

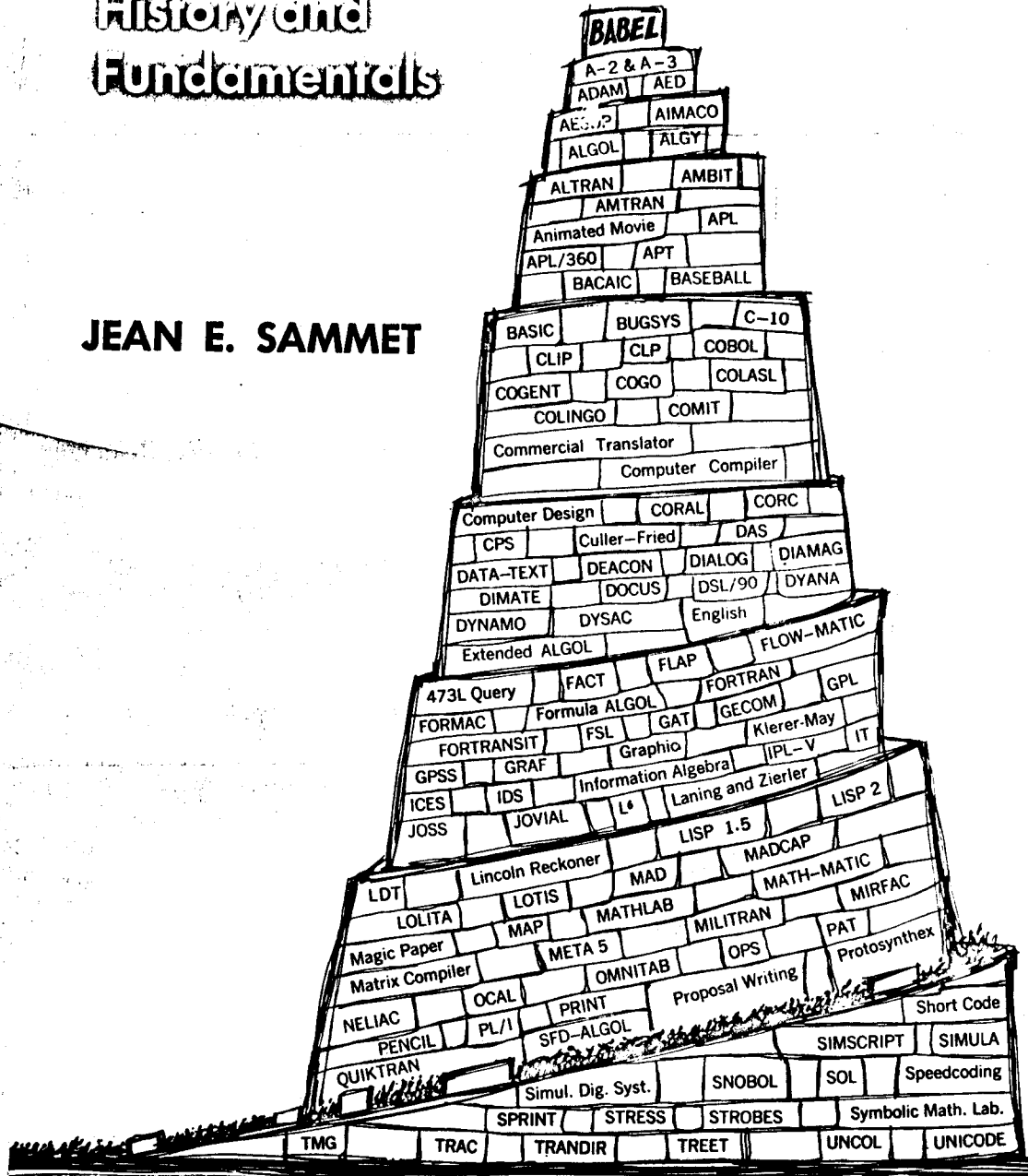
SAMMET

# PROGRAMMING LANGUAGES:

## History and Fundamentals

JEAN E. SAMMET

PROGRAMMING LANGUAGES:  
History and Fundamentals



Prentice-Hall

PRENTICE-HALL SERIES IN AUTOMATIC COMPUTATION

## SAMPLE PROGRAM—JOVIAL

**Problem:** Construct a subroutine with parameters  $A$  and  $B$  such that  $A$  and  $B$  are integers and  $2 < A < B$ . For every odd integer  $K$  with  $A \leq K \leq B$ , compute  $f(K) = (3K + \sin(K))^{\frac{1}{2}}$  if  $K$  is a prime, and  $f(K) = (4K + \cos(K))^{\frac{1}{2}}$  if  $K$  is not a prime. For each  $K$ , print  $K$ , the value of  $f(K)$ , and the word PRIME or NONPRIME as the case may be.

Assume there exists a subroutine or function PRIME ( $K$ ) which determines whether or not  $K$  is a prime, and assume that library routines for square root, sine and cosine are available. This program also assumes the existence of three output routines. [Note: JOVIAL has ODD as a primitive.]

Program:

```

PROC   SPEC (A1, B1)$  ITEM A1 I U 47 $
                                ITEM B1 I U 47 $
BEGIN  WRITE (0) $
        IF NOT ODD (A1)$
        A1 = A1 + 1$
        FOR K = A1, 2, B1$
            BEGIN  WRITEN (15, K, 0)$
                    IFEITH PRIME (K) $
                    BEGIN WRITEN (30, SQR, (3*K + SIN (K)), 5)$
                            WRITEH (45, 5H(PRIME))$
                    END  ORIF 1 $
                    BEGIN WRITEN (30, SQR (4*K + COS (K)), 5) $
                            WRITEH (45, 8H(NONPRIME))$
                    END
            END  WRITE (1) $
        END  WRITE (4) $
END

```

The language serves simultaneously as a reference, publication, and hardware language. JOVIAL was designed for the professional programmer and definitely to be used in a batch environment. However, a much later and much simpler version called JTS (see Sandin and Foote [SN65]) was installed under SDC's time-sharing system, and an interpretive extended subset version called TINT was specifically designed and implemented for on-line use. (See Kennedy [KE 65].)

JOVIAL has had the misfortune to suffer throughout its history from all the problems that could possibly arise from an attempt to have wide usage, maintain compiler independence, avoid dialects, and control subsetting and extensions. The proliferation of documents and systems on differing machines did not help the situation, although there were continuous attempts in SDC to control this problem. The earliest description seems to be the one by Schwartz, Petersen, and Olson [SC60]. The reader interested in pursuing which versions existed on which machines should see the papers by Shaw [SH63b] and Steel [ST66], but even these are not complete. (See also Figure

VIII-1.) There have been several the latest one by Perstein [PE more positive side, in August, were made in SDC with the release of any new JOVIAL compiler manual. If the new compiler (JOVIAL 3), it must implement the specification given in Perstein [PE66a]. These are not included in the specifications of JOVIAL, but are dependent, e.g., precision of definitions have been given in a little problem of incompatibility.

Until 1967 there was no standard viewpoint of American standards, obviously there has been tremendous interest in itself. As a result of interest by that a USASI standard might be developed.

The original CLIP language, Book, and the former supercomputer system.<sup>4</sup> Since then, numerous further development of JOVIAL directly involved with the impact of groups to control the maintenance.

The basic objective of the system by professional programmers in processing problems. In the various notations for defining syntax and arbitrary notation that in my opinion, whatsoever, the latter appearing.

One complaint which no activity is a shortage of documents of JOVIAL weighs almost. Since that was written in 1962, decreased in quantity. Naturally, papers of interest to limited groups of it is of widespread interest. Kennedy [KE62]) to detailed description and tutorial. Other references are listed at

<sup>4</sup> Shaw [SH00].

<sup>5</sup> Shaw [SH63b], p. 90.

ment is the creation and clear usage of the language (SL) and intermediate language has been entirely about the source language resembles ALGOL 60. On the other hand, LISP 1.5. IL is designed to retain the same structure as the user and system programs. Thus, the user programs look like data, and the system programs look like data at the source language level. There are macro expansions which are useful com-

portable onto a new machine, and in fact, in any case, since earlier versions of LISP, this facility can certainly be carried over to the source language level and, presumably, also at the

language on the compiler is the reason, because a person who merely has no background whatsoever of LISP tends to develop an internal image of what has been done to improve the language by single words rather than to pack the language with a large collection in LISP, it was the subject of this issue in Weizenbaum

and usage, no comment along those lines, if any—further work will be done in the near future, immediate change.

REFERENCES

"A Unification of ALGOL-COBOL, the Question of One or Several Languages for Data Processing, Gordon and Breach,

"Combining ALGOL and COBOL", *Proc.*

"Minutes of 8/67", *Appendix 1, Minutes of 8/67*.

- [AF67] *Standard Computer Programming Language for Air Force Command and Control System* (CED 2400), Air Force Manual AFM 100-24 (June, 1967).
- [CE67] Clark, E. R., "On the Automatic Simplification of Source-Language Programs", *Comm. ACM*, Vol. 10, No. 3 (Mar., 1967), pp. 160-65.
- [CO61] Coffman, E. G., Jr., *A Brief Description and Comparison of ALGOL and JOVIAL*, System Development Corp., FN-5618, Santa Monica, Calif. (June, 1961).
- [KE62] Kennedy, P. R., *A Simplified Approach to JOVIAL* (A Training Document), System Development Corp., TM-780/000/00, Santa Monica, Calif. (Sept., 1962).
- [KE65] Kennedy, P. R., *The TINT Users' Guide*, System Development Corp., TM-1933/000/02, Santa Monica, Calif. (Mar., 1965).
- [KK65] Klein, S., "Automatic Paraphrasing in Essay Format", *Mechanical Translation*, Vol. 8, Nos. 3 and 4 (June, Oct., 1965), pp. 68-83.
- [MD64] Marsh, D. G., "JOVIAL in Class", *Annual Review in Automatic Programming*, Vol. 4 (R. Goodman, ed.). Macmillan, New York, 1964, pp. 167-81.
- [MI62] *FAST-FORTRAN Automatic Symbol Translator* (reference manual), MITRE Corp., SR-24, Bedford, Mass. (Jan., 1962).
- [PE66] Perstein, M. H., *Grammar and Lexicon for Basic JOVIAL*, System Development Corp., TM-555/005/00, Santa Monica, Calif. (May, 1966).
- [PE66a] Perstein, M. H., *The JOVIAL (J3) Grammar and Lexicon*, System Development Corp., TM-555/002/04, Santa Monica, Calif. (May, 1966).
- [SC60] Schwartz, J. I., Petersen, K. E., and Olson, W. J., *JOVIAL and its Interpreter, A Higher Level Programming Language and an Interpretive Technique for Checkout*, System Development Corp., SP-165, Santa Monica, Calif. (Apr., 1960).
- [SH00] Shaw, C. J., *Programming Languages and JOVIAL*, System Development Corp., BR-3/11-60, Santa Monica, Calif.
- [SH61] Shaw, C. J., "System Development Corporation's Procedure-Oriented JOVIAL", *Datamation*, Vol. 7, No. 6 (June, 1961), pp. 28-32.
- [SH61a] Shaw, C. J., *The JOVIAL Manual*, pt. 3, *The JOVIAL Primer*, System Development Corp., TM-555/003/00, Santa Monica, Calif. (Dec., 1961).
- [SH63] Shaw, C. J., "A Specification of JOVIAL", *Comm. ACM*, Vol. 6, No. 12 (Dec., 1963), pp. 721-36.
- [SH63a] Shaw, C. J., "JOVIAL—A Programming Language for Real-time Command Systems", *Annual Review in Automatic Programming*, Vol. 3 (R. Goodman, ed.). Pergamon Press, New York, 1963, pp. 53-119.
- [SH63b] Shaw, C. J., "JOVIAL and Its Documentation", *Comm. ACM*, Vol. 6, No. 3 (Mar., 1963), pp. 89-91.
- [SH64] Shaw, C. J., *A Comparative Evaluation of JOVIAL and FORTRAN IV*, System Development Corp., N-21169, Santa Monica, Calif. (Jan., 1964).
- [SN65] Sandin, N. A. and Foote, E. B., *JTS User's Manual*, System Development Corp., TM-1577/000/01, Santa Monica, Calif. (Apr., 1965).
- [ST66] Steel, T. B., Jr., *Some Observations on the Relationship Between JOVIAL and PL/I*, System Development Corp., TM-2930/000/01, Santa Monica, Calif. (May, 1966).

## 750 BIBLIOGRAPHY ARRANGEMENTS AND AUTHOR LIST

- Roos, D. [RS65], 694, 620; 759; [RS67], 694; [RS67a], 694; 759.  
 Roos, D., *et al.* [RS64], 694, 612, 613; 756.  
 Rose, A. J. [RJ66], 310, 247.  
 Rosen, S. [RO61], 303; 153; [RO64], 697; [RO64a], 740; [RO67], 740.  
 Rosen, S., *et al.* [RO65], 303, 164.  
 Ross, D. T. [RD61], 705, 681; [RD62], 705, 682; [RD67], 705, 681; [RD67a], 705, 682, 683.  
 Ross, D. T., *et al.* [RD63], 705.  
 Roth, J. P., *et al.* [RQ67], 310, 247.  
 Ruyle, A., *et al.* [RU67], 308, 217, 240, 245, 254, 258.  
 Sakoda, J. M. [SA65], 303, 164; 465, 388.  
 Sammet, J. E. [SM61], 306; 379, 380, 330; 598; [SM61a], 63, 56; 380, 340, 344; [SM61b], 380, 332; [SM61c], 381, 332; [SM62], 381, 345; [SM66], 465, 385; 520, 472; [SM66a], 520, 471, 472; [SM66b], 61, 35; 720, 715; 737; [SM67], 520, 521, 471, 472; [SM67a], 465, 385; 520, 472; [SM68], 740, 753.  
 Sammet, J. E., *et al.* [SM64], 521, 476, 486.  
 Sanders, N., *et al.* [SS63], 306.  
 Sandin, N. A., *et al.* [SN65], 599, 528.  
 Satterthwait, A. C. [SR66], 465.  
 Savitt, D. A., *et al.* [SK67], 702.  
 Saxon, J. A. [SX63], 381, 345.  
 Saxon, J. A., *et al.* [SX66], 307.  
 Scheff, B. H. [SD66], 699, 648, 649; 757; [SD66a], 699, 648.  
 Schlaeppli, H. P. [QY64], 695, 621; 761.  
 Schlesinger, S., *et al.* [QL67], 28, 20; 313, 299.  
 Schneider, F. W., *et al.* [QV64], 698.  
 Schorré, D. V. [QT64], 602, 592; 698.  
 Schwalb, J. [SB63], 379, 329.  
 Schwartz, J. I. [SC65], 29, 23.  
 Schwartz, J. I., *et al.* [SC60], 599, 528.  
 Schwarz, H. R. [QN62], 306, 181.  
 Schwinn, P. M. [UZ67], 704, 677; 759.  
 Sconzo, P., *et al.* [SO65], 521, 474.  
 Seitz, R. N., *et al.* [UV67], 311, 258, 259, 263; 754.  
 Selfridge, R. G. [YD55], 696, 627.  
 Shantz, P. W., *et al.* [SY67], 303, 164.  
 Shaw, C. J. [SH00], 599, 525, 529; [SH61], 599, 529; [SH61a], 599; [SH62], 29, 23; 126, 65; [SH63], 599; 760; [SH63a], 599, 529; [SH63b], 599, 524, 525, 528, 529, 530; [SH64], 599, 530; [SH64a], 29; [SH65], 377; [SH66], 29, 19, 23.  
 Shaw, J. C. [JC64], 308, 217, 218; 760; [JC65], 308, 220.  
 Shaw, J. C., *et al.* [JC58], 466, 389.  
 Sibley, R. A. [QS61], 697.  
 Siegel, M., *et al.* [SG62], 380.

- Simmons, R. F. [SE65], 702, 703.  
 Simmons, R. F., *et al.* [SE63], 70.  
 Singman, D., *et al.* [SI65], 307, 2.  
 Skinner, F. D. [QX66], 704.  
 Slagle, J. R. [SL61], 468, 410; [SI65], 307, 2.  
 Spiegel, J., *et al.* [SP65], 740.  
 Spitzer, J. F., *et al.* [SZ65], 702, 6.  
 Standish, T. A. [QM67], 126.  
 Steel, T. B., Jr. [ST57], 29, 8; [ST66], 525, 528, 530 [ST66a], 4.  
 Stefferud, E. [SF63], 467, 393.  
 Steil, G. P., Jr. [QZ67], 702, 666;  
 Stone, H. S. [TS67], 306, 192.  
 Stone, P. J., *et al.* [SJ63], 468, 43.  
 Stotz, R. H. [UU63], 310, 240; 70.  
 Stotz, R. H., *et al.* [UU67], 310, 240.  
 Stowe, A. N., *et al.* [SW66], 310, 240.  
 Strachey, C. [SQ65], 470, 450.  
 Strachey, C., *et al.* [SQ61], 126, 9.  
 Strong, J., *et al.* [QR58] and [QR67], 70.  
 Summers, J. K., *et al.* [UT67], 70.  
 Summit, R. K. [UW67], 702.  
 Sutherland, I. E. [QW63], 470, 46.  
 Sutherland, W. R. [SU66], 470, 46.  
 Swets, J. A., *et al.* [UY63], 702.  
 Swigert, P. [SV66], 521.  
 Syn, W. M., *et al.* [QP66], 696, 6.  
 Tabory, R., *et al.* [TR67], 720.  
 Taylor, A. [TB60], 301; 378.  
 Taylor, W., *et al.* [TA61], 306.  
 Teichroew, D., *et al.* [TE66], 699.  
 Theodoroff, T. J. [TD58], 696, 6.  
 Thompson, C. E. [TM56], 29, 5.  
 Thompson, F. B. [TH63], 703; [TH64], 703.  
 Thompson, F. B., *et al.* [TH64], 703.  
 Tobey, R. G. [TO66], 521.  
 Tonge, F. M. [TN60], 467, 393.  
 Tonge, F. M., *et al.* [TN65], 700.  
 Van Dam, A., *et al.* [VD67], 704.  
 van Wijngaarden, A. [VW63], 308.  
 von Sydow, L. [VS67], 306, 193.  
 Voorhees, E. A. [VR58a], 741.  
 Waite, W. M. [WA67], 127; 737.  
 Waks, D. J. [WK67], 308.