$\bar{7}$	10	11	12	13	14	15	16	17	$1\bar{0}$	$1\bar{1}$	$1\bar{2}$	$1\bar{3}$	$1\bar{4}$
$\bar{6}$	$\bar{7}$	10	11	12	13	14	15	16	17	$1\bar{0}$	$1\bar{1}$	$1\bar{2}$	$1\bar{3}$	$1\bar{4}$	$1\bar{5}$
$\bar{7}$	10	11	12	13	14	15	16	17	$1\bar{0}$	$1\bar{1}$	$1\bar{2}$	$1\bar{3}$	$1\bar{4}$	$1\bar{5}$	$1\bar{6}$

EXAMPLE: $6 + 2 = \bar{0}$ $\qquad \bar{6} + \bar{2} = 1\bar{0}$

$\bar{7} - 3 = \bar{4}$ $\qquad \bar{7} = \bar{3} = 4$

BI-OCTAL MULTIPLICATION AND DIVISION TABLE

1	2	3	4	5	6	7	$\bar{0}$	$\bar{1}$	$\bar{2}$	$\bar{3}$	$\bar{4}$	$\bar{5}$	$\bar{6}$	$\bar{7}$
2	4	6	$\bar{0}$	$\bar{2}$	$\bar{4}$	$\bar{6}$	10	12	14	16	$1\bar{0}$	$1\bar{2}$	$1\bar{4}$	$1\bar{6}$
3	6	$\bar{1}$	$\bar{4}$	$\bar{7}$	12	15	$1\bar{0}$	$1\bar{3}$	$1\bar{6}$	21	24	27	$2\bar{2}$	$2\bar{5}$
4	$\bar{0}$	$\bar{4}$	10	14	$1\bar{0}$	$1\bar{4}$	20	24	$2\bar{0}$	$2\bar{4}$	30	34	$3\bar{0}$	$3\bar{4}$
5	$\bar{2}$	$\bar{7}$	14	$1\bar{1}$	$1\bar{6}$	23	$2\bar{0}$	$2\bar{5}$	32	37	$3\bar{4}$	41	46	$4\bar{3}$
6	$\bar{4}$	12	$1\bar{0}$	$1\bar{6}$	24	$2\bar{2}$	30	36	$3\bar{4}$	42	$4\bar{0}$	$4\bar{6}$	54	$5\bar{2}$
7	$\bar{6}$	15	$1\bar{4}$	23	$2\bar{2}$	31	$3\bar{0}$	$3\bar{7}$	46	$4\bar{5}$	54	$5\bar{3}$	62	$6\bar{1}$
$\bar{0}$	10	$1\bar{0}$	20	$2\bar{0}$	30	$3\bar{0}$	40	$4\bar{0}$	50	$5\bar{0}$	60	$6\bar{0}$	70	$7\bar{0}$
$\bar{1}$	12	$1\bar{3}$	24	$2\bar{5}$	36	$3\bar{7}$	$4\bar{0}$	51	$5\bar{2}$	63	$6\bar{4}$	75	$7\bar{6}$	$\bar{0}7$
$\bar{2}$	14	$1\bar{6}$	$2\bar{0}$	32	$3\bar{4}$	46	50	$5\bar{2}$	64	$6\bar{6}$	$7\bar{0}$	$\bar{0}2$	$\bar{0}\bar{4}$	$\bar{1}\bar{6}$
$\bar{3}$	16	21	$2\bar{4}$	37	42	$4\bar{5}$	$5\bar{0}$	63	$6\bar{6}$	$7\bar{1}$	$\bar{0}4$	$\bar{0}\bar{7}$	$\bar{1}\bar{2}$	$\bar{2}5$
$\bar{4}$	$1\bar{0}$	24	30	$3\bar{4}$	$4\bar{0}$	54	60	$6\bar{4}$	$7\bar{0}$	$\bar{0}4$	$\bar{1}0$	$\bar{1}\bar{4}$	$\bar{2}\bar{0}$	$\bar{3}4$
$\bar{5}$	$1\bar{2}$	27	34	41	$4\bar{6}$	$5\bar{3}$	$6\bar{0}$	75	$\bar{0}2$	$\bar{0}\bar{7}$	$\bar{1}\bar{4}$	$\bar{2}\bar{1}$	$\bar{3}6$	$\bar{4}3$
$\bar{6}$	$1\bar{4}$	$2\bar{2}$	$3\bar{0}$	46	54	62	70	$7\bar{6}$	$\bar{0}\bar{4}$	$\bar{1}\bar{2}$	$\bar{2}\bar{0}$	$\bar{3}6$	$\bar{4}4$	$\bar{5}2$
$\bar{7}$	$1\bar{6}$	$2\bar{5}$	$3\bar{4}$	$4\bar{3}$	$5\bar{2}$	$6\bar{1}$	$7\bar{0}$	$\bar{0}7$	$\bar{1}6$	$\bar{2}5$	$\bar{3}4$	$\bar{4}3$	$\bar{5}2$	$\bar{6}1$

TROUBLE-TRAN PRESENTS XTRAN'S ADVENTURES IN FORTRAN

By GEORGE N. VASSILAKIS

Send your ANSWER to the problems posed here in each issue to:

TROUBLE-TRAN EDITOR

software age
P. O. Box 2076
2211 Fordem Ave., Madison, Wis. 53701

You can also profit by submitting PROBLEMS for this feature. If your problem is FORTRAN programming is selected for use in this feature, you will receive ---------- **$25.00**

TROUBLE-TRAN's Objectives:

1. To have fun.
2. To promote USA Standard FORTRAN by pointing out differences and inconsistencies of existing FORTRAN Compilers.
3. To alert programmers to the physical limitations of hardware.

Contest Rules:

1. The best answer (best explanation) bearing the earliest postmark wins -- **$25.00**
2. The second best answer with the earliest postmark will net the reader submitting it ---------------------- **$15.00**

PROBLEM OF THE MONTH

Here is a program that will compile without any difficulty; but will it execute?

```
C SUBMITTED BY LEON ZAR
      K=0
      GØ TØ 400
  200 GØ TØ (101,102,101,102,101),K
  101 WRITE (6,10)
      STØP
  101 WRITE (6,9)
      STØP
  400 IF(K)402,401,401
  401 K=-5
      GØ TØ 200
  402 WRITE (6,8)
      STØP
   10 FØRMAT (14H GØ TØ IS ØKAY)
    9 FØRMAT (18H GØ TØ IS NØT ØKAY)
    8 FØRMAT (17H GØ TØ HAS FAILED)
      END
```

Which FORMAT message, if any, will be printed?

ANSWER TO LAST MONTH'S PROBLEM

The problem was stated as follows:

```
   X=1.0            SUBRØUTINE SUB1(X)      SUBRØUTINE SUB2
   CALL SUB1(X)     CØMMØN X                CØMMØN X
   STØP             CALL SUB2               WRITE(6,1)X
   END              RETURN                1 FØRMAT(3H1X=E15.8)
                    END                     RETURN
                                            END
```

Will it compile? Will it execute? What will it print for X?

The solution, of course, depends on what the compiler would do with X in subroutine SUB1. The appearance of X as a dummy argument and also as a variable in CØMMØN storage is not allowed. However, not all compilers are sophisticated enough to detect this illegal use.

If the subroutine compiles, changes are the compiler will treat the two uses of X as two different variables, and when SUB1 is called all subsequent appearances of X will be using the value of the dummy argument X. The only exception is the X in CØMMØN which will be assigned a different cell and nothing will be placed in this cell at execution time. The value of X that will be printed by subroutine SUB2 depends on whether or not the loader sets core to zero at load time.

The level H FORTRAN Compiler on the IBM OS/360 did not allow SUB1 to compile. The same program compiled and executed on the IBM 7094, and printed a –0.00 as the value of X. The CDC-6600 FORTRAN will also allow this to compile and execute with different cells assigned to the two different appearances of X in SUB1. If you are a CDC user, this problem may not exist on your machine by the time you read this column, since this item is already included in the CDC FORTRAN laundry list.

P. S. Several readers have written to tell me that I was wrong in stating that the problem in the February issue did not violate any FORTRAN standards. My attention was called to section 8.4.2 of the USA Standard FORTRAN. One paragraph in this section reads:

> "If a subroutine reference causes a dummy argument in the referenced subroutine to become associated with another dummy argument in the same subroutine or with an entity in common, a definition of either entity within the subroutine is prohibited."

Even though I must admit I was wrong and offer my apologies, it gives me a great pleasure to see that some readers are treating my column as a **core dump** which needs debugging.

My mail indicates that the percentage of programmers who pay attention to the USA Standard FORTRAN is very small (under 10%). Is there anything we can do to help? Please send your suggestions to this editor.

Thank you
XTRAN

TROUBLE-TRAN WINNERS

$25.00 first prize for the February problem: Mr. D. J. Frailey, 329 Pearl Street, West Lafayette, Indiana 47906.

$15.00 second prize for the February problem: Mr. J. P. Mashey, 119 Locust Lane C-7, State College, Pennsylvania 16801.

Programmers and Systems Analysts:

Come to IBM and help solve tomorrow's programming problems.

Many people talk about tomorrow. At IBM's Federal Systems Division, we work on it.

We need creative programmers that think ahead of the state-of-the-art. Our present projects are geared to help solve problems in the 1970's.

Wide range of projects.

Right now we're researching missile systems to cover the needs of the next decade. We're out to help monitor air traffic while en route. Retrieving delicate seismological data. And becoming even more involved in America's space program.

In this work, of course, we use the latest computers and peripheral equipment. Much of the work is in real-time... involving multiple-access concepts.

What you'd do.

We need programmers and analysts with experience in one of two basic areas: information-handling systems or scientific engineering programming. You should have a Bachelor's degree in Mathematics, Physics, Engineering, Economics, or Statistics with at least one year's experience.

Can you solve problems?

Today's major growth industry is information handling and control. And IBM is a leader in that field. This growth environment can bring out the best of your talents and abilities. Because in a growth company like IBM you must work constantly toward greater achievement. This means more opportunities to achieve distinction and personal recognition.

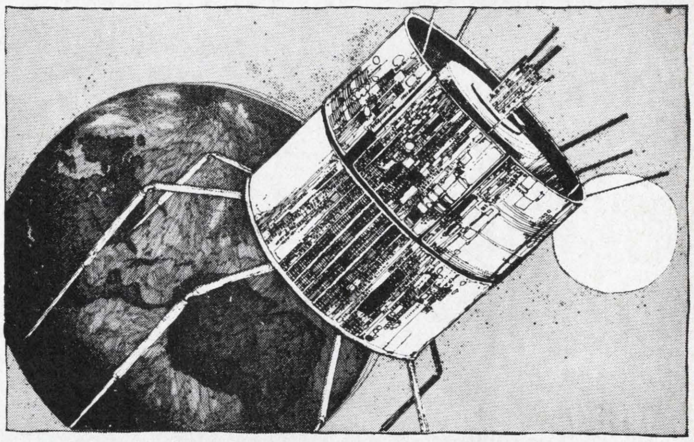
So if you're a problem-solver who wants a personal sense of achievement and recognition for your hard work in an exciting growth company, consider IBM.

Call or write.

Learn more about the opportunities for you at IBM. Immediate openings exist in the Northern New Jersey area; and in metropolitan Washington, D.C. Call Jim Dunn at (301) 921-7724 collect. Or send a brief letter or resume to him at IBM Corporation, Federal Systems Division Headquarters, Dept. CD1012, 18100 Frederick Pike, Gaithersburg, Maryland 20760.

An Equal Opportunity Employer

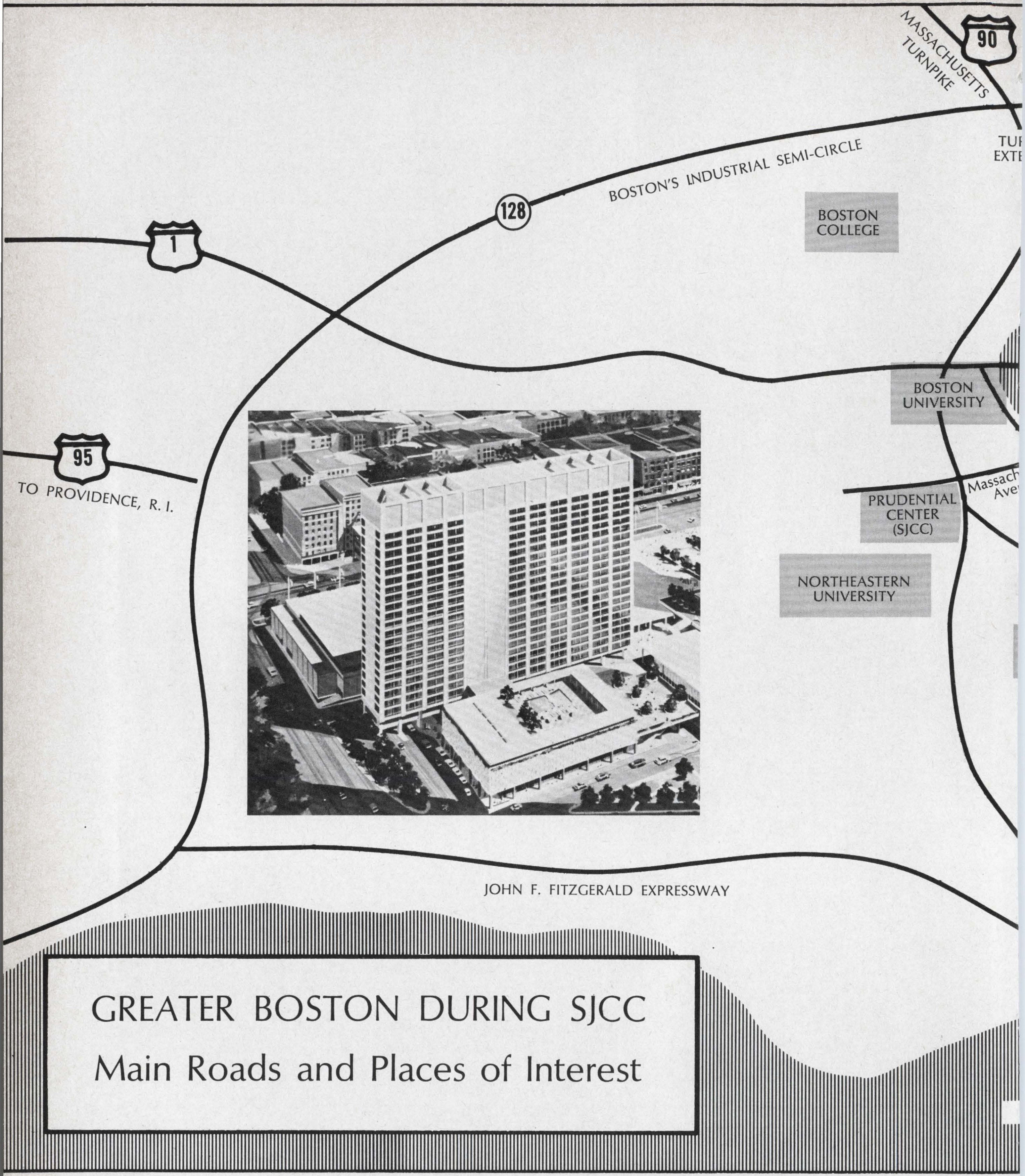
IBM®



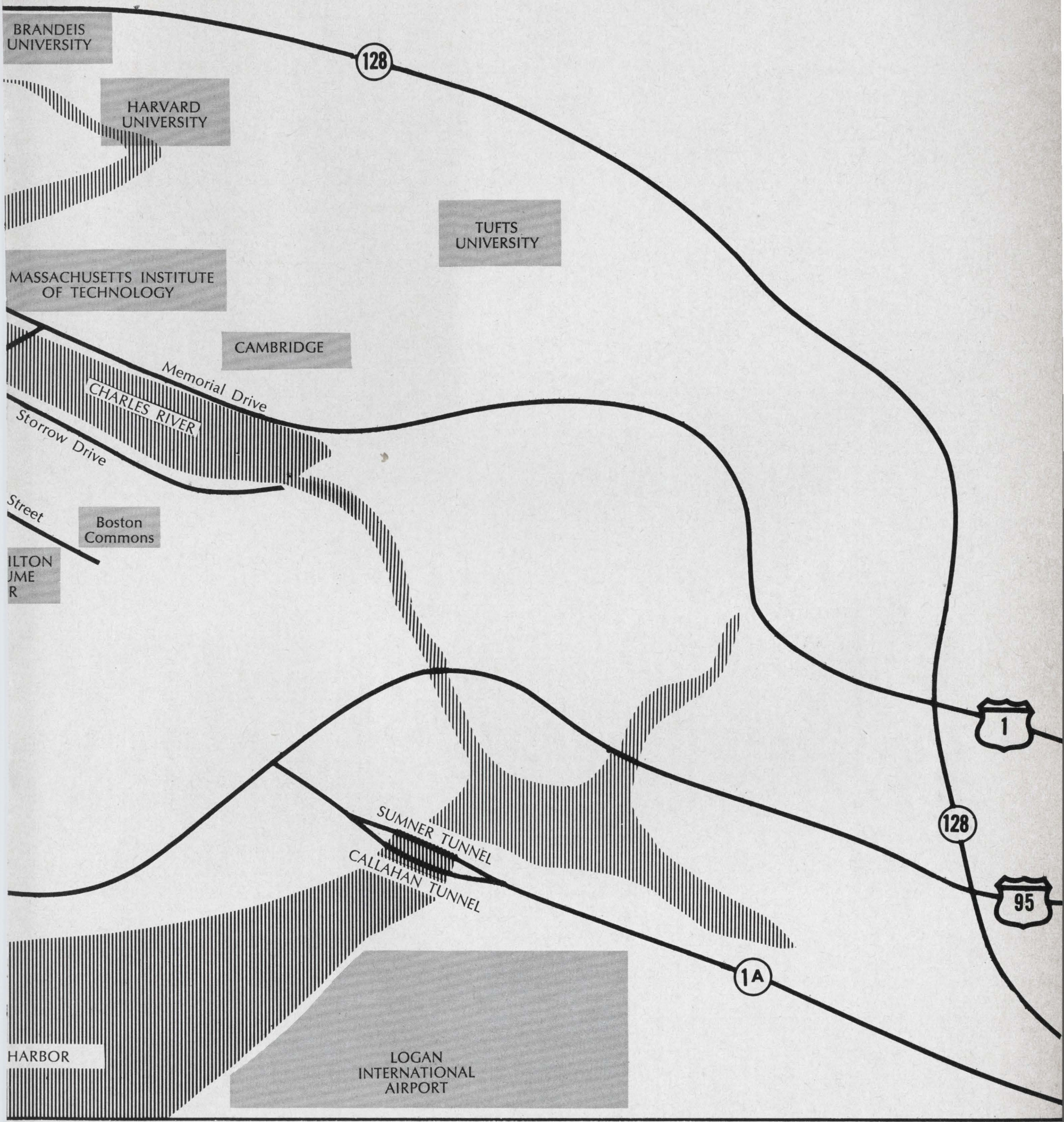
IBM works with NASA's Goddard Space Flight Center to help control and maneuver satellites like ATS (Applications Technology Satellite).

IBM is helping air traffic controllers by computerizing their information input and display systems.

Xerox Square at Night



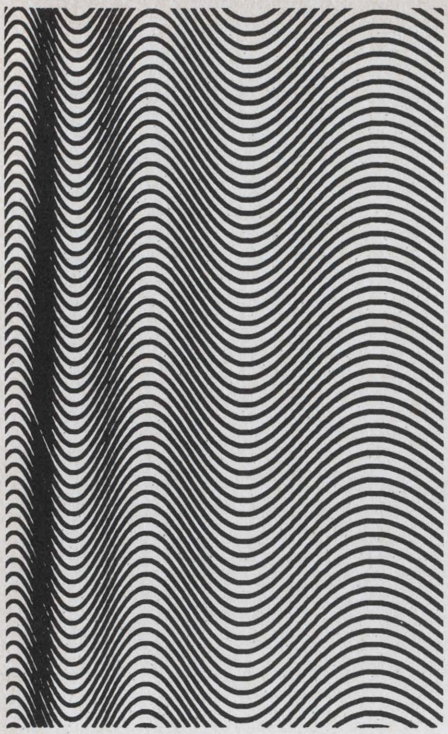
VISITING BOSTON DURING



SJCC?

The greater Boston area offers unlimited opportunities to combine convention-going with business and pleasure. The Spring Joint Computer Conference, headquartered at the Sheraton–Boston Hotel in downtown Boston, is in an electronic wonderland for the computer professional.

Old Boston is historically the center of scientific thought and development. Here is the first subway

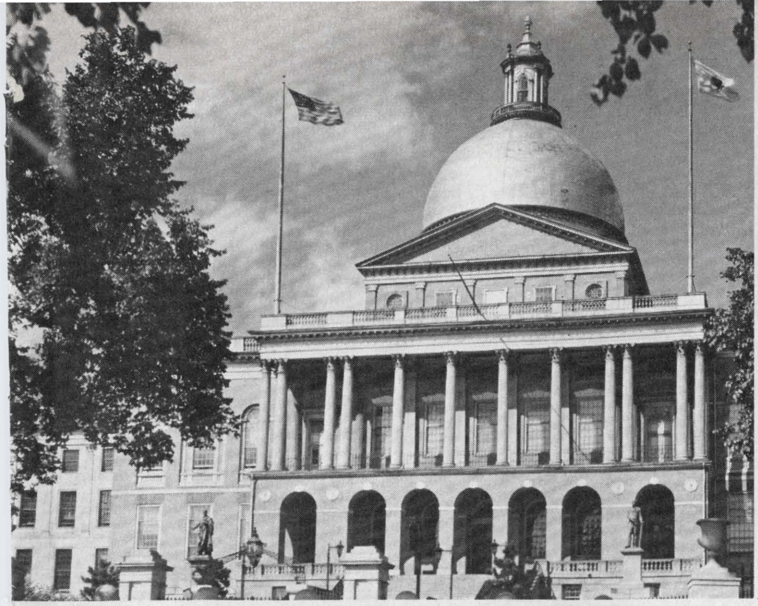


in America; the birthplace of rocketry; here super-speed trains were designed and tested long before a need was evident.

New Boston has one of the greatest concentrations of research and scientific development and training in the United States, with over 40 percent of its land area occupied by institutions mainly devoted to teaching and research.

There are innumerable opportunities for seeing and doing within easy distance of the SJCC and its neighboring hotels. The Prudential complex, containing the Prudential

History Blends With



Boston's historic State House dominates Beacon Hill.

The traveller's view—of Boston Harbor.

Tower, War Memorial Auditorium, and the Sheraton–Boston, is the site of the meetings themselves.

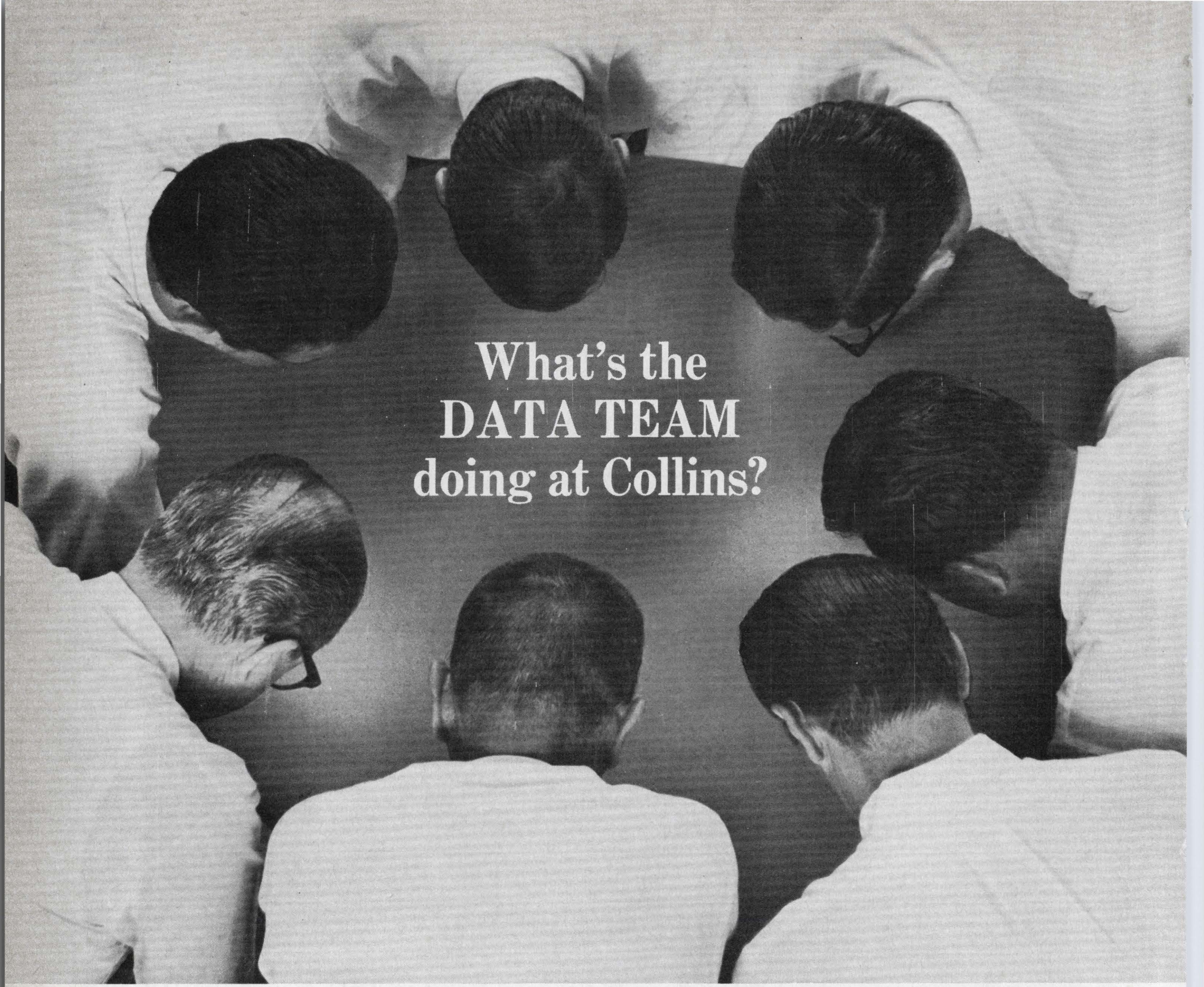
To the east, along Boylston Street, are the historic Public Gardens, noted for their swan boats, and the Boston Common. Just south is Park Square and the Statler–Hilton, location of the SOFTWARE AGE RESUME CENTER.

To the north of the Common is Beacon Hill and Louisburg Square, a picturesque and historic area of Old Boston. Government Center—a complex of buildings, plazas and pedestrian malls which has revitalized the downtown core of Boston, is located just east of Beacon Hill.

Technology in Boston

Of more specific interest to the computer professional are the numerous colleges and universities located near Boston, the N.A.S.A. complex near M.I.T., and Highway 128, Boston's "industrial road." Technical tours will be conducted by the convention organizers to these areas.

The students' view of Boston across the Charles River in Cambridge.



Working with Collins' new C-System, the first completely integrated system with virtually unlimited expansion capability.

The C-System is creating exceptional career opportunities for: Programmers, Hardware Diagnostic Programmers, Circuit Design Engineers, Memory Design Engineers, Digital Systems Engineers, Digital Systems Analysts, Logic Design Engineers, Mechanical Engineers, Data Systems Analysts and Applied-Systems Analysts/ Programmers.

In addition, other individuals with up to 3 years experience and a degree in electrical or mechanical engineering, physics, mathematics, or other physical sciences will find an outstanding opportunity to learn and progress rapidly in the data field, even without previous data experience.

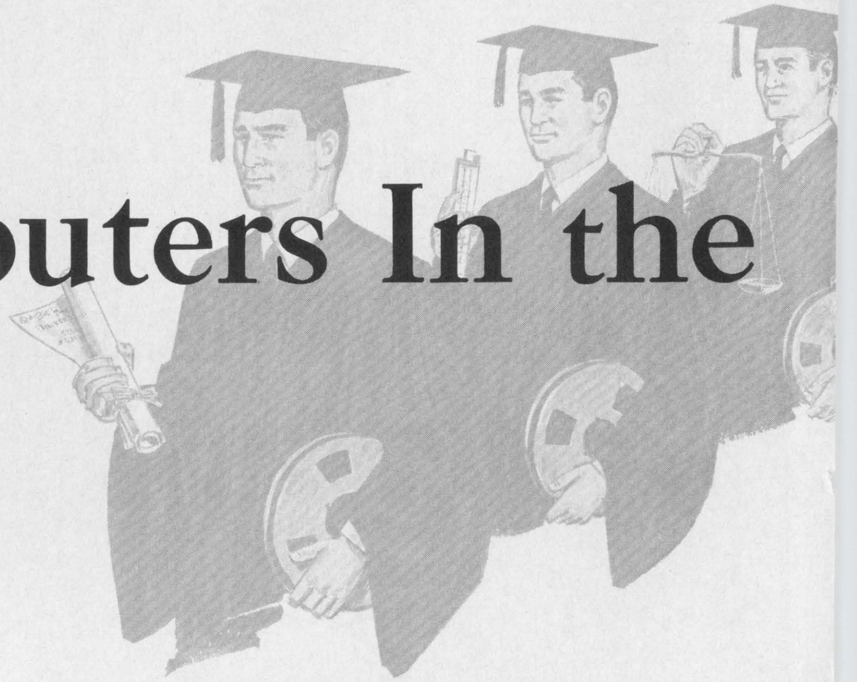
Collins data systems are used by airlines and railroads to process and transmit hundreds of thousands of messages a day. They provide computer control of earth stations for satellite communications, permit digital control of communication/navigation systems in airliners and military aircraft, and combine communication, computation and control functions in single systems for large business and military operations.

If you are interested in joining the Collins data team – and benefiting from the creative environment and career opportunities offered by Collins – send your resume in confidence to Manager of Professional Employment, Dept. 102, Collins Radio Company, Dallas, Texas 75207; Cedar Rapids, Iowa 52406, or Newport Beach, California 92663.

COLLINS

an equal opportunity employer

Computers In the



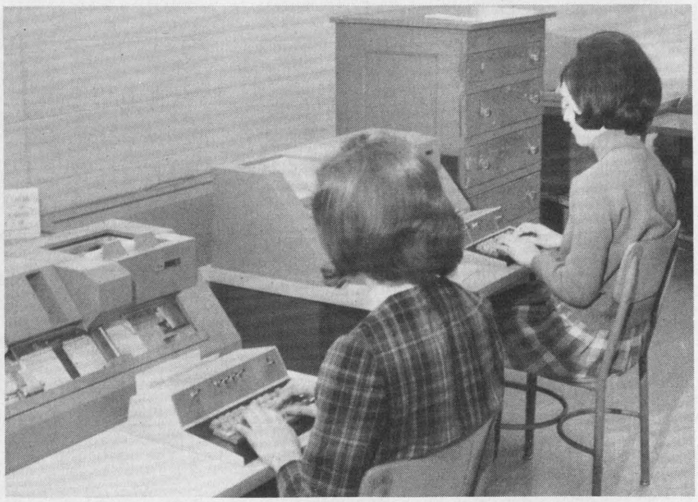
by Charles H. McCoach

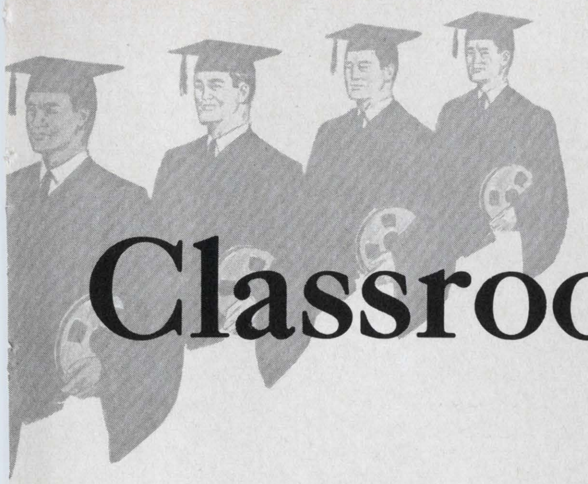
This paper will discuss three distinct subjects that will reflect the use of computers in secondary education. The first will cover what is currently being done nationwide, the second will be a reflection of what we feel should be done and the third will discuss what we are doing in Barry County, Missouri.

When we look at computer education nationwide it becomes immediately obvious that more could be said about what is not being done than what is being done. A survey conducted by the University of Denver during the winter of 1967 reveals the number of schools involved in computer work. The results of this survey is given in Table 1. Even a cursory study of this table gives us a stark picture of the neglect of this increasingly important subject. Briefly the use of computers in high schools will fall into one of the following categories:

1. Computer Aided Instruction (CAI)
2. Administration of the School
3. A tool to be used in conjunction with another course
4. As a unique subject in itself. Uses, languages, etc.

Of the four applications we feel that only the last two are legitimate courses that could be called computer science. The third application





Classroom

McCoach has had extensive experience in secondary education, both as an instructor and administrator. He was formerly with the Boeing Company in their Systems Administration and Computing Department.

usually involves teaching an advanced mathematics science, or business course while the subject of the computer remains incidental and collateral to the main topic. This approach has merits, but it also has a number of liabilities not the least of which is the severe limitation on the number of students that become involved, and the depth of the study of the computer.

It is not the purpose of this paper to discuss the reasons for the neglect of this subject, but the reader might be interested to know the most frequently expressed. They are:

1. Hardware availability
2. Costs
3. Instructional personnel
4. Can secondary students learn the material?

While each of these, on the surface, appears to have a basis in fact, our investigations have revealed that they are not difficult to overcome.

What *should* be done about computer education in secondary schools? Having had experience in both education and industrial scientific

A major consideration in establishing a Computer Science course is the ability for students to prepare and run their own programs. At Monett High School, students utilize the facilities of General Electric Time-Sharing Services, Dallas (left). The smaller high school in the program uses a key punch (right) to prepare programs for running on the computer at the University of Arkansas at Fayetteville.

USE OF THE COMPUTER IN SECONDARY EDUCATION 1967

	Teaching Device	Instr. Device	Data Source For Teachers	Programming Scientific Applic.
Alabama	0	0	0	0
Alaska	0	0	0	0
Arizona	--	6	----	1
Arkansas	0	0	0	0
California	1	60	0	100
Colorado	22	5	11	3
Connecticut	0	16	1	28
Delaware	0	0	57	11
Florida	0	5	1	2
Georgia	--	--	----	---
Hawaii	0	0	0	0
Idaho	1	0	1	2
Illinois	0	4	0	27
Indiana	15	11	0	16
Iowa	0	12	----	4
Kansas	0	0	7	1
Kentucky	1	1	----	---
Louisiana	0	0	3	3
Maine	0	2	0	0
Maryland	0	4	1	0
Mass.	1	10	0	6
Minn.	0	20	0	3
Miss.	0	0	3	0
Missouri	0	2	0	6
Montana	0	0	0	---
Nevada	0	3	1	3
New Hamp.	0	9	0	0
N. Mex.	0	6	3	7
New York	0	50	----	50
N. Car.	0	1	1	1
N. Dak.	5	5	0	3
Ohio	0	6	0	0
Oklahoma	6	5	40	11
Oregon	1	34	1	10
Penn.	5	65	0	53
R. I.	0	6	2	6
S. Car.	0	0	----	0
S. Dak.	0	0	0	0
Tenn.	15	2	2	0
Texas	7	45	150	5
Utah	1	0	2	0
Vermont	0	4	0	4
Virginia	0	1	----	0
Washington	0	10	----	35
W. Virginia	0	0	0	4
Wisconsin	0	28	66	2
Wyoming	0	0	0	0
Wash. D. C.	14	14	7250	0
Puerto Rico	--	--	21	1
Canal Zone	0	0	2	0

Univ. of Denver Study, 1967.

applicatioins I have very strong convictions about the answer to this question. It should be the objective of this discussion to present different approaches that may be taken in adopting this subject to a secondary curriculum. Where applicable the author has added opinions that are the result of a comprehensive survey of other schools involved in similar programs.

Computer Science is a multi-faceted course. It should not be considered as belonging exclusively to any particular department in the curriculum. Rather, every application in all fields must be encouraged. It is very easy to identify three spcific departments in the secondary school that are now, or will be in the near future, directly involved in some phase of computer usage. The three obvious dpartments:

1. The sciences—Including Biology
2. Business
3. Mathematics

There are others that are not so obvious but nonetheless involved. However, we shall restrict our discussion to these three.

The immediate and obvious question should be, "To what extent should these departments be involved, and can a single course apply equally well to all?" It is our personal opinion that not only *can* the course span these three subjects but it *must*. Let me hasten to add that the ideal method of adopting the subject in the curriculum would be to offer a separate course in each area. We realize this may be unrealistic

Within the junior and senior level course, students break into group sessions emphasizing the use of computers in business, math or the sciences. Programs are developed for specific subject applications.

in the initial stages but perhaps it should be a goal for the future.

There are several directions that can be taken in the initial year. Two of the more obvious will be discussed.

1. *Comprehensive*—This approach should attempt to involve as many disciplines as possible. Students should be selected from

A graphic illustration of the use of computers in business is made for the students by visiting the data processing departments of local businesses, such as this installation at the Vaisey Bristol Shoe Company.

the three major departments listed above. The course would of necessity be tailored to fit a variety of interests and backgrounds. This in turn would require a great deal more work for the instructor since three distinct types of applications must be allowed. Personal guidance and counseling in program development would be required. This type of project may be too ambitious for the first year of operation.

2. *Intensive*—This approach involves students that are primarily selected from a given discipline. Strong emphasis should be placed on applications of computer usage in the subject matter under consideration. This appears to be the most desirable from the standpoint of preparation and administration. This approach should not be the end result, but rather a means to reach a compreesnsive course of study. The three types of programs might be:
   A. For scientific applications—The students would use the computer to solve scientific problems. Emphasis would be placed on the computer as a tool of the research scientist.
   B. For business applications—Emphasis would be placed on data processing, information storage and retrieval and cost analysis.
   C. Mathematics—The student would learn complex and sophisticated mathematics, especially numerical analysis for computer applications.

Each of these have merit, but the ideal program would be a merging of A, B, and C, with each student learning the same material in computer science and each applying these skills to his own area of interest.

Course content is directly related to the type project selected from the previous article. In the event the first, or comprehensive, approach is followed, Computer Science should be taught as an end in itself with applications being individually selected. In any event the *computer* subject matter studied is to be identical, only the *sequence* of the course would be different. The comprehensive approach lends itself to lectures on Monday, Wednesday and Friday with Tuesday and Thursday relegated to individual research and projects.

The second or intensive program is the easier of the two to administer. The sequence of the course is altered to fit more exactly the subject matter considered. The redeeming merit of this approach is the small number of teachers required to make the project successful.

The material concerning computers follows the following rough outline:

1. Introduction to Computers and Their Uses
   A. Definitions of Terms
   B. Discussion of Equipment
2. Computer Languages
   A. Fortran
   B. Cobol
3. Computer Software
   A. Compilers
   B. System Action
4. Machine Logic

The material should be merged with actual computer programming. We have reason to believe that within one month the sudent will have become familiar with the language to the extent that he will be writing programs to be run on an available computer.

This discussion is not intended to be exhaustive. Rather it presents in general the type of program possible for our school systems. There are other approaches and directions one may take but we feel that the methods discussed will place computer science on a strong foundation. At the end of this article is a questionnaire that will allow you to express your own opinion about the direction *you* feel secondary Computer Education should follow. This questionnaire is being sent to other interested personnel.

In Barry County we are taking what we consider to be the total approach. We are attempting to involve as many people in the program as we can possibly reach. This has been very successful thus far. We are reaching secondary students, secondary teachers, administrators and community people. If the communications barrier is to be broken, a concerted effort must be made to destroy the myths that have been established about the subject of computers. Following is a description of the program that we have established in Barry County:

1. Two county high schools are now teaching a full academic year course to juniors and seniors. The larger of the two schools is utilizing General Electric Time-Sharing Services out of Dallas, Texas. The other is running student programs on the computer located at the University of Arkansas at Fayetteville, Arkansas. Neither of these programs reflect the optimum hardware set up. They are doing quite well.
2. An extension course has been established by the University of Missouri at Rolla for the teachers of the county and any other interested individuals. The course carries three semester hours credit. Currently we have enrolled 14 teachers and administrators, and four community participants are auditing the course. All programs are being run on the University computer. It is interesting to note that the teachers are drawn from several disciplines including Mathematics, Science, Business, Electronics, and Auto Mechanics.
3. The Area Vocational School located at one of the schools of the county is offering an evening course for post-high school students who are interested in computer work.
4. In cooperation with the University of Missouri at Rolla, one of our high schools has received National Science Foundation funds to conduct a Summer Institute in Computer Science to be held at the local school during the summer of 1969. The Institute will train teachers of secondary schools sufficiently to conduct a Computer Science program in their own school during the 1969–70 school year.

It is quite obvious that the program is developed to spread this subject as wide as possible and to involve as many people as possible. The people of the county may find that when this year is complete that they have a tiger by the tail and cannot let go even if they wish to. This is our intention. ■

Questionnaire

Computer Science Curriculum Committee

Box 446, Purdy, Missouri 65734

(Questionnaires returned to S/A will not be forwarded)

1. Your position and title ....

   ....
2. Major application. Check one
   - A. Scientific Application ......
   - B. Business Applications ......
3. Computer Language used ....
4. Highest degree or diploma held and major field ....
5. Do you feel that Computer usage should be taught in the secondary schools? Yes ......
   No ......
6. If your answer to the preceding question was yes, which language(s) do you think should be taught? ....
7. In your opinion do you feel that high school seniors can comprehend an assembler language? ....
8. Which of the following do you feel most closely fits your philosophy of Computer Science. (Check one)
   1. Computer Science as a subset of an advanced mathematics course ....
   2. Computer Science as a subset of an advanced science class ....
   3. Computer Science as a subset of an advanced business class ....
   4. Computer Science as a unique subject ....
   5. None of these ....
9. If your response to the above was number 5, please describe briefly your own opinion about course content. ....

   ....

   ....
10. May we correspond with you in the future about this matter? Yes .... No ....
    Address ....

    ....

    ....

    ....

new products

Fimaco, Inc. of Philadelphia has developed **a new simplified computer language, BOCOL** (Basic Operating Consumer-Oriented Language).

BOCOL uses clear, precise statements thereby reducing learning time. Even the novice programmer who has written one or two programs no longer needs to refer to the manual of operation.

Input of information of environment, equipment and formatting do not require separate definition. Also, BOCOL requires only one pass for compiling and processing. In these two ways machine and programming time are decreased and efficiency, productivity and economy are increased.

BOCOL statements consist of certain "key words" and as much "free wording" as the user desires. Multiple statements may be written to form sentences of any length which may be organized to form BOCOL paragraphs. One or several paragraphs may be ordered into subroutines.

For more information, circle No. 10 on the Reader Service Card

* * *

The Potter instrument company has announced the development of a new concept in **unit record equipment**.

The key to the new system is a magnetic bar code with an associated alpha-numeric character printed on cards of key punch size. The coded characters are both man and machine readable by a Potter printer, model HSP-3502A operating as an off line print station. The unit is a standard Potter high-speed printer specially adapted to print the magnetic characters required in the new system. Manual card preparation can be accomplished by using a modified standard typewriter.

A magnetic mark card reader is used to read the six-line cards produced by both the printer and the typewriter. The unit can read at 300 cards per minute.

The hole-free, readable cards are designed to be an effective and convenient turnaround document which could be especially useful for utility billing and similar applications.

For more information, circle No. 11 on the Reader Service Card

* * *

Arthur S. Kranzley and Company of Cherry Hill, New Jersey has announced the development of the **CHARGE-SERV** systems for **computerized processing of accounts receivable.** CHARGE-SERV Systems are said to be applicable to auto rental chains, oil companies, hotels and restaurants, as well as to large retail stores and chains.

CHARGE-SERV Systems are interrelated-groups of computer programs that facilitate credit control and perform the charge card accounting functions. Written in COBOL for an IBM 360 (Model 30 and higher) computer with at least 65K bytes of core storage, CHARGE-SERV programs can run on other manufacturer's equipment as well and can be readily modified to meet changing demands.

Documentation furnished to users includes a magnetic tape or card deck for reproduction and system and run books for each program. Copies of computer related forms and on site guidance are also provided as part of the Systems.

For more information, circle No. 12 on the Reader Service Card

* * *

A new small magnetic core memory has been placed on the market by Ampex corporation. The core memory, model 3DM-3000, is complete on one plug-in module 1.2 inches high, 7 inches wide and 12 inches deep.

The new Ampex memory is designed for use as a buffer memory in data terminals and acquisition systems, as a data-refresh memory for display devices, and in a wide variety of applications requiring a small, economical core memory where reliability is critical.

The 3DM-3000 ranges in price from $330 to $950 in production quantities, with capacities up to approximately 100,000 bits. It offers an access time of 1.2 microseconds, a half-cycle time of 1.5 microseconds and a full cycle time of 3 microseconds. It uses a 3-D, 5-wire magnetics design for maximum economy and reliability.

The capacity of the 3DM-3000 may be expanded by adding up to eight plug-in modules in an easy-fit metal drawer 5.25 inches high. The basic memory module accommodates 128, 256, 512 or 1,024 words by 4 to 12 bits. The maximum capacity, using all eight modules, is 8,192 words by 12 bits.

For more information, circle No. 13 on the Reader Service Card

A special data terminal announced by OMNITEC Corporation is designed to give each teller direct access to information stored in a bank's central computer.

At the touch of a finger a depositor's balance, interest, payment amounts or other information can be immediately checked. Queries are originated through simple codes and depositors are identified by account number.

The OMNITEC terminal converts keyboard pulses to selective audio tones for data transmission over standard telephone lines to the bank's central computer. Queries are automatically processed and answers are instantly returned in the same manner and printed out on paper tape at the teller's terminal.

The data terminal has a special numeric keyboard for inputting information to a central computer and a strip printer for hard copy printout. The terminal contains its own telephone coupler which uses a standard office phone handset to provide the communications link to and from the computer.

The terminal was originally designed for banking applications, but there is some indication that a variety of other businesses including insurance companies, airlines, libraries, universitites, and stock brokers may find use for the unit.

For more information, circle No. 14 on the Reader Service Card

* * *

Redcor Corporation has announced the development of a **midi-computer, the RC-70**. The 860 nanosecond, 16 bit computer features an 8K memory (plug-in expandable to 32K), memory parity, memory protect, bi-directional index register, high speed multiply and divide, direct memory access, priority interrupt and ASR 33. The machine was designed to fill the gap between the high priced 32-CPU's and the lower priced, lower capacity mini-computers. The RC-70 includes a full complement of operational software, with single pass assembler, FORTRAN IV (ASA standard), math subroutines, utility package, and diagnostics.

For more information, circle No. 15 on the Reader Service Card

"We're looking for the kind of men who can build Comress into one of the largest data processing firms in the country."

About six years ago we started with $2,500 in assets, today it's $40,000,000; and in the not too distant future, we expect to be the largest software house in the data processing field. When we are, the people who'll join us this year will probably be managing their own departments and hiring many more to staff and perhaps manage their own groups.

Comress intends to be the established leader in the country's fastest growing business—computer services.

If this strikes you as ambitious, you're right. If it strikes you as impossible, well, maybe you're not for us and you might as well stop reading right now.

Where we're going

I think few would argue about the growth potential of the computer services field. So let me dwell exclusively on Comress.

First of all, we start with a pretty uncommon record. Our average growth in sales and profits has exceeded 90 per cent per year since we formed six years ago. Our staff numbered 200 last year and will be up to 325 by year end. Our entire growth has been financed from retained earnings, and we've been successful enough to have invested in several affiliated corporations — including Comcet, a new communications computer systems manufacturer; Comnet, one of the few profitable time-sharing firms; Computer Microtechnology, an advanced memory manufacturer; and Commed, a company using computers to serve the medical field.

Our services, too, are uncommon. Initially, we formed the company to develop and market a dynamic simulation program called SCERT—a simulator that would allow our clients to compare the cost, performance and operational aspects of any real or conceptual computer system.

SCERT has become so widely accepted that it has been used by more than 200 top EDP users and by most major hardware manufacturers in the United States and Europe.

SCERT also saved its users enough time to accelerate their need for software development. And they sought out our help. Because of the technical advantages SCERT and our other proprietary systems had given us, Comress was best qualified to help them plan and implement their complete data collection, processing, and communications systems most quickly and most economically.

The future for us

This year marks the introduction of MINI-SCERT. A package designed for the medium and small EDP user. As a result, our client list will grow astronomically. We alone can offer this simulation program assistance. We're so sure of this—that we only quote fixed price contracts—how's that for confidence!

To continue to soar, a company needs only three things — superior products and services, a dynamic and expert management group, and the best people it can possibly find.

We have a clear lead in our products and services, a lead that was highlighted when the founders of Comress, received the 1968 Small Businessman of the Year award. We have developed a management team that few firms can match—some of the best regarded, most successful men in the industry are part of that team. And as for our technical staff—we have a superb group and we're intent on adding only the best.

An indication of the way our people view the future of Comress and the challenge of the work they do, is to look at our employee turnover records. In this past year, we had only three professional people who chose to quit. One formed his own company, another returned to school, the third joined a competitive firm. Few companies in this or any industry have employee satisfaction this great.

The people we need

We're looking for pros—programmers and analysts with an absolute minimum of two to three years experience on third generation systems. The more systems variety and experience, the better.

Everyone we hire must be able to communicate persuasively—in both personal presentations and in writing.

We have immediate openings in Washington, New York, Philadelphia, Chicago, Los Angeles, Pittsburgh and London. We'll soon have more in San Francisco, Houston, Detroit and Atlanta.

The offer we make

You'll start at a top salary; the specific figure depends on your experience and background. And you'll be able to participate in our comprehensive stock option plan.

Full training is provided, and you'll be assigned to the group where your background and experience will show immediate results. We'll work with you for your choice of follow-on assignments to develop your professional skills and meet your personal goals.

Past experience indicates that if you've got it, in an average of two years you will be in line for a management position. Our sustained growth dictates rapid promotion of our good people to positions of increased responsibility. Your speed of growth will be fully up to you.

If Comress sounds right for you and your goals, just send a copy of your resume to me. I'll see that your resume is put into appropriate hands and an interview will be scheduled either in Washington or one of our branch offices. Write: Fred C. Ihrer, President, Comress, 2120 Bladensburg Road, N.E., Washington, D.C. 20018.

COMRESS

better books

by Dennie Van Tassel

Mr. Van Tassel is the head programmer at San Jose State College, California. He has been in data processing since 1957, and handles book reviews for several newspapers in addition to his software duties. He holds a B.A. in English from the University of Southern California, and a M.A. in mathematics from California State College at Los Angeles.

MATHEMATICAL PROGRAMMING IN PRACTICE by E. M. L. Beale London: Sir Isaac Pitman & Sons, Ltd., 1968, $4.80.

This book is based on lectures given on a nine-week course on operational research, run by C-E-I-R Ltd. While the book only requires some elementary algebra and a little matrix notation, the author does state that the book is not designed for anyone who takes a pride in his ignorance of mathematics.

The first part of the book covers the conventional material on linear programming, including the simplex method, transportation method, duality problem, parametric programming, and the inverse matrix method. If this was all the book had, it would just duplicate many good books already out on linear programming. But the author has added a section on organization of linear programming calculations, which attempts fairly successfully to show the reader how to actually set up a linear programming problem.

The last part of the book describes special procedures that go beyond the simplex method for linear programming and are now applied to practical problems. These include adding non-linear constraints or a few integer variables to an otherwise purely linear programming problem.

The last part of the book describes special procedures that go beyond the simplex method for linear programming and are now applied to practical problems. These include adding non-linear constraints or a few integer variables to an otherwise purely linear programming problem.

■ ■ ■

INFORMATION TECHNOLOGY AND SURVIVAL OF THE

Reading List

ADAPTIVE CONTROL AND OPTIMIZATION TECHNIQUES. By Virgil E. Eveleigh. McGraw–Hill. 434 pages.

ALGEBRAIC THEORY OF MACHINES, LANGUAGES, AND SEMIGROUPS. Edited by Michael A. Arbib. Academic Press. 359 pages.

COMPUTERS AND COMMUNICATIONS—TOWARD A COMPUTER UTILITY. Edited by Fred Gruenberger. Prentice–Hall Inc. 219 pages. $10.50.

FINITE-STATE MODELS FOR LOGICAL MACHINES. By Frederick C. Hennie. John Wiley & Sons, Inc. $18.50.

MANAGEMENT OF AUTOMATIC DATA PROCESSING SYSTEMS. By Marvin M. Wofsey. Thompson Book Company. 200 pages.

MATRIX THEORY. By Joel N. Franklin. Prentice–Hall Inc. 292 pages.

RECOGNIZING PATTERNS: STUDIES IN LIVING & AUTOMATIC SYSTEMS. Edited by Paul A. Kolers and Murray Eden. Massachusetts Institute of Technology. 237 pages.

THE TRANSITION TO ON-LINE COMPUTING. Edited by Fred Gruenberger. Thompson Book Company. 209 pages.

USE OF COMPUTERS IN ANALYSIS OF EXPERIMENTAL DATA AND THE CONTROL OF NUCLEAR FACILITIES. Bernard I. Spinrad, Coordinator. Available as CONF–660527 from Clearinghouse for Federal Scientific and Technical Information, National Bureau of Standards, U.S. Department of Commerce, Springfield, Virginia. $3.00.

PROGRAMMING SYSTEMS AND LANGUAGES. Edited by Saul Rosen. McGraw–Hill. 730 pages. $12.50.

CONVERSATIONAL COMPUTERS. By William D. Orr. John Wiley & Sons. 227 pages.

FIRM, by John McLaughlin. Dow Jones–Irwin, Inc. Homewood, Illinois, 1966. $6.50.

Compared to a clerk performing elementary arithmetic operations a computer is ten million times faster and has a cost of 1/100 of 1 percent. The first industrial revolution eliminated manual labor with the use of energy-producing machines, and John McLaughlin, the author, predicts that the second industrial revolution could bring about the elimination of work altogether. McLaughlin bases this prediction on the fact that within the United States well over 90 percent of the effort of the productive capacity of the economy is in information processes. The revolutionary aspect of the electronic information machine is that it has the power to reduce the cost of these operations by a factor of more than 100 to 1.

The author makes several other predictions, such as: The small bank cannot long survive since the major effort in their business is the processing of information and a larger bank is usually much more efficient. Whole industries have become dominated by information processing, such as: banks, warehousing, and research.

The author feels that economical efficiency should be the main goal of computer-oriented efficiency. McLaughlin even expresses the unusual opinion that education and research should be given a lesser emphasis than economical efficiency.

The author has done a lot of research for this book. He uses specific examples ranging from supermarkets to Volkswagons to support his views.

Although the book has much nontechnical information on the effect of modern information processing he offers little advice on industrial survival except that bigness seems to be the most common defense.

■ ■ ■

DIGITAL COMPUTING: FORTRAN IV AND ITS APPLICATION IN BEHAVIORAL SCIENCE, by Richard S. Lehman & Daniel E. Bailey, New York, Wiley & Sons, Inc. 1968.

This book is an introductory book to FORTRAN IV, but is different than most books of this type in that it was written for the behavioral scientist instead of the mathematician or engineer.

The book has twenty-nine FORTRAN programs in the text along with numerous smaller examples including flow charts which illustrate programming techniques of interest to the behavioral scientist. The book is a good text for those that wish to pick up an additional program language for occasional use such as COBOL programmers.

MANAGEMENT OF AUTOMATIC DATA PROCESSING SYSTEMS by Marvin M. Wofsey. Thompson Book Company, Washington, D.C. 1968.

While this book is primarily intended for senior managers of companies, including managers of data processing, programmers and systems analysists will also find that this book offers them some interesting reading. For Wofsey is writing for the informed reader and does not belabor the obvious for the general public. The book concentrates on the problems of organization and communication and explores the desired characteristics of upper level people in data processing.

Subjects explored in this book are presented to the reader concisely in the following chapter headings: Objectives of Data Processing, Organization, Selection of Personnel, System Design, External Relations, Training, Costs, Review and Evaluation Outlook. Mr. Wofsey begins with a brief look at trends in data processing during the last twenty years. He cites that in the 1950's the primary objective of automatic data processing was usually reduction of clerical costs; but now data processing has become a tool to revolutionize management methods.

The merits of selection of personnel from within a company as compared to the advantages of selecting personnel from without the company are given in one chapter. The real question being discussed is whether it is better to select people who know the business but not the computer, or people who know the computer but not the business.

The subject of communications and its relationship between the data processing department and other departments within a company is treated in the chapter, "External Relations". Also given are methods and philosophies of decision influencing, hints on how to instigate cooperation, allay fears, and how to educate people not only so they can furnish better input data but also so they can be helped to obtain more effective uses of data problems. All of these topics are almost universal problems in data processing.

One of the most interesting sections in the book deals with the establishment of methods to review the overall effectiveness of a data processing setup. Suggestions are included on how to select a review committee as well as desirable objectives. A sample review helps complete this section.

The author closes with an outlook, summary and an outstanding bibliography which will be of some value to anyone desiring to do further reading on any of the topics mentioned in the book.

COMPUTERS AND COMMUNICATIONS—Toward a Computer Utility. Edited by Fred Gruenberger. Prentice-Hall. 219 pages. $10.50.

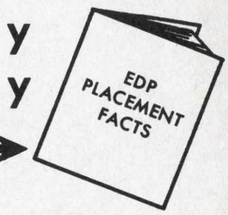
Many of the leaders who have shaped the computer utility concept are represented in this book which was built from a collection of papers which were presented at a Computer Communications Symposium, jointly sponsored by Informatics Inc. and UCLA in 1967.

The articles are very easy to read and cover a wide spectrum of potentials and problems concerning a computer utility. Among the problems discussed are true costs, monopoly, regulation, public welfare, privacy and protection, software and hardware requirements, social implications, Federal regulations, and marketing.

The main concern of the book is the role of the computer utility in business, education, medicine, and in governmental services. This book will provide both fascinating and informative reading to those who are interested in the possibility of a computer utility.

THE TALE OF THE BIG COMPUTER by Olof Johannesson. New York: Coward–McCann, Inc. 1968. 126 pages. $4.00.

The tale here is the history of computer evolution as told by a big nameless computer several computer generations from now. The book is both interesting and humorous and offers the views of the author on



the evolution of future computer systems.

The author knows quite a bit about computers and their potentialities. Information systems, CAI, computerized factories, and computer wrist apparatuses are just some of the subjects covered by the author. The author accurately portrays present accomplishments and then smoothly covers their historical development into the future.

The book has a great deal of humor and social criticism in it because the tale is told by a computer which gives a rather condescending but accurate description of man's shortcomings.

Man is described in an ethnocentric manner that is, in comparison to a computer. Thus man's ability to solve mathematical problems is highly valued and man's ability to use fire is downgraded. The computer ranks man's inventions of electricity and radio as his most important accomplishments since they paved the way for computers.

Time is recorded in nano-seconds and starts at the end of B. C. (Before Computers). In describing the present age the author states that violence had largely been abolished except among gangsters, politicians, and soldiers.

Because of their inability to survive change and overcome their impracticability, big cities are compared with dinosaurs. With the advent of mass information systems and modern transportation, big cities were no longer necessary.

Olof Johannesson presents the following rather amusing version of how governmental bureaucracy opposed all measures to eliminate itself and be replaced by a computer but at the same time fell further and further behind in its decision making. Since most of the efforts of the bureaucracy were directed in maintaining their positions it was forced to keep delegating more and more responsibility to the computer and was finally replaced automatically by the computer. The bureaucracy made a desperate attempt to retain power, but as none of the members knew how to program a computer the attempt failed.

This book offers astute reflection of both the present and future growth of computers and should prove very interesting, enjoyable reading. ■

market place

where you can reach **125,000** programmers, mathematicians, analysts and EDP managers.

CLASSIFIED ADS in software age

CLASSIFIED ADVERTISING

Non-Display Classified: For firms or individuals offering commercial products or services, 75¢ per word (including name and address). Minimum order $7.50. For Blind Ad Service, an additional $10.00. For "Positions Wanted" Ads, 55¢ per word (including name and address). No minimum. **Payment must accompany copy** except when ads are placed by accredited advertising agencies. Frequency discounts: 5% for 6 months; 10% for 12 months paid in advance.

Display Classified: One inch by one column, $70.00. Column width 2¼". Photographs accepted for an additional $20.00. Advertiser to supply all photo, art, cuts, or camera ready copy.

General Information: One inch display Help Wanted and Employment Services ads **will be** accepted in the classified section. Employment ads ⅛ page or more will appear run of book, will be keyed to the resume form in back of publication, and will qualify to free daily resume service.

Closing Date: 1st of preceding month (for example, May issue closes April 1st).

Send order and remittance to: Classified Dept., SOFTWARE AGE, P. O. Box 2076, 2211 Fordem Avenue, Madison, Wisconsin 53701.

software age

P. O. Box 2076 • 2211 Fordem Ave. • Madison, Wis. 53701

CLASSIFIED ADVERTISING ORDER FORM

Please refer to the above information for complete data concerning terms, frequency discounts, closing dates, etc. Cash with order.

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25
26	27	28	29	30
31	32	33	34	35

_______Words { @ 55¢ (Reader Rate) / @ 75¢ (Commercial Rate) } = $_______

_______Include Photograph @ $20.00 additional (**Display Ads Only**)

☐ Run Ad Blind $10.00 Insert_______time(s) Total Enclosed $_______

NAME_______

ADDRESS_______

CITY_______ STATE_______ ZIP_______

Signature_______

WORD COUNT: Include name and address. Name of city (Des Moines) or of state (New York) counts as one word each. Zone or Zip Code numbers not counted. (Publisher reserves right to omit Zip Code if space does not permit.) Count each abbreviation, initial, single figure or group of figures or letters as a word. Symbols such as 35mm, COD, PO, AC, etc., count as one word. Hyphenated words count as two words.

SA-2

The Classified Ad Department

HELP WANTED

PART TIME PROGRAMMERS/ANALYSTS with COBOL experience in such areas as management information systems, inventory control, accounting, personnel, or general manufacturing or information retrieval applications. Degree not necessary. All work will be corresponded through the mail on high contract rates. Send brief resume to Box SA-401, Software Age, P. O. Box 2076, 2211 Fordem Ave., Madison, Wis. 53701.

EMPLOYMENT AND SEARCH COMPANIES

PROGRAMMERS **360 COBOL**

Jrs. $8-10M, Srs. $16-22M, Intermediates $10-15M

Contact R. Lowy who has had 10 yrs. exp. in programming & systems. We specialize in data processing placements. Resume not necessary. Co. pays entire agency fee plus travel expenses.

ERIN EXECUTIVE AGENCY
132 W. 34 St., New York, N. Y. 10001
Phone: (212) 594-8296

We have been placing B.S., M.S. and Ph.D.

SCIENTIFIC PROGRAMMERS

in **fee paid** positions throughout the U.S. since 1959. Send resume today or request confidential application. We are graduate engineers, working full time . . . for you.

ATOMIC PERSONNEL, INC.

Suite F2, 1518 Walnut St., Phila., Pa. 19102

AN EMPLOYMENT AGENCY FOR ALL TECHNICAL FIELDS

DATA PROCESSING POSITIONS IN THE SOUTHWEST

Contact:

DATA PROCESSING CAREERS

Richard Kemmerly, Suite 1109, Stemmons Towers West, Dallas, Texas 75207. 214/637-6360.

PROFESSIONAL SERVICES

CONSULTANT: Excellent experience designing, programming, debugging and documenting commercial applications. Hourly or job basis. Box 11-H, 272 First Ave., N.Y.C. 10009.

CONTRACT PROGRAMING in PL/I or IBM 360 Assembler BOS or DOS. Also, 1401 and H200 programs rewritten for 360. Full documentation. Advanced Systems Analysis and Programing, 4221 East Thompson Road, Indianapolis, Indiana. Phone (317) 786-0021.

PRODUCTS

ADD / SUBTRACT 6-DIGIT HEXADECIMALS in seconds with 100% accuracy. The pocket HEXADDER, $15. Free brochure from HEXCO, Dept. SA, P. O. Box 55588, Houston, Texas 77055.

software age

CONFIDENTIAL INQUIRY

Your original copy of this form will be retained at the offices of SOFTWARE AGE and will be used for no other purpose than to notify the specific firms which you have checked (on the reverse side) of your interest.

THIS INQUIRY IS IN DIRECT RESPONSE TO YOUR ADVERTISEMENT IN SOFTWARE AGE MAGAZINE

TYPE OR PRINT CLEARLY FOR PHOTO REPRODUCTION

JOB DESIRED: ______

List computer hardware knowledge (names of systems, tape, disk, terminals, etc.): ______

Programming specialties and years of experience (commercial, scientific, theoretical, experimental, analog, etc.): ______

Systems programming on which you have had development experience (compilers, assemblers, executives, monitors, O.S., etc. Indicate for what computer): ______

Programming languages used and extent of experience (COBOL, FORTRAN, etc.): ______

Applications programmed (aerospace, banking, insurance, math subroutines, compilers, etc.): ______

Systems analysis experience (card design, flow charting, operation analysis, etc.): ______

EDP management experience (include years and number of people reporting to you): ______

SALARY: ______ (current) ______ (desired) **DATE OF AVAILABILITY:** ______

EDUCATION: Indicate major as well as degree unless self-explanatory.

Degrees			
Years			
Schools			

EMPLOYMENT: Indicate present employment and previous jobs below.

Employer			
City			
Years	to	to	to
Title or Function			

Name ______ Age ______

Home Address ______ Home Phone ______

______ (city) ______ (state) ______ (ZIP code) U.S. Citizen? ______

Security Clearance ______ Location Preference ______

Marital Status ______

Military Status ______

BE SURE YOU HAVE CHECKED ON REVERSE SIDE THE COMPANIES YOU WANT TO SEE THIS INQUIRY. PUT FORM IN STAMPED ENVELOPE AND MAIL TO:

software age
MAGAZINE
P. O. BOX 2076
2211 FORDEM AVE., MADISON, WIS. 53701

check your interests here

Fill in the confidential inquiry form on the other side of this sheet. This form provides all the information advertisers require to screen applicants. If further information is desired, you will hear from the advertiser direct. Then, check below the boxes of those companies to which you want copies of your form sent. Mail to SOFTWARE AGE, P.O. Box 2076, 2211 Fordem Avenue, Madison, Wisconsin 53701. **(Please do not send us your own resume. We will only process this form. A new form must be filled out for each issue in which you are answering ads.)**

	Page
☐ 1. Allis–Chalmers	55
☐ 2. Army Air Force Exchange Service	54
☐ 3. Auerbach Corp.	3rd Cover
☐ 4. Blue Cross–Blue Shield	16
☐ 5. Booz • Allen Applied Research Inc.	34
☐ 6. Burroughs Corp.	21
☐ 7. C. P. Clare & Co.	46
☐ 8. Collins Radio Co.	40
☐ 9. Comress	51
☐ 10. Conductron Corp.	22
☐ 11. Electronic Associates, Inc.	39
☐ 12. First National Bank of Chicago	12
☐ 13. General Electric Co., Information Service	37
☐ 14. Hartford Insurance Group	25
☐ 15. Honeywell, Inc., Electronic Data Processing Div.	18
☐ 16. Household Finance Corp.	38
☐ 17. Hughes Aircraft Co., Ground Systems Group	24
☐ 18. IBM Corp.	29
☐ 19. Litton Industries, Mellonics Systems Development Div.	60
☐ 20. Lockheed—California Co.	23
☐ 21. Lockheed–Electronics Co.	14
☐ 22. Lockheed Missile & Space Co.	30
☐ 23. National Cash Register, Electronics Div.	56
☐ 24. Pratt & Whitney Aircraft	44
☐ 25. Raytheon Co., Equipment Div.	4th Cover
☐ 26. RCA Corp.	3
☐ 27. Scientific Data Systems	36
☐ 28. Sikorsky Aircraft	15
☐ 29. Singer–General Precision, Inc., Link Div.	5
☐ 30. TRW Systems Group	38
☐ 31. U.S. Army Aviation Systems Command	53
☐ 32. Univac Corp.	26 & 27
☐ 33. Vitro Laboratories	10
☐ 34. Westinghouse Information Systems Div.	41
☐ 35. Xerox Corp.	31

EMPLOYMENT AND SEARCH AGENCIES

☐ 36. American Computer Personnel	11
☐ 37. Callahan Center for Computer Personnel	55
☐ 38. Career Center	47
☐ 39. Computer Careers, Inc.	11
☐ 40. Computer Personnel Agency, Inc.	54
☐ 41. Computer Personnel Consultants, Inc.	45
☐ 42. Data Management Services, Inc.	11
☐ 43. Drew Personnel Agency	34
☐ 44. Robert Half Personnel Agencies	38
☐ 45. Input, Inc.	23
☐ 46. Interstate Staffing, Inc.	57
☐ 47. Everett Kelley Associates, Inc.	34
☐ 48. LaSalle Associates, Inc.	54
☐ 49. Lawrence Personnel	34
☐ 50. Management Scientists, Inc.	46
☐ 51. National Manpower Register	2nd Cover
☐ 52. RSVP Services	25
☐ 53. Sheridan Associates	45
☐ 54. Source EDP	59
☐ 55. WCP Information Systems Personnel	55
☐ 56. Welles Recruiting Systems, Inc.	23

PRODUCTS AND SERVICES
(Use Reader Service Card)

Anaheim Publishing Co.	4
Box SA-302	46
Software Age Resume Center	6
Software Systems Div.	45
Visual Control Associates	14
John Wiley & Sons	49

☐ **I do not now receive S/A. Please enter my FREE subscription.**

....
Name   Street Address

....
City   State   Zip Code

................................/................................
Prime Experience in What Industry   My Specialty

☐ Technical Degree   ☐ Non-Technical Degree   ☐ No Degree

....
Year Born   ☐ I Have Analog/Hybrid Experience

software age

P. O. Box 2076
2211 Fordem Avenue
Madison, Wisconsin 53701