March 1964

NEWSLETTER

Committee on Computers in Research
Federal Reserve System

A Note from the Editor

On February 28, 1964, the Committee on Computers in Research took several steps forward in its program of instruction, cooperation, and exchange of information on computer-related activities within the Federal Reserve System. Among its actions, it placed the Newsletter on a bimonthly basis and increased the reportorial responsibilities of Committee members. In taking these steps, it is striving to develop the Newsletter into an effective tool through which individuals and institutions within the System may communicate. By providing a forum for exchange of information about programming needs, programming plans, programs being written, data needs, and research results, the Committee believes that the Newsletter will be a vital supplement to the Research Program Library, which by its nature is limited to only one phase of computer-related activity.

In accordance with the wishes of the Committee, your Editor has scheduled issues of the Newsletter for January, March, May, July, September, and November. The May 1964 issue will be the first to reflect the greater scope of reporting by Committee members.

* * * * *

Programmers, economists, and others in the Federal Reserve System are invited to submit contributions to the Newsletter. Contributions may consist of program routines, programming techniques, computer applications for economic and statistical research, and similar matters of interest to System personnel.

Contributions may be submitted to members of the Committee on Computers in Research or to Emanuel Melichar, Economist, Division of Research and Statistics, Board of Governors.

CONTENTS

NEWS NOTES

Boston

In late March, we expect delivery of a 1443 printer for our 1620 computer. This printer will have a speed of 150 lines (120 characters) per minute and uses a 52 character type bar. This will be a vast improvement over the present typewriter output; the time savings will enable us to nearly double our workload which is presently nearing capacity on our 8-hour workday.

Most of our recent work has involved updating and improving existing programs in addition to the regular flow of work. We would like to get a program for the Durbin-Watson statistics written for a 1620, 20 K machine. If anyone has one, we would appreciate it.

Stephan Lofgren
John J. Arena

Atlanta

We have completed a 4K 1401 program that computes an exponential trend and related ratios and changes for a series containing up to 999 observations.

Frank Edwards

Richmond

Another research assistant, a math major, has completed the IBM course on the 1401 computer and has been assigned to Research programming. It is contemplated that our two research assistants will spend approximately three-quarthers of their time on programming and related fields.

Our program for computing the annual rate of growth, based on an exponential curve, has been used extensively. This program incorporates the Board's logarithm subroutine (file No. 5.10.01.0).

Minor changes have been made in the departmental program for the deletion of stocks data and the program for printing operating ratios slip sheets to include revised 1962 ratios with those for 1963. A number of programs, primarily listings, have been required for the special study on Banking Markets and for the change-over of the debits series to a metropolitan base. Our programs for the F.R. 416 report and for the electric power series, consequently, have not been completed.

Elizabeth W. Angle

Minneapolis

Projects being considered for the future include work on seasonal adjustment, forecasting by use of exponential smoothing and development of an industrial production index. By July of this year we expect to be equipped with a new IBM 1620 system which will include a 1443 ON-LINE PRINTER and a 1311 DISK STORAGE DRIVE. This will permit use of FORTRAN II-D, and major programming tasks will probably be delayed to take advantage of the more powerful equipment and language.

Ronald E. Kaatz

ERRATA

The following corrections should be made in the August 23, 1963 issue of the Newsletter:

On page 14, the 12th line should read:

"by Autocoder, because our undebugged program deck is in Autocoder"

On page 24, the 4th line of the second note (middle of page) should read:

"was a signed field of zeros and so the comparison with an unsigned"

FORTRAN TRAINING FOR THE RESEARCH DEPARTMENT

In looking at the growth of FORTRAN popularity, I wonder if some members of the Computer Committee are considering FORTRAN training courses for their own research staff. If they are, they might be interested in the following report covering our recent experience.

In the interests of brevity, many of the details of the course have been omitted. Anyone undertaking a program of training might want to ask questions about these details. I'll be happy to help them as much as I can.

Edward C. Christ,
Supervisor Computer Section
Federal Reserve Bank of Philadelphia

* * *

During the week of September 23, 1963, the Research Department of the Federal Reserve Bank of Philadelphia undertook a FORTRAN training program with the view of making computer results, particularly ad hoc operations, more easily available to our economists and their assistants.

The course was conducted on site at the Bank, and required the full week. Economists and their research assistants comprised the bulk of the class, which also included the two full-time research programmers, one person from Data Processing and one from the Bank Examination Department. Eleven persons took the complete course; five others sat in for varying periods of time.

The teaching chore was divided between a representative of IBM and Edward Christ from Research. The IBM instructor explained the FORTRAN instructions and the compiler. Mr. Christ demonstrated the use of the instructions in three different types of problems.

The instructor-team approach was suited to our purposes for several reasons. For one thing, IBM did not have available an instructor who was familiar with the types of problems found in economic research. The man assigned to us was at a loss for concrete, relevant examples, so someone here had to fill this gap.

Secondly, IBM's usual method of teaching FORTRAN relies completely on the student teaching himself from a set of books. We felt that a better result would be achieved if the learning process was abetted by lectures from an instructor. For this reason our course was designed for instruction by people competent to handle the technical questions thrown by the experienced programmers.

A condensed outline of the course would read like this:

Monday - Tuesday	Explanation of the FORTRAN instructions and the compiler.
Wednesday - Thursday (A.M.)	Demonstration problems in data processing--moving averages, index numbers, row-column-totals. Variation in technique for each.
Thursday (P.M.)	Students write programs for moving averages.
Friday	Individual instruction for each student at the computer.

Three of the 11 students who completed the course could point to successful programs by the end of the week. At least five others had written their programs and were in various stages of debugging them. These programs were finished the following Monday.

Whether the course was successful is still an unanswered question. Judging by the students' comprehension of FORTRAN as manifested in their programs, the course came up to our expectations. But the significant criteria will be whether interest is sustained and the analytical staff increases its use of the computer. It will take several months before we can gauge this.

A DESIRABLE ADDITION TO ABSTRACTS OF FORTRAN PROGRAMS*

FORTRAN has many qualities to recommend its use. One of these is terseness. Large numbers of SPS instructions can be wrapped up in a single statement. For example, the enclosed program reduces over 900 machine-language instructions to 29 FORTRAN statements. A program that might take 7-8 pages is printed on one.

This quality of terseness can be an advantage in swapping programs around the System. It allows us to distribute along with the abstracts a complete program listing. With the abstract and the listing one can see for himself whether the program will help him. He can tell what changes to make to adapt the program to his own particular needs. He can answer a lot of questions for himself. He can save himself the trouble of contacting the program's author.

Furthermore, he won't have to wait for the author to duplicate cards. He can punch his own. No waiting for two days or longer for program cards. He can be ready to run in--how long does it take to punch 30-35 cards and compile your own program?

To give you an idea what you can expect, there's an abstract that includes a listing shown on the next two pages. I would strongly urge anyone in the System who has written a FORTRAN program to submit a program listing with his abstract. And I think both should be distributed for filing in the Computer Committee's Library Index.

---

*Contributed by Edward C. Christ, Supervisor, Special Studies and Computer Section, Federal Reserve Bank of Philadelphia.

FEDERAL RESERVE SYSTEM RESEARCH PROGRAM LIBRARY ABSTRACT

Program: Exponentially - Weighted Input Generator (E-WIG) File No. 2.03.144.0

Originating Institution: Federal Reserve Bank of Philadelphia Date: September 11, 1963

Programmers and Collaborators: E. C. Christ

Supply the following information and other pertinent data in the space below.

1. Description, function, and/or purpose
2. Methods employed
3. Equipment required 1401:8-K
4. Programming language FORTRAN (see Appendix)

E-WIG generates data for correlation analysis. It is to be used in cases where one variable leads or lags another and where the lead or lag is distributed over several time periods.

The program constructs sixteen new time series from one original series. Each item in each new series is a weighted average that incorporates the observations of the corresponding period in the original series so as to give most weight to the current period, and successively less weight for the periods going back (or forward) in time. When these weights are plotted, they approximate an exponential curve, hence the title.

Whether the distribution of the weights produces a leading or a lagging series depends on the sequence with which the original data is read into the computer. If a leading series is desired, the data should be read in reverse order; i.e., January of the first year occupying the last position in the series and December of the last year in first position. For a lagging series the data should be read in the natural order; i.e., January of the first year read in first. In either case the program need not be altered.

Computing the series takes place after the data is read. Beginning with the first position each observation is weighted to produce the first item of the first new series:

$$Z_1 = BX_1 + (B)(1-B)X_2 + (B)(1-B)^2 X_3 + \ldots + (B)(1-B)^n X_{n+1}$$

where the subscripts refer to particular items in the series and the coefficients of X are normalized weights.

The number of items making up each Z depends on the weights. When they add to .99 the value of Z is printed. The next Z is then computed. This time the observations to be weighted are the second one and its successors. So that

$$Z_2 = BX_2 + (B)(1-B)X_3 + (B)(1-B)^2 X_4 + \ldots$$

After the last Z has been computed and printed, the value of B (originally set at .20) is increased by .02 and a new series is computed.

This process is repeated until B reaches a value of .50. When the sixteenth series is finished, the machine prints END OF JOB. The new series can then be punched for empirical testing with a correlation program.

continued on next page

FEDERAL RESERVE SYSTEM RESEARCH PROGRAM LIBRARY ABSTRACT

Program: Exponentially-Weighted Input Generator (continued) File No. 2.03.14.0 (continued)

Originating Institution: Date:

Programmers and Collaborators:

Supply the following information and other pertinent data in the space below.

1. Description, function, and/or purpose
2. Methods employed
3. Equipment required
4. Programming language

```
SEQ    STMNT    FORTRAN STATEMENT
  1             DIMENSION X%144¤
  2             READ 50,%X%I¤,I#1,144¤
  3             B#.20
  4             L#120
  5        8    N#1
  6             K#0
  7             A#1.-8
  8             DO 5 I#1,L
  9             G#B*A
 10             D#B£G
 11             Z#X%I¤*B
 12        3    Z#Z£X%I£N¤*G
 13             N#N£1
 14             G#G*A
 15             D#D£G
 16             IF%D-.99¤ 3,4,4
 17        4    K#K£1
 18             N#1
 19        5    PRINT 51,K,B,Z
 20             L#L£1
 21             PRINT 53
 22             B#B£.02
 23             IF%B-.50¤8,8,9
 24        9    PRINT 52
 25       50    FORMAT%8X,12F6.0¤
 26       51    FORMAT%1X,1HZ,2X,I3,2X,F6.2,5X,F6.0¤
 27       52    FORMAT%1H0,10HEND OF JOB¤
 28       53    FORMAT%1H0¤
 29             END
```

Program Abstracts Received by the Research Program Library
January-February 1964

1.03.06.0 Monthly Department Store Series--Consumer Credit Report
Francis Byers, Richmond

1.04.01.0 Department Store Report on Revolving Credit
Robin Cramme', Richmond

1.05.08.0 Generate 3-Digit SIC Code Summary Cards
Ron Kaatz, Minneapolis

2.02.12.0 Phase II of Census II Seasonal Adjustment
Fritz Biermeier, Minneapolis

2.02.13.0 Ratio of Monthly Data to Average of Preceding and Following Months
Martha Bethea, Atlanta

2.03.12.0 Regression of De-Trended Series with Lead-Lag Option
Ron Kaatz, Minneapolis

2.03.13.0 Simple Linear Regression and Correlation with Lead-Lag Option
Fritz Biermeier, Minneapolis

2.04.07.0 Annual Rate of Growth (of time series)
Robin Cramme', Richmond

2.04.08.0 Expontential Trend
Martha Bethea, Atlanta

2.07.16.0 Miscellaneous Statistics Including Variable Term Moving Average
Fritz Biermeier, Minneapolis

2.07.17.0 Percent of Co-Movements (of time series)
Ron Kaatz, Minneapolis

2.07.18.0 Profitability of a Cash Flow Stream (return to investment)
Ron Kaatz, Minneapolis