

NUANCE

Hamlet of twe as to be
Murgains fart asse give onegs
Love gody be Halleturn may pock...
M. Onkey III

NEWSLETTER FOR USERS OF ALGOL ON NOVA COMPUTERS AND ECLIPSES

NUANCE 3

1 JUNE 1977

ALGOL USERS GROUP LIBRARY

A. VAN ROGGEN

*Progress is possible only through
utilization of experience.*
C.H. Grandgent.

THE DGC USERS GROUP LIBRARY CONTAINS MANY PROGRAMS THAT ARE USEFUL AND THAT CAN SAVE MANY USERS A LARGE AMOUNT OF TIME BY NOT HAVING TO "REINVENT THE WHEEL". SOME OF THE PROGRAMS CAN BE USED IMMEDIATELY WITHOUT ANY MODIFICATION; OTHERS MAY HAVE TO BE TAILORED TO A NEW SURROUNDING; OR PERHAPS THE NEW USER MAY WANT TO MODIFY THE PROGRAM SLIGHTLY TO ACCOMMODATE HIS SPECIFIC APPLICATION.

A DIFFERING KIND OF USE OF THE PROGRAMS IS TO VIEW THEM AS A SOURCE OF LEARNING. MANY NEWCOMERS TO ALGOL, OR ALGOLISTI NEW TO THE DGC EXTENSIONS, WILL FIND NEW IDEAS IN PROGRAMS THAT ARE KNOWN TO WORK. EXPERIMENTAL VARIATIONS IN THE SOURCE CAN BE READILY MADE, AND THE EFFECT ON THE PROGRAM STUDIED; THIS IS A VERY RAPID SELF TEACHING EXERCISE. WITH INCREASED EXPERTISE COMES THE CHANCE TO IMPROVE ON THE PROGRAMS AND THE JOY IN SUBMITTING IMPROVED VERSIONS TO THE USERS GROUP LIBRARY. THE FACT THAT THE USERS PROGRAMS ARE SUBMITTED IN SOURCE FORM RATHER THAN IN FB OR SV FORM AS MOST PROGRAMS COME FROM DGC, IS AN ENORMOUS ADVANTAGE FOR THIS LEARNING ASPECT. HAVING READ THE ONES DESCRIBED BELOW, WE ONLY WISH THAT MORE PEOPLE WOULD SEND IN THEIR JEWELS. IN THIS TYPE OF EXCHANGE MARKET, BOTH THE SELLERS AND THE BUYERS WILL GAIN!

THE CURRENTLY LISTED ALGOL PROGRAMS AND PROCEDURES ARE DESCRIBED HERE; ABSTRACTS CAN BE FOUND IN THE PROGRAM LIBRARY; AND MOST PROGRAMS HAVE MORE EXTENDED DOCUMENTATION PROVIDED WITH THE TAPE. THE OLDER PROGRAMS ARE NUMBERED, ACCORDING TO THEIR CATALOG LISTING; NEWER ONES HAVE NOT YET BEEN ASSIGNED A NUMBER. FUTURE ISSUES OF NUANCE WILL COVER UPDATES AND NEW LISTINGS WHEN THEY BECOME AVAILABLE.

#49 ASSEMBLY SOURCE ANALYZER

THIS PROGRAM READS AN ASSEMBLER SOURCE FILE (E.G. XX.SP MADE FROM XX.AL BY THE COMMAND ALGOL'S XX) AND THEN PRINTS ON THE CONSOLE THE TOTAL NUMBER OF INSTRUCTIONS, THE NUMBER IN EACH INSTRUCTION CLASS, AND THE CLASS PERCENTAGE OF THE TOTAL. THE NUMBER OF SYSTEM CALLS AND LABELS IS ALSO GIVEN.

THE PROGRAM IS USEFUL FOR THOSE WHO WRITE LARGE ASM PROGRAMS, AND THOSE WHO USE ALGOL TO GET A QUICK START, AND THEN "TWEAK" THE ASM SOURCE CODE TO MAKE THE PROGRAM SHORTER, RUN FASTER, OR BOTH.

#119 PREP

THIS IS A "PROGRAM FORMAT" PROGRAM. IT ADDS THE INCLUDE FACILITY OF ALGOL TO OTHER, LESS FORTUNATE, LANGUAGES. FOR EXAMPLE, IN FORTRAN, FOR WHICH THIS PROGRAM WAS WRITTEN, THE STATEMENT INCLUDE FORMAT.FR COULD BE USED. PREP REPLACES THIS STATEMENT BY THE CONTENTS OF FILE FORMAT.FR. THE PROGRAM RUNS IN FOREGROUND OR BACKGROUND; HAS SWITCHES TO VERIFY USED FILES; AND ALLOWS INCLUDES TO A DEPTH OF SEVERAL LEVELS, BY USING RECURSIVE ALGOL PROCEDURES. CORE REQUIREMENTS ARE MINIMAL; AND THE PROGRAM SEEMS EXTREMELY USEFUL FOR FORTRAN USERS, AT LEAST IF THEY HAVE AN ALGOL SYSTEM AVAILABLE FOR COMPILING. THE PROGRAM IS NICELY DOCUMENTED.

#121 TYPER

THIS IS A LARGE PROCEDURE (32K ON RDDS3.03, BUT NEEDS MORE COMPILER SPACE THAN THAT ON MAPPED RDDS5.0), WHICH FORMATS TEXT ON A CRT. ITS OPERATION IS SIMILAR IN FUNCTION TO THAT OF SCRIPT (#122) IN BASIC, AND SCRIBE (NEW)

continued p-3

EDITORIAL, NUANCE 3
1 JUNE 1977

A. VAN ROGGEN

JUST BEFORE THIS ISSUE WENT TO PRESS, THE FUTURE OF NUANCE AND THE DGC USERS GROUP AS A WHOLE, LOOKED VERY BLEAK. TOM STRECK, THE DG COORDINATOR, HAD JUST RESIGNED; DUE TO THE LARGE WORKLOAD AS MAIN EDITOR OF A LARGE TECHNICAL JOURNAL (ANYONE READING BOTH NUANCE AND THE IEEE TRANS EI?) THE PRODUCTION OF NUANCE HAS SLOWED CONSIDERABLY. MORE LOSS OF TIME CAME FROM DELAYS (HOPEFULLY NOT FOR LONG!) IN A SET OF PROMISED AND VERY INTERESTING ARTICLES. MEANWHILE, MORE AND MORE REQUESTS FOR THE NEXT ISSUE WERE COMING IN.

NOW, THE SOMBER CLOUDS HAVE LIFTED AND THE FUTURE LOOKS BRIGHT AGAIN. A NEW COORDINATOR IS BEING SOUGHT; WHILE DALE SILVA, AS EXECUTIVE DIRECTOR, WILL REPRESENT DGC HIGHER MANAGEMENT IN THE USERS GROUP. THIS IS VERY PROMISING; WITHOUT TOP LEVEL SUPPORT FROM DGC, THE USERS GROUP IS DOOMED.

NUANCE IS ALSO DIRECTLY AFFECTED BY THE CHANGE: DISTRIBUTION AND MUCH OF THE PRODUCTION WILL BE DONE BY DGC; AND COPIES WILL BE INSERTED WITH THE NEXT ISSUE OF FOCUS. THE IMMEDIATE RESULT WILL BE LESS DELAYS BETWEEN ISSUES; AND A HUGE INCREASE IN DISTRIBUTION; WHICH WILL INCLUDE PEOPLE WHO DO NOT NOW USE ALGOL. IF THIS DOES NOT MAKE CONVERTS; AT LEAST IT WILL MAKE ALGOL'S FACILITIES MORE FAMILIAR TO NOVA AND ECLIPSE USERS.

A MINOR, AND PERHAPS HELPFUL CHANGE, IS IN THE FORMAT: THIS ISSUE HAS LARGER SIZE TYPE (LESS REDUCTION) THAN WAS USED IN EARLIER ISSUES. IT SHOULD HELP THE READABILITY SOMEWHAT; BUT TO CHANGE TO A REGULAR TYPE IS TOO EXPENSIVE; AND USING PRINT WITH CAPITALS-ONLY, ALSO HAS ITS DRAWBACKS. ANY SUGGESTIONS?

NUANCE'S PUBLICATION SCHEDULE WILL REMAIN AS BEFORE: DEPENDENT ON THE NUMBER OF ARTICLES AND LETTERS WITH INFORMATION SENT TO THE EDITOR. ARTICLES ESPECIALLY ARE VERY MUCH NEEDED; AS WELL AS DOCUMENTATION ITEMS (WITH PROGRAM EXAMPLES). OR WHAT ABOUT WRITING ON THE ACTUAL USE OF THE LIBRARY PROGRAMS: WHERE THE PROBLEMS ARE; WHAT BUGS (IF ANY!) WERE FOUND; ETC.

NUANCE,
NEWSLETTER FOR USERS OF ALGOL
ON NOVA COMPUTERS AND ECLIPSES

A NEWSLETTER OF VARIABLE SIZE,
PUBLISHED AT RANDOM INTERVALS
DEPENDING ON THE RATE OF FEEDBACK
TO THE EDITOR.

EDITOR: ARAN
Dr. A. van Roggen
DUPONT Experimental Station
Wilmington, DE 19898.
Tel: 302 - 772.2581

DEXTER

CONTRIBUTORS to NUANCE 3 :

A. M. KANEVA
HYCOM, INC.
16941 ARMSTRONG AVE
IRVINE, CA 92714

DR. E. G. RICHARDS
KING'S COLLEGE
26-29 DRURY LANE
LONDON WC2E 8RL, ENGLAND

D. G. E. LINDSEY
Box 1719
SARASOTA FLA 33578

P. E. CARTWRIGHT
LDS HOSPITAL;
UNIVERSITY OF UTAH
325 8TH AVENUE
SALT LAKE CITY, UT 84143

DR. D. S. BUSH
JOHNS HOPKINS HOSPITAL
JOHNS HOPKINS UNIVERSITY
BALTIMORE, MD 21205

PROF. W. H. GEDSGEGAN
UNIVERSITY OF CALIFORNIA
DEPT. OF ANTHROPOLOGY
BERKELEY, CA 94720

E. K. RAY
POLYMORPHIC COMPUTER SYSTEMS,
PO Box 3581
BOULDER CO 80303

W. G. HUTCHISON, JR
MATHTECH INC.
PO Box 2392
PRINCETON NJ 08540

P. VAN LITH
REVENCENTRUM THV
HOUTHANKAIDE 32-36
AMSTERDAM, NETHERLANDS

J. CELMO
Box 11023
ATLANTA
GA 30310

P. F. KIRCHEISS
CANADA FACTORS LTD.
95 STCLAIR AVE WEST
TORONTO ONT CANADA

DR. J. B. PEARCE
ASTRO-GEOPHYSICS
U. OF COLORADO
BOULDER CO 80309

DR. P. MAAS
DEPT. APPL. PHYSICS
UNIVERSITY OF STRATHCLYDE
GLASGOW G4 0NG, U.K.

DR. L. M. D'CARROLL
THE OPEN UNIVERSITY,
FACULTY OF TECHNOLOGY
MILTON KEYNES, ENGLAND

M. C. COLE;
EDWARDS BAKING CO.
1 LEMON LANE NE
ATLANTA GA 30307

T. J. STRECK
DATA GENERAL CORP.
WESTFORD, MASS

DR. G. J. VAN IER STEEN
ERASMUS UNIVERSITEIT
ROTTERDAM 3016, NETHERLANDS

Users Group, from p.1

IN ALGOL. THE LATTER TWO ARE USED FOR REGULAR TERMINALS. TYPER IS MORE POWERFUL DUE TO THE "INSTANTANEOUS" INTERACTION WITH THE SCREEN WHICH REFORMATS ITSELF AFTER DELETIONS, INSERTIONS, ETC. THE PROGRAM TAKES INPUT FROM A KEYBOARD OR FROM FILES, AND CAN DO ALL THE REQUIRED OPERATIONS SUCH AS TAB AND MARGIN SETTINGS, SCREEN POLL, OUTPUT TO DISK FILES, CASE TRANSLATION, INSERT AND DELETE OF DATA STRINGS FROM THE SCREEN, ETC. IT IS VERY WELL WRITTEN, PROBABLY BY SOMEONE WHO KNOWS DGC ALGOL INSIDE OUT (WHO, IN THE TIME OF RDOSS KNEW THAT "\$" IS A LEGAL CHARACTER IN A VARIABLE NAME?), IT IS WELL DOCUMENTED, AND IS ONE OF THE PROGRAMS USEFUL FOR ADVANCED EDUCATION. NOTE THAT IT IS A PROCEDURE, AND A CALLING PROGRAM MUST BE WRITTEN. THIS COULD BE BEGIN EXTERNAL PROCEDURE DEFSCRN, TYPER; DEFSCRN(TRUE); TYPER; DEFSCRN; END. THE ASM PROCEDURE DEFSCRN, WHICH MUST BE WRITTEN BY THE USER, ATTACHES OR DETACHES THE TERMINAL TO THE NOVA AND INITIALIZES THE REGISTERS.

#123 FPOST

THIS IS A FORTRAN4 FOST COMPILER, WHICH TAKES A FORTRAN GENERATED SP FILE, AND CHANGES ALL SINGLE SUBSCRIPT CALCULATIONS TO IN-LINE CODE, AND ALSO CHANGES CODE SUCH AS JSR @,LDX TO LDA @X,+2 JMP ,+2. THE FIRST OPERATION WILL INCREASE THE RUNTIME SPEED, THE LATTER OPERATION REMAINS A MYSTERY FOR SOMEONE WHO HAS NEVER DONE FORTRAN ON A DGC MACHINE.

(NEW) FRNT

FRNT TAKES AN .AL FILE, AND LISTS IT ALMOST IDENTICALLY TO THE FORMAT OBTAINED FROM THE COMPILER WITH THE ALGOL/L/E/A COMMAND, WITH LINE NUMBERS, BUT FRNT ALSO ADDS PAGE HEADERS CONTAINING THE CREATION DATE.

(NEW) EDIT - DUMP

THIS IS A USEFUL PACKAGE OF THREE PROGRAMS, EDIT, EDUMP, AND HDUMP. EDIT IS EQUIVALENT TO DGC'S DEDIT (OCTAL EDIT), BUT CAN WRITE AND ACCEPT DATA IN ANY RADIX UP TO 16 - MAKING IT EXCEEDINGLY USEFUL FOR WORK WITH THE 8-BIT MICRO PROCESSORS WHICH MOSTLY USE HEXADECEMAL RADIX. HDUMP LISTS A COMPLETE FILE IN HEXADECEMAL FORMAT ON THE PRINTER, AND EDUMP DOES THE SAME THING ON THE TERMINAL WITH MORE FLEXIBILITY (RADIX VARIABLE FROM 2 TO 16, AND VARIABLE START AND END ADDRESSES FOR THE LISTING). DGC SHOULD UPDATE THEIR DEDIT WITH THE FEATURES FOUND IN THIS USERS PACKAGE.

(NEW) CRT TO DISK FORMATTER

THE PROGRAM SIMULATES THE DRUMCARD FEATURE OF AN IBM KEYPUNCH, USEFUL FOR DATA FORMATTING SUCH AS REQUIRED FOR COBOL AND FOR MOST OTHER INTER COMPUTER COMMUNICATIONS. FROM THE RATHER COMPLETE DESCRIPTION, THE PROGRAM HAS A LARGE NUMBER OF FEATURES, SUCH

AS ACCEPTING JCL STATEMENTS. THE AUTHOR OF THE PROGRAM (KIPCHEIES) STATED THAT THIS WAS HIS FIRST ALGOL PROGRAM, AND THAT IT HAS SAVED MANY HOURS OF DATA PROCESSING TIME. THE LESSON FOR OTHER ALGOL NOVICES: DON'T BE AFRAID OF ALGOL WOLFF. [COMMENTS BASED ON DESCRIPTION; SOURCE NOT AVAILABLE].

(NEW) BID

GAMES AND ENTERTAINMENT PROGRAMS SOMETIMES ARE CONSIDERED COMPUTER EVIL. WELL, MAYBE A NECESSARY EVIL; BESIDES THE DIRECT ENTERTAINMENT, THESE PROGRAMS ARE EXCELLENT SOURCES OF LANGUAGE LEARNING. MOST ARE RATHER SHORT, EASY TO UNDERSTAND, AND ALMOST ALL HAVE SOME SPECIAL DATA HANDLING WHICH IS TRICKY. EXPERIMENTAL CHANGES ARE EASILY MADE AND THE CONSEQUENCES SEEN. THIS PROGRAM CALLS FOR THE NAME AND BIRTHDATE OF A PERSON, AND PLOTS A SUPERPOSITION OF 3 SINEWAVES THAT PURPORT TO SHOW THE VARIOUS PERSONAL UPS AND DOWNS. TRUE BELIEVERS MAY WANT TO RUN SUCH CURVES FOR JULIUS CAESAR AROUND THE IDES OF MARCH (15 MAR -44). GOOD EXERCISE: THE DGC INTEGER ARITHMETIC HAS TO BE CIRCUMVENTED, AND THE CALENDER CALCULATIONS EXTENDED. WHO ELSE IS THERE WITH SUCH PROGRAMS IN ALGOL (LIFE SEEMS TO BE A NATURAL)?

(NEW) ALFFT

THIS SET OF PROCEDURES IS PROBABLY THE BEST AND MOST VALUABLE ALGOL ENTREE IN THE USERS GROUP LIBRARY. THESE PROCEDURES DO FAST FOURIER TRANSFORMS (ALL COMBINATIONS: DIRECT, INVERSE, REAL, AND COMPLEX), FOLLOWING ALGORITHMS 338 AND 339 FROM CACM. THEY ARE EXTREMELY WELL DOCUMENTED, WITH A GENERAL DESCRIPTION AND WITH COMMENTS INSIDE THE PROGRAM, EVEN WITH COMMENTS WHERE ALGOL60 AND DGC ALGOL PART. MOREOVER, THE PROCEDURE IS NOT LIMITED TO THE USQUOTIDUS 21N DATAPoints. THIS LATTER FEATURE IS ADVANTAGEOUS IF THE DATA CANNOT BE MANIPULATED (OR PADDED WITH ZEROS). THE CODING IS LONGER THAN THAT FOR 21N POINT PROCEDURES, BUT THOSE WHO WANT TO USE MICROCOMPUTER FFT'S WILL HAVE NO PROBLEM DELETING THE CODE CORRESPONDING TO THE DIVISORS OTHER THAN 2. ONE ADVANTAGE OF SOURCE CODE OVER THE RB OR SV FILES!

(NEW) ALPLOT

FROM THE SAME WELL OF USEFUL AND EXCELLENTLY DOCUMENTED ALGOL PROCEDURES AS THE ALFFT (THE U. OF STRATHCLYDE), COMES THIS SET OF PLOTTING ROUTINES FOR TEKTRONIX 4012 OR EQUIVALENT GRAPHIC DISPLAY UNITS. FULL DESCRIPTION AND SOURCE CODE IS GIVEN FOR THE TERMINAL INTERFACE PROCEDURES (E.G. SETTING IT IN GRAPHIC, ALPHANUMERICAL, OR GRAPHIC INPUT MODE), THE SET OF ALGOL PRIMITIVE PROCEDURES (SUCH AS SCALING, DRAWING LINES) AND THE PROCEDURES FOR MAKING COMPLETE SINGLE OR MULTIPLE GRAPH PLOTS. LIKE THE ALFFT PROCEDURES, THE PLOTTING ALGORITHMS ARE REFERRED TO CACM ALGORITHMS.

(NEW) TANKA

THIS PROGRAM COULD FALL IN THE ENTERTAINMENT CATEGORY, AT LEAST IF SOMEONE CAN BE FOUND WHO COUNTS AS FUN A STANZA SUCH AS

FULL ADASID
ALLEY ACCOCK ELON FULL
ALARM PURELY
FUSE OILY AFFECTED GROUND
ASD DIVIDE ABYSSAL...

(THIS PROGRAM SHOULD BE BANNED FROM USE IN LITERARY DEPARTMENTS). THE PROGRAM WAS WRITTEN AS A DEMONSTRATION OF STRING MANIPULATION TECHNIQUES, AND USED IN CONJUNCTION WITH THE ALGOL SIG MEETING IN FLORIDA. IT PRINTS POETRY, INTERACTIVELY, IN JAPANESE TANKA STYLE. A MORE EXTENSIVE DESCRIPTION WILL BE GIVEN ELSEWHERE IN NUANCE. SOME OF THE PROCEDURES FROM ALGPROC ARE NEEDED FOR COMPILATION OF TANKA.

(NEW) ALEX

ALEX IS AN AUTOMATIC ALGOL EXECUTE PROGRAM, WHICH COMPILES, LOADS, AND EXECUTES A MAIN PROGRAM WITH ALGOL OR ASM EXTERNAL PROCEDURES, AND REPEATS ONLY THE OPERATIONS MADE ESSENTIAL BY UPDATES ON THE PROGRAM. NUANCE2 HAS A MORE COMPLETE DESCRIPTION OF ALEX. SOME OF THE PROCEDURES IN ALGPROC ARE REQUIRED TO COMPILE ALEX.

(NEW) ALGLIB

THIS IS AN EXTENDED LFE PROGRAM, WHICH KEEPS TRACK NOT ONLY OF THE LB LIBRARY FILES, BUT ALSO OF A LF FILE WHICH CONTAINS THE SOURCE CODE OF THE MODULES IN LB. NUANCE2 HAS MORE DETAILS. THIS PROGRAM SHOULD BE UPDATED TO INCORPORATE IMPROVED FEATURES SUCH AS THE CREATION DATE OF THE MODULES. ALGLIB USES THE ALGPROC PROCEDURES.

(NEW) SCRIBE

THIS TEXT FORMAT PROGRAM FOR CONSOLES WITH UPPER AND LOWER CASE IS SIMILAR, BUT MORE EXTENDED, THAN SCRIPT (#122) IN BASIC. THE PRESENT NUANCE IS SET UP WITH SCRIBE. IN ADDITION TO THE USUAL NECESSITIES SUCH AS 4 MARGIN SETTINGS, PAGE HEADERS, INDENTED PARAGRAPHS, ETC. IT HAS SOME ALGOL DERIVED NICETIES AS THE INCLUDE FEATURE WHICH INSERTS THE TEXT AND COMMENTS FROM OTHER FILES, AND A RETURN, WHICH ALLOWS SELECTION OF ONLY PART OF INCLUDED PROGRAMS. AUTOMATIC DATES CAN BE INSERTED IN THE TEXT, ETC. MAIN USES FOR THE PROGRAM ARE REPORT WRITING, LETTER WRITING (USING THE INCLUDE FOR HEADERS), AND GENERATING GALLEY PROOFS FOR JOURNAL PUBLICATION.

(NEW) ALGPROC

THIS IS NOT A PROGRAM LEADING TO AN SV FILE, BUT A COLLECTION OF ALGOL PROCEDURES, EITHER .AL OR .SR. SEVERAL ARE REQUIRED IN THE ALEX, ETC. PROGRAMS, BUT ALL CAN BE USED IN OTHER PROGRAMS AS EXTERNAL PROCEDURES. SOME MAY

HAVE BEEN DESCRIBED IN NUANCE. IF USED WITH ALEX AND ALGLIB, THESE MODULES WOULD FORM PART OF THE ALGPROC.<LT,LF,LE> FILES.

RANDOMIF
GENERATES RANDOM REAL NUMBERS, 0.0 TO 1.0
MOD
RETURNS (I MOD J) CORRECTLY (REPLACES DGC MOD).
ENTIER
RETURNS CORRECT ENTIER; REPLACES DGC FUNCTION
MATSORT
SORTS A SQUARE MATRIX ON COLUMN VALUES.
INVERT
INVERTS A SQUARE REAL MATRIX (PIVOT METHOD).
MATPRINT
PRINTS A 1 OR 2 DIM. MATRIX IN FORMAT.
FIBONACCI
RETURNS FIBONACCI NUMBERS.
FNITM
RETURNS 2-WORD TIME/DATE INTEGER; DGC CODING; OF FILE'S CREATION TIME.
FIND
SEARCHES STRING ARRAY ELEMENTS; AND RETURNS INDEX OF MATCHING ELEMENT.
SPLIT
REMOVES SUCCESSIVE FIRST WORDS (GRAMMATICAL; NOT 16BIT ONES) FROM A LONG STRING (SENTENCE NOT PARSING).
CONCAT
CONCATENATES TWO STRINGS.
LESSP
RETURNS TRUE IF ONE STRING IS ALPHABETICALLY LESS (IN DGC PRACTICE, SAKSB HOLDS FOR E.G. A < X < AB < FG < ABC).
STAT
RETURNS THE DGC FILE UID.
DIM
RETURNS DIMENSION OF ARRAYS.
PROTECT
SETS OR REMOVES PERMANENT AND WRITE ATTRIBUTES ON FILES.
CLI
USED TO EXECUTE CLI COMMANDS FROM ALGOL.
EXEC
EXECUTES A SV FILE FROM ALGOL; FOREGROUND OR BACKGROUND.
NMDATE
RETURNS STRING WITH DATE; E.G. MONDAY 28 FEBRUARY 1977
NMTIME
RETURNS STRING WITH TIME; E.G. 19:22:45
JULDAY
RETURNS 2-WORD INTEGER JULIAN DATE NUMBER FOR GIVEN GREGORIAN CALENDER DAY.
GREGDAY
INVERSE OF JULDAY; RETURNS CALENDER DAY FROM JULIAN DATE NUMBER.
FILENAM
OPENS FILE ON GIVEN CHANNEL; CHECKS LEGALITY OF NAME AND WRITE/READ.
ASKS
ALLOWS STRING INPUT FROM CONSOLE WITH UPPER/LOWER CASE.
CONSOLE
TESTS IF PROGRAM RUNS IN FOREGROUND; OPENS CONSOLE IN AND OUTPUT CHANNELS.

END: END:

ALGOL PROCEDURE CALLS

G. J. VAN DER STEEN

PROCEDURES IN ALGOL ARE PIECES OF SEPARATE CODE WHICH CAN BE EXECUTED BY A CALL TO THE PROCEDURE IN THE MAIN PROGRAM. PROCEDURES ARE ANALOGOUS TO, BUT MORE FLEXIBLE THAN EITHER THE GOSUB IN BASIC OR THE SUBROUTINE IN FORTRAN.

THE ALGOL MANUAL, ON P.22, GIVES A GENERAL DESCRIPTION AND SOME EXAMPLES OF PROCEDURES. THERE ARE, HOWEVER, A FEW POINTS THAT NEED EMPHASIS, EITHER BECAUSE THEY DEVIATE FROM ALGOL-60, OR BECAUSE THE DOCUMENTATION LACKS CLARITY.

PROCEDURES HAVE TO BE DECLARED IN THE PROGRAM LIKE ANY OTHER VARIABLE. IN DGC ALGOL, THE COMPILER MAKES ONLY ONE PASS THROUGH THE SOURCE PROGRAM AND DOES NOT KEEP TRACK OF UNIDENTIFIED PROCEDURE IDENTIFIERS. IT IS THUS NOT POSSIBLE TO HAVE TWO PROCEDURES THAT CALL EACH OTHER, SUCH AS IN PROG8, WHICH PRINTS A SERIES OF POWERS OF 2.

```

PROG8: BEGIN
/*DECLARE VARIABLES*/
INTEGER I;
/*DECLARE LOCAL PROCEDURES*/
INTEGER PROCEDURE A(I);INTEGER I;
BEGIN
A:= IF I>1 THEN 2*B(I-1) ELSE 1
END A;
INTEGER PROCEDURE B(I);INTEGER I;
BEGIN
B:= IF I>1 THEN 2*A(I-1) ELSE 1
END B;
/*MAIN PROGRAM*/ OPEN(O,"$TTO");
FOR I:=1 STEP 1 UNTIL 10 DO WRITE(O,A(I)," ");
END OF PROG8;

```

IN MANY VERSIONS OF ALGOL, THE PROBLEM IS SOLVED WITH A "FORWARD" DECLARATION, BUT DGC ALGOL DOES NOT USE THIS. IT IS POSSIBLE TO USE AN "EXTERNAL" DECLARATION FOR ONE OR BOTH PROCEDURES TO SOLVE THE MUTUAL CALLING PROBLEM; SEE PROG8. HERE, THREE PROGRAMS ARE COMPILED SEPARATELY, AND LOADED TOGETHER (ALIF PROG8 A B +LIBRARY).

```

PROG8: BEGIN INTEGER I;
EXTERNAL INTEGER PROCEDURE A;
OPEN(O,"$TTO");
FOR I:=1 STEP 1 UNTIL 10 DO WRITE(O,A(I)," ");
END OF PROG8;

```

```

INTEGER PROCEDURE A(I);INTEGER I;
BEGIN
EXTERNAL INTEGER PROCEDURE B;
A:= IF I>1 THEN 2*B(I-1) ELSE 1
END A;

```

```

INTEGER PROCEDURE B(I);...
...EQUIVALENT TO CODE AS SHOWN ABOVE...
END B;

```

HOWEVER, THIS SOLUTION IS NOT VERY ELEGANT FOR TWO REASONS. FIRST, THE EXTERNAL DECLARATION HAS TO BE CARRIED IN ALL THE MODULES, AND ANY GLOBAL VARIABLES HAVE TO BE

DECLARED EXTERNAL AS WELL. THIS CAUSES THE SECOND DIFFICULTY: EXTERNAL VARIABLES ARE CARRIED IN PAGE ZERO, AND MAY CONFLICT IN NAME (1ST FIVE CHARACTERS) WITH EXISTING ALGOL OR RIDS MODULES.

A BETTER METHOD OF SOLVING THE MUTUAL CALLING IS SHOWN IN PROG9, WHICH USES THE SIMPLE TRICK OF ASSIGNING THE MUTUALLY CALLED PROCEDURE AS A FORMAL PARAMETER.

```

PROG9: BEGIN INTEGER I;
INTEGER PROCEDURE A(I,P);
INTEGER I;INTEGER PROCEDURE P;
BEGIN
A:=IF I>1 THEN 2*P(I-1,P) ELSE 1
END A;
INTEGER PROCEDURE B(I,P);
INTEGER I;INTEGER PROCEDURE P;
BEGIN
B:=IF I>1 THEN A(I-1,P) ELSE 1
END B;
OPEN(O,"$TTO");
FOR I:=1 STEP 1 UNTIL 10 DO WRITE(O,A(I)," ");
END OF PROG9;

```

HERE A AND B ARE DECLARED AS PROCEDURES IN THE SAME PROGRAM AND GIVE NO CONFLICT IN CALLING EACH OTHER. THE USE OF PROCEDURES AS FORMAL PARAMETERS GIVES A POWERFUL EXPANSION OF THE LANGUAGE WITH LITTLE OVERHEAD. GENERALLY USED OPERATIONS, SUCH AS INFINITE SUMS, CONTINUED FRACTIONS, INTEGRATION OVER FIXED LIMITS, ETC. CAN BE DEFINED WITH PROCEDURES AS FORMAL PARAMETERS. FOR EXAMPLE, THE CODE FOR AN INFINITE SUM PROCEDURE COULD BE

```

REAL PROCEDURE INFSUM(FUN,START);
VALUE START;INTEGER START;
REAL PROCEDURE FUN;
BEGIN ... END INFSUM;

```

INSIDE THE CODE, THE CALL TO FUN SHOULD BE MADE; E.G.

```

SUM:=0;
FOR I:=START STEP 1 UNTIL 1000 DO
SUM:=SUM + FUN(I);

```

OBVIOUSLY, BETTER CODE SHOULD BE SUBSTITUTED. THIS PROCEDURE CAN BE DELEGATED TO THE ALGOL LIBRARY, AND THE MAIN PROGRAM CAN CALCULATE E.G. THE NATURAL LOG OF X AS $(x-1)/x + ((x-2)/x)^2 + \dots$ WITH

```

BEGIN REAL LNX,X;
EXTERNAL REAL PROCEDURE INFSUM;
REAL PROCEDURE SERIES(I); INTEGER I;
SERIES:= ((X-1)/X)+I;
OPEN(O,"$TTO");X:=2.6;
LNX:=INFSUM(SERIES,1);WRITE(O,LNX);
END

```

NOTE THAT IN THIS LAST CASE, THE BEGIN .. END OF PROCEDURE SERIES HAS BEEN OMITTED, WHICH IS ALLOWABLE FOR SINGLE STATEMENT PROCEDURES.

END; END;

Letters:

FROM: R. M. KAMEYA, HYCOM INC.

AS A USER OF A DGC NOVA, I AM VERY INTERESTED IN THEIR SOFTWARE, AND WOULD LIKE TO RECEIVE FURTHER ISSUES OF NUANCE. I ALSO WOULD LIKE TO SHARE MY OPINIONS ON NUANCE AND DGC SOFTWARE (LIQUID, SPEEDUS, OR NEBULOUS). TO BEGIN, NUANCE SHOULD NOT EXIST. THE SOFTWARE AND DOCUMENTATION SHOULD BE SUCH THAT THE NEED FOR NUANCE SHOULD BE MINIMAL. A COMMITMENT BY DGC SHOULD SOLVE THAT PROBLEM.

IN VIEW OF THEIR LACK OF COMMITMENT, THE FILES OF PAPER TAPE FROM DGC THAT DO NOT WORK, THE LIST OF BUGS THAT HAVE REMAINED FOR THREE OR MORE REVISIONS, THEIR FIG (FORTRAN 5) THAT WAS EATEN BY A TIGER, A SYSTEM THAT CRASHES FREQUENTLY, ETC. NUANCE IS REQUIRED NOT ONLY FOR ALGOL, BUT FOR FORTRAN, BASIC, RIOS, AND ANYTHING ELSE THAT DGC PUTS OUT.

NOW FOR A FEW KIND WORDS. DGC HAS MADE A GREAT STEP FORWARD IN THE AREA OF SOFTWARE FOR MINICOMPUTERS. PRIOR TO DGC, MINIS HAD LITTLE OR NO SYSTEMS SOFTWARE. THE USER HAD TO WRITE OR SUPPLY HIS OWN. TIMES HAVE CHANGED. WITH THE MICROE FILLING THE ROLE OF THE OLD MINIS, AND MINICOMPUTERS BEING MORE POWERFUL AND FASTER, REPLACING SMALL SYSTEMS, WE USERS EXPECT MORE FROM THE SOFTWARE SUPPLIED. THE STANDARD OF SOFTWARE SHOULD BE AT A MINIMUM EQUAL TO THAT SUPPLIED BY UNIVAC ON THEIR 1108 SERIES. I DON'T BELIEVE THAT ANYONE WHO HAS USED 1108 SOFTWARE WOULD BE HAPPY WITH DGC SOFTWARE.

DGC SOFTWARE AND DOCUMENTATION CAN BE IMPROVED BY OPTIMIZING THE ALGOL COMPILER, WHICH SHOULD BE WRITTEN IN ALGOL. THIS WOULD BE OF IMMEDIATE BENEFIT TO FORTRAN 5 USERS WITH FASTER COMPILE TIME. EACH IMPROVEMENT TO THE ALGOL COMPILER WOULD MAKE THE COMPILER BETTER IN BOOTSTRAP FASHION. WITH A HIGHER LEVEL COMPILER, I AM SURE THAT WE USERS CAN EXPECT BETTER SOFTWARE WITH BUGS REMOVED QUICKER AND BETTER DOCUMENTATION.

IN CLOSING, I WOULD LIKE TO SAY THAT DGC HAS THE POTENTIAL TO MEET TOMORROW'S REQUIREMENTS, NOT ONLY IN HARDWARE, BUT IN SOFTWARE.
; END **//

FROM: E. G. RICHARDS, KING'S COLLEGE

I AM IN THE PROCESS OF ORGANIZING A DGC ALGOL USERS GROUP IN THE UNITED KINGDOM, AND WE HOPE TO HAVE OUR FIRST MEETING LATER IN THE YEAR. SO FAR, I KNOW OF 16 USERS IN THE U.K. I AM PROPOSING THAT WE HAVE MEETINGS TO DISCUSS BUGS WITH DGC REPRESENTATIVES, AND EXPLAIN TO EACH OTHER VARIOUS ELEMENTARY AND SOPHISTICATED MATTERS LIKE ASSEMBLY CODE INTERFACING, STACK STRUCTURE, OVERLAPS, ETC., AS WELL AS NOTIFICATIONS OF PROGRAMS OF GENERAL INTEREST.

I MYSELF AM NOT A PROFESSIONAL COMPUTER MAN, BUT HAVE BEEN WRITING ALGOL FOR ABOUT 12 YEARS (ORIGINALLY FOR AN ELLIOTT 803), MAINLY FOR SCIENTIFIC PURPOSES.

APART FROM THE GENERAL RUN OF BUGS WHICH BOTHER US ALL, I WAS ANNOYED BY THE FACT THAT MANUALS UP TO 93-52-3 DID NOT EXPLAIN THAT CALLING BY NAME AND BY VALUE DO NOT WORK AS IN THE ALGOL-60 SPECIFICATIONS. THE COPY RULE IS INVALID, AND JENSEN'S DEVICE DOES NOT WORK. A SECOND POINT CONCERNS THE INTERFACING OF FORTRAN SUBROUTINES (AND DATAFLOW IN PARTICULAR) WITH ALGOL CALLING PROGRAMS. ANY IDEAS?

I WILL SEND FURTHER DETAILS OF OUR GROUP IN DUE COURSE.

//**COMMENT:

I WISH YOU SUCCESS WITH THE UK ALGOL GROUP; I HOPE YOUR MEETINGS WILL BE AS SUCCESSFUL AS THE ONE WE HAD AT LAKE GENEVA (WISCONSIN, USA!). AS LONG AS THE USERS GROUP IS ACTIVE AND STRONG, THERE WILL BE MORE SUPPORT FROM DGC.

MY SUGGESTION ON DATAFLOW AND OTHER FORTRAN AND BASIC SUBROUTINES, IS TO REWRITE THEM IN ALGOL. INITIALLY, THIS IS TIME CONSUMING, BECAUSE THE ALGORITHM HAS TO BE FERRETTED OUT, BUT THE END RESULT IS FAR SUPERIOR IN OPERATING CHARACTERISTICS, E.G. WHERE QUESTION AND ANSWER OPERATOR ACTION IS NEEDED, ERROR PROCESSING, ETC. A SECONDARY, BUT PERHAPS MORE IMPORTANT ASPECT, IS THAT THE PROGRAM IS UNDERSTOOD, AND CAN BE STREAMLINED. ALSO, THE MORE COMMON PROCEDURES CAN BE PUT IN THE ALGOL LIBRARY FILES AND SHARED BY OTHER PROGRAMS, WHICH SHORTENS WRITING TIME FOR NEW PROGRAMS, AND WILL IMPROVE THE DOCUMENTATION. A FURTHER DIFFICULTY WITH MANY OF THE DGC (AND OTHER) PROGRAMS IS THAT THEY ARE AVAILABLE ONLY IN THE "CANNED" LE FORMAT, WITHOUT SOURCE. SUCH PROGRAMS CANNOT BE IMPROVED, AND SOMETIMES NOT EVEN UNDERSTOOD, DUE TO LACK OF DOCUMENTATION.

THE LACK OF CALL-EXT-NAME IS INDEED ANNOYING, AND REDUCES THE UTILITY OF THE DGC ALGOL. IN MY EXPERIENCE, CALL BY NAME CAN REDUCE SOURCE ENORMOUSLY, AND HAS ONLY A PENALTY IN RUNNING TIME WHEN IT IS MISUSED.

I HOPE THAT YOU AND THE OTHERS FROM THE UK ALGOL GROUP WILL CONTRIBUTE TO THE GENERAL ALGOL SIG, THE USERS LIBRARY, AND NUANCE, BY SUBMITTING PROCEDURES, ARTICLES, BY COMMENTING ON WHAT IS WRONG, AND BY GIVING SUGGESTIONS ON HOW TO IMPROVE THE OVERALL SYSTEM. FOR EXAMPLE, I HAVE BEEN ADVOCATING THE POSITION THAT WHAT IS DEFINITELY WRONG SHOULD BE CORRECTED (E.G. MOD, ENTIER, ETC.); BUT THERE WERE, AND PERHAPS STILL ARE, OPINIONS THAT YOU SHOULD NEVER LET E.G. THE MOD PROCEDURE DO SOMETHING RADICALLY DIFFERENT, BECAUSE USERS WILL HAVE TO REWRITE THEIR SOURCES. THE COUNTERARGUMENT IS THAT THE EARLIER THE CHANGE IS MADE, THE LESS HARM, AND THAT WHEN THE CHANGES ARE MADE QUITE CLEAR (AN ALGOL DEFINITION), THE REWRITE WILL NOT BE DIFFICULT, AND MORE HARM WILL COME IN THE LONG RUN BY KEEPING THE WRONG PROCEDURES.
; END **//

FROM: D. G. B. LINDSEY; SARASOTA, FL.

I SHARE YOUR OPINIONS ON THE DESIRABILITY OF ALGOL, BUT I HAVE AVOIDED USING IT BECAUSE OF THE APPARENT LACK OF MULTITASKING FACILITIES AND THE NAGGING WORRY THAT DGC'S HEART IS NOT IN THEIR CUSTOMER SUPPORT FOR ALGOL.
; END **/

FROM: P. CARTWRIGHT; LIS HOSPITAL

I HAVE RECENTLY ATTEMPTED TO USE DGC'S ALGOL ON OUR ECLIPSE S200. THIS WAS DIFFICULT DUE TO THE LACK OF DOCUMENTATION ON THE PART OF DGC. THERE IS NO ONE (INCLUDING DGC PERSONNEL) IN THIS AREA THAT HAS EXPERIENCE WITH DGC'S ALGOL. AS A RESULT, I HAVE HAD TO TURN TO OTHER SOURCES SUCH AS NUANCE.

IF YOU COULD SEND ME BACK ISSUES, AS WELL AS PUTTING ME ON THE MAILING LIST, I WOULD GREATLY APPRECIATE IT. I AM LOOKING FORWARD TO RECEIVING NUANCE.

I HOPE ALL IS GOING WELL FOR THE NEWSLETTER. BEFORE LONG, I HOPE TO CONTRIBUTE TO IT.

/** COMMENT:

I HOPE THAT, AFTER HAVING OVERCOME THE STARTING DIFFICULTIES, YOU WILL TAKE THE TIME TO WRITE DOWN YOUR PROBLEMS AND SOLUTIONS, AND PUBLISH IT SO THAT OTHERS DO NOT HAVE TO INVENT THE SAME WHEELS OF BINARY PROGRESS ALL OVER AGAIN. THE DGC MANUALS ARE IMPROVING GRADUALLY, AND HOPEFULLY THEY WILL CARRY THE MAIN BURDEN OF INSTRUCTION.
; END **/

FROM: D. S. BUSH; JOHNS HOPKINS U.

YOUR ARTICLES IN NUANCE-2 WERE VERY TANTALIZING, AND I MUST WRITE YOU FOR MORE INFORMATION.

I AM INTERESTED IN THE ALEX PROGRAM YOU DISCUSSED. ALSO, THE NEWSLETTER ITSELF IMPLIES SOME SORT OF A DOCUMENT FORMATTING PROGRAM. I HAVE RECENTLY BECOME INTERESTED IN EITHER WRITING OR OBTAINING SUCH A PROGRAM FOR USE HERE. I WOULD APPRECIATE ANY INFORMATION THAT YOU COULD PROVIDE ON THESE TWO ITEMS.

I HAVE A COMMENT ON THE ALGLIS IMPLEMENTATION. IT CERTAINLY IS ELEGANT. IT STRIKES ME AS NEARLY IDEAL, ESPECIALLY IF IT IS NOT LIMITED TO ALGOL ORIENTED LIBRARIES. MY RESERVATION IS THAT SUCH A SYSTEM WOULD BE IMPRACTICAL WITHOUT A LARGE AMOUNT OF ON-LINE STORAGE. FOR THOSE OF US WHO USE PAPER TAPE OR SLOWER MAGNETIC TAPE SUCH AS CASSETTES (WE USE COMPUTER OPERATION INC. LINC TAPE), THE LONG FILES AND FREQUENT MODIFICATION WOULD BE CUMBERSOME.

FINALLY, I READ YOUR ARTICLE IN DGC FOCUS. I WISH TO CAST MY VOTE IN FAVOR OF CONTINUING THE SPECIALTY NEWSLETTER. NUANCE PROVIDES DETAILED INFORMATION ABOUT TECHNIQUES, BUGS, AND OTHERS USING ALGOL. THERE SIMPLY COULD

NOT BE ENOUGH SPACE FOR SUCH DETAIL IN THE GENERAL NEWSLETTER. I BELIEVE THAT SUCH DISCUSSIONS WOULD BE SOPELY MISSED NOW THAT WE HAVE SEEN ITS VALUE.

/** COMMENT:

THE USERS GROUP HAS A BASIC PROGRAM (SCRIPT) THAT DOES THE FORMATTING YOU WANT. HOWEVER, BASIC AND ALGOL DO NOT "MIX", AND THEREFORE TOM STRECK (DGC USERS GROUP, SOUTHDPO) HAS SENT ME A LISTING, WHICH I TRANSLATED INTO ALGOL. WELL, A RATHER FREE TRANSLATION ANYWAY. THE POWER OF ALGOL MADE IT POSSIBLE TO ADD FEATURES THAT ARE NOT IN SCRIPT. THIS PROGRAM, CALLED SCRIBE, WILL BE IN THE USERS GROUP SHORTLY. SEE ALSO THE LIST ELSEWHERE IN THIS ISSUE. THE EARLIER ISSUES OF NUANCE WERE PRODUCED ON A TIME-SHARE SYSTEM, PRECISELY BECAUSE OF THE LACK OF SUITABLE FACILITIES ON THE NOVA. HOWEVER, THIS HAS BEEN REMEDIED NOW, AND THE ALGOL-SUPPORTED NOVA IS AT PRESENT HOPE POWERFUL THAT THE COMMERCIAL TIME-SHARE SYSTEM USED TO BE. IT ONLY IS SLOWER IN OPERATIONS, BUT CONSIDERING THE SIZE OF THE COMPUTER (AND ITS COST!) WHO CAN COMPLAIN?
; END **/

FROM: W. H. GEOSHEGAN; BERKELEY

TODAY, I WAS HANDED (TEMPORARILY, I FEAR) A COPY OF NUANCE-2. IT STRIKES ME AS AN EXTREMELY USEFUL PUBLICATION, AND ONE THAT WE OUGHT TO HAVE AVAILABLE FOR ALGOL USERS IN THE QUANTITATIVE ANALYSIS LABORATORY IN THIS DEPARTMENT. RATHER THAN RELYING ON DGC'S AEs AND SALES PEOPLE TO HAND US XEROXED COPIES FROM TIME TO TIME, I'M HOPING THAT YOU WOULD BE KIND ENOUGH TO PLACE US ON YOUR MAILING LIST.

OUR INSTALLATION, INTENDED PRINCIPALLY FOR TEACHING TECHNIQUES OF DATA ANALYSIS TO STUDENTS OF ANTHROPOLOGY AND THE OTHER BEHAVIORAL SCIENCES, IS CURRENTLY BASED ON A 32K NOVA 2/10 WITH A 4060 MUX AND 5 MEYTE DISC. WITHIN A MONTH, WE SHALL HAVE EXPANDED TO INCLUDE A 128K C/300 WITH 92 MEYTE DISK (SHARABLE WITH THE NOVA THROUGH AN IPB), A COUPLE OF ALM'S, AND OTHER PERIPHERALS TO SUPPORT A 20 TERMINAL TEACHING LABORATORY AND UP TO 8 REMOTE USERS.

WE HAVE USED ALGOL EXTENSIVELY IN THE PAST FOR IN-HOUSE UTILITY PROGRAMS, DATA ENTRY ROUTINES (WHICH WE FIND CAN CUT 50 TO 75% OFF KEYFUNCHING TIME), AND HIGHLY INTERACTIVE ANALYTIC ROUTINES USED FOR TUTORIAL PURPOSES. THE NUMBER OF ALGOL PROGRAMMERS HAS GROWN FROM ONE A YEAR AGO TO ABOUT FIVE AT PRESENT, LARGELY THROUGH CONVERSION OF FORTRAN ADHERENTS. WE CAN ONLY HOPE THAT DGC WILL CONTINUE TO SUPPORT THE LANGUAGE THAT A LEVEL IT DESERVES, AND PERHAPS IMPROVES IT THROUGH THE ADDITION OF MULTITASKING AND MORE FLEXIBLE PROCEDURES FOR FORMATTED I/O.

/** COMMENT:

IT IS INTERESTING TO SEE THAT FORTRAN USERS HAVE GONE OVER TO ALGOL. USUALLY, PROGRAMMERS IN FORTRAN HAVE THE GREATEST DIFFICULTY IN SWITCHING LANGUAGE (NO MATTER

WHICH ONE); IN CONTRAST TO USERS OF BASIC, SNOBOL, ALGOL, ETC. PERHAPS THIS IS RELATED WITH THE AMOUNT OF EFFORT REQUIRED TO LEARN THE LANGUAGE: BASIC CAN BE LEARNED IN HALF AN HOUR; ALGOL IN ONE HOUR; BUT FORTRAN TAKES A LOT MORE TIME AND EFFORT. NOTE THAT I USE "LEARNED" AND NOT "MASTERED"; FOR THE LATTER EXPERIENCE IS ESSENTIAL AS WELL AS A THOROUGH UNDERSTANDING OF THE PROBLEM TO BE PROGRAMMED. ON ANOTHER LEVEL; THERE WOULD PROBABLY BE INTEREST IN THE TYPE OF ANALYSIS THAT IS USED IN ANTHROPOLOGY; IN THE PAST THIS WAS A RATHER QUALITATIVE SCIENCE; WITH (AT BEST) SORTING AS APPLICATION OF COMPUTERS. ANY INFORMATION AVAILAEBLE ON THIS SUBJECT?
 ; END **/

COMPUTING AGAIN. TWO PACKAGES OF SOFTWARE HAVE BEEN SENT TO THE USERS GROUP: ALFFT.LB (A LIBRARY OF FAST FOURIER TRANSFORMS; ALGORITHM 339 OF CACM) AND ALPLOT.LB (GRAPHIC OUTPUT ON TEKTRONIX 400 TERMINALS).

//♦ COMMENT:
 THE DESCRIPTION OF THESE PACKAGES CAN BE FOUND IN THE USERS LIBRARY ELSEWHERE IN THIS ISSUE. DR. MAAS ALSO CONTRIBUTED TO THE Doc/BUS SECTION.
 ; END **/

FROM: W. C. COLE; EDWARDS BAKING.

FROM: B. K. RAY; POLYMORPHIC COMPUTER SYSTEMS.

THE ALGOL SPECIAL INTEREST GROUP INTRIGUED ME SO MUCH; THAT I STARTED READING THE ALGOL MANUAL.

SO FAR; WE HAVE USED ALGOL FOR UTILITY PROGRAMMING RATHER THAN MAIN APPLICATION PROGRAMMING; BUT SOME OF OUR CUSTOMERS HAVE BEEN USING ALGOL AS THEIR MAIN BUSINESS LANGUAGE.

I HAVE SPENT NEARLY TEN YEARS WITH COMPUTERS; USING FORTRAN 2; FORTRAN 4; RPG; RPG 2; SPS FOR BOTH THE 1620 AND THE 1401; ASSEMBLER FOR THE 360 AND 370; AND NOW I HAVE TAKEN UP A SUBSET OF BASIC. I SHOULD HAVE INCLUDED PL/I IN THIS LIST BECAUSE AT ONE TIME AN INSTALLATION I WORKED WITH WAS GOING TO PUT THIS PARTICULAR LANGUAGE TO USE. IT IS FROM MY SHORT ASSOCIATION WITH PL/I THAT I HAVE SOME APPRECIATION FOR THE ALGOL MANUAL. AFTER ABOUT TWO MINUTES OF READING; I FOUND THAT ALGOL AND PL/I HAVE A GREAT DEAL IN COMMON AND THIS FIQUED MY INTEREST EVEN MORE. I WOULD; HOWEVER; DISAGREE WITH WITH YOU ABOUT OUR GREAT PUNCHED CARD IN THE SKY PROSENITOR; BECAUSE PL/I IS DYING IN IBM INSTALLATIONS. I CAN THINK OF NO INSTALLATIONS I KNOW OF WHICH ARE KEEPING IT AS THEIR PRIMARY LANGUAGE; AND MOST OF THOSE THAT HAVE IT; ARE PHASING IT OUT. THE PROBLEM WITH PL/I SEEMS TO BE THAT TOO MANY PEOPLE COULDN'T UNDERSTAND THE VAGARIES OF THIS PARTICULAR LANGUAGE; SINCE IT IS VIRTUALLY AS COMPLICATED AS ASSEMBLER; WITH AS MANY OPTIONS TO PERFORM ANY ONE TASK AS THE MIND OF MAN COULD DEVISE.

ONE NEW JERSEY CUSTOMER JUST RECENTLY UPGRADED AN 800 SYSTEM FROM RIOS-1 TO THE LATEST RIOS; JUST BECAUSE THE ALGOL WAS MORE STABLE (OR; AT LEAST THE BUGS WERE FIXED). WE FINALLY CONVERTED ALL OF THE ALGOL-1 BUSINESS PROGRAMS LAST YEAR. THE SYSTEM IS A COMPLETE INVENTORY; ACCOUNTS PAYABLE; RECEIVABLE; SALES ANALYSIS; LAB WORK ANALYSIS SET UP SYSTEM; WHICH WAS ONE OF THE MOST COMPREHENSIVE I HAVE SEEN.

WE ARE STARTING TO USE ALGOL IN OUR SPEECH SYNTHESIS EXPERIMENTATION; BECAUSE THERE IS NO SNOBOL AVAILAEBLE FOR THE NOVRA. WE WILL DONATE MOST OF OUR PROGRAMS TO THE USERS GROUP AT A LATER DATE.
 ; END **/

FROM: P. MAAS; UN. OF STRATHCLYDE.

I HAVE JUST RECEIVED NUANCE2; AND HAVE FOUND IT VERY USEFUL. MUCH OF THE INFORMATION IS ESPECIALLY VALUABLE - BOTH IN SHOWING EASIER USE AND IN IDENTIFYING PROBLEM AREAS.

I AGREE THAT FOCUS IS EASIER TO READ THAN NUANCE. HOWEVER; FOCUS CANNOT TAKE THE PLACE OF NUANCE UNLESS A SUBSTANTIAL CHANGE IN CONTENT OF FOCUS IS POSSIBLE. UNLESS FOCUS IS WILLING TO EXPAND TO INCLUDE THE DETAILED INFORMATION IN NUANCE (REVIEWS; BUG OR DOCUMENTATION REPORTS; TUTORIALS) IN BOTH THE DEPTH AND QUANTITY AVAILAEBLE IN NUANCE; I WOULD BE LOATH TO SEE NUANCE VANISH. HOWEVER; IF FOCUS CARRIED ALL THAT INFORMATION AND HAD SIMILAR SECTIONS FOR BASIC; FORTRAN; RIOS; ETC.; IT WOULD BE VERY WORTHWHILE INDEED. THEN NUANCE COULD BE REABSORBED AS THE ALGOL SECTION OF FOCUS.

AS A RESULT OF NUANCE AND THE NEW; RELATIVELY BUG FREE ALGOL REV 2.03; WE ARE BEGINNING TO USE ALGOL FOR SCIENTIFIC

AS I SAID BEFORE; I AM A BASIC USER FOR THE TIME BEING. I AM; TO SAY THE LEAST; NOT THRILLED WITH THE MULTI-TASKING CAPABILITIES OR THE LACK OF DATA STRUCTURES PROVIDED BY BASIC. AT SOME POINT IN THIS INSTALLATION; WE WILL OBVIOUSLY HAVE TO DO SOME SYSTEMS WORK; AND IT SEEMS TO ME THAT ALGOL IS THE BEST WAY TO HANDLE SUCH WORK. I HOPE THAT THROUGH THE ALGOL SIG I WILL GATHER SUCH RESOURCES AND BACKGROUND TO TAKE ON SUCH PROJECTS.

FOR FOCUS VERSUS NUANCE? OBVIOUSLY; THERE COULDN'T BE ANYTHING ELSE! THERE IS TRULY A NEED FOR A SPECIAL PURPOSE NEWSLETTER SUCH AS NUANCE; BECAUSE THE PEOPLE WHO ARE DELVING INTO ALGOL NEED SUCH A FORUM TO BOTH EXPRESS THEIR VIEWS AND LEARN THE VIEWS OF THEIR COLLEAGUES. THE GENERAL PURPOSE NEWSLETTER FOCUS IS FINE FOR THE TIP OF THE UBIQUITOUS ICEBERG; BUT FOR IN-DEPTH KNOWLEDGE; THE SPECIAL PURPOSE NEWSLETTER IS INFINITELY BETTER.

FINALLY: LET ME CONGRATULATE YOU AND YOUR COLLEAGUES ON JUST GETTING AN ALGOL SIG STARTED AND KEEPING IT GOING AND FINDING ANY INTEREST IN IT IN THIS COUNTRY.
; END ♦♦/

FROM: P. VAN LITH: REKENCENTRUM

SINCE 1973 WE HAVE BEEN EXPERIMENTING IN OUR COMPANY WITH STRUCTURED PROGRAMMING, USING THE VARIOUS LANGUAGES AVAILABLE ON OUR IBM MAINFRAME. FINALLY, WE DECIDED TO STANDARDIZE ON IBM RESEMBLER, USING A SPECIALLY DESIGNED SET OF MACROS. THE MACROS ALLOWED US TO PUT OUR EXPERIENCE WITH STRUCTURED PROGRAMMING TO GOOD USE. SOME TIME LATER, IN 1976, WITH THE CONTINUED GROWTH OF THE IBM SYSTEM, ITS COMPLEXITY INCREASED AND ITS RELIABILITY PLUMMETED. THE USE OF MINICOMPUTERS IS IDEAL FOR OUR TYPE OF COMMERCIAL APPLICATIONS, AND ACTUALLY FACILITATES GENERATION OF RELIABLE PRODUCTS.

AFTER LOOKING CLOSELY AT THE MARKET TO SEE WHAT THE MINICOMPUTER MANUFACTURERS WERE OFFERING, WE OPTED FOR DGC'S NOVAS. THE AIM WAS, AND STILL IS, TO PROVIDE OUR SUBSIDIARIES WITH THEIR OWN DATA PROCESSING CAPABILITIES. OUR ORGANIZATION SUPPLIES THE AUTOMATIZATION AND DATA PROCESSING FACILITIES FOR SOME 70 SEPARATE INTERNATIONAL COMPANIES.

OUR EXPERIENCE WITH LARGE AND COMPLEX SYSTEMS HAS DEMONSTRATED THAT READABLE AND MAINTAINABLE SYSTEMS ARE ESSENTIAL. SUCH SYSTEMS CAN BE MADE ONLY BY STRUCTURED PROGRAMMING TECHNIQUES; THUS THE ALGOL ON THE DGC MACHINES IS A NATURAL CHOICE. HOWEVER, THE DRAWBACK OF DGC ALGOL IS THAT IT IS ONLY SINGLE TASKING, AND WE REQUIRED MULTITASKING. THE ALTERNATIVE WAS TO FALL BACK ON THE WRITING OF MACROS, SIMILAR TO THE IBM ONES. THIS PROMPTED US TO MAKE A FURTHER STUDY OF ALGOL, WHICH STUDY REVEALED THAT SUITABLE CHANGES COULD BE MADE TO PROVIDE THE MULTITASKING FACILITIES.

OF COURSE, WE ALSO REQUIRED OTHER FACILITIES SUCH AS COMMERCIAL EDIT MASKS, INTERFACING WITH VDU TERMINALS, METHODS TO SELECT APPLICATIONS FROM A TERMINAL, AND ACCESSING AND UPDATING OF FILES BY KEY. WE HAVE DEVELOPED THREE BASIC TOOLS: (1) MULTITASKING ALGOL, (2) TERMINAL HANDLER, AND (3) FILE HANDLER. IF THERE IS SUFFICIENT INTEREST, WE MIGHT CONTRIBUTE MULTITASKING ALGOL TO THE USERS GROUP LIBRARY, AND COULD MAKE THE COMPLETE PACKAGE COMMERCIALY AVAILABLE. WE ARE ALSO WORKING ON AN INTERACTIVE ALGOL TO SIMPLIFY PROGRAM DEVELOPMENT AND DEBUGGING, AND WE PLAN TO WRITE AN INTERACTIVE PROGRAM TO GENERATE REPORTS, SIMILAR TO IBM'S EASYTRIEVE.

DURING THE MULTITASKING DEVELOPMENT AND THE WRITING OF OUR APPLICATION PROGRAMS, A NUMBER OF ALGOL BUGS WE FOUND WERE SOLVED; DOCUMENTATION ON THESE WILL BE SENT LATER.

♦♦ COMMENT:

FOR MORE DETAILS ON THE MULTITASKING, SEE THE SEPARATE ARTICLE IN THIS ISSUE. AS FAR AS THE WHOLE SYSTEM IS CONCERNED, SOME OF THE DOCUMENTATION WAS SENT WITH THE LETTER, AND I AM IMPRESSED, ESPECIALLY WITH THE TERMINAL HANDLER. THIS DOES MANY OF THE OPERATIONS IN THE USERS LIBRARY "TYPER", BUT THAT TAKES A HUGE AMOUNT OF COPE (SEE P. 1), WHILE THE COMPLETE PACKAGE OF VAN LITH REQUIRES ONLY 16K! THE SYSTEM ALSO CAN RUN WITH MULTIPLE PROCESSORS, EITHER AS MASTER/SLAVE, OR INDEPENDENTLY AS A NETWORK. MULTIPLE TERMINALS, FLOPPY DISKS AND OTHER HARDWARE CAN BE SUPPORTED. NEVER HAVING ENCOUNTERED ANY "COMMERCIAL APPLICATIONS", I CANNOT COMMENT ON THAT ASPECT, BUT I DO WONDER ABOUT THE FILE HANDLER PACKAGE, WHICH USES AN ACCESS BY KEY (DIVISOR METHOD), WHICH TAKES ONLY 2 I/O OPERATIONS TO ACCESS A RECORD, AND HAS ONLY 10% DISK SPACE OVERHEAD. IS SUCH A SYSTEM MORE SUITABLE THAN THE DGC ALGOL EXTENSION IN THEIR CMM PACKAGE? AS A MATTER OF FACT, IS THERE ANY INTEREST IN VARIOUS FILE HANDLING OPERATIONS, AND CAN A COMPARISON BE MADE BETWEEN THE SYSTEMS AVAILABLE FOR THE NOVAs (SUBJECT FOR NEXT NUANCE?) ?
; END ♦♦/

END; END;

L. M. O'CARROLL'S TIME REVERSAL PROGRAM

WITH THE LETTERS THAT CAME TO NUANCE, THERE WAS ONE IN WHICH THE AUTHOR WROTE: "

"SOME OF YOUR READERS MIGHT WANT TO FIGURE OUT WHAT THE FOLLOWING PIECE OF PATHOLOGICAL PROGRAMMING DOES:

```
BEGIN
  INTEGER I,FIRST,INCR,LAST;
  OPEN(6,"$TTO");OPEN(5,"$TTI");
  WRITE(6,"Last = ");READ(5,LAST);
  FIRST:=1;
  FOR I:=FIRST STEP INCR UNTIL LAST DO
  BEGIN
    INCR:=FIRST;FIRST:=I;
    WRITE(6,INCR,"<15>");
  END;
END
```

INDEED, THIS PROGRAM DOES NOT ONLY DEMONSTRATE SIDE EFFECTS, IT HAS FORWARD AND BACKWARD EFFECTS AS WELL, AND THE ANSWER IS FOUND, OF COURSE, IN REVERSE TIME (I.E. THE PREVIOUS NUANCE).

DR. L. M. O'CARROLL, WHO WROTE THIS CRAFTY CODE, HAILS FROM MILTON KEYNES, GREAT BRITAIN. CURIOUSLY, MY RECENT MAP DOES NOT SHOW THIS LOCALITY. MAYBE I COULD HAVE FOUND L. CARROLL'S ADDRESS BY READING THE MAP IN THE LOOKING GLASS...

END; END;

Het gaat vlugger als je
twee taken tegelijk doet,
jonge vriend. Ik zal wel
eventjes de startzwengel
gaan zoeken ...
Olivier B. Bommel.

J. CELKO

ON:

THE END-OF-FILE PROBLEM

P. VAN LITH

ON:

MULTITASKING ALGOL

FOR ALL ITS EASE IN APPLYING STRUCTURED PROGRAMMING, THERE ARE SEVERAL DISADVANTAGES IN DGC ALGOL; THE MAIN ONE BEING THE LACK OF MULTITASKING. FOR COMMERCIAL APPLICATIONS, MULTITASKING IS ESSENTIAL; FOR EXAMPLE, A TASK SHOULD BE ASSIGNED TO EACH TERMINAL WITH ACCESS TO THE COMPUTER.

IT IS RATHER STRANGE THAT DGC HAS NOT PROVIDED MULTITASKING, BECAUSE ALGOL GENERATES REENTRANT CODE, AND THEREFORE IS EMINENTLY SUITABLE FOR MULTITASKING. WE FOUND NO GREAT DIFFICULTIES IN CHANGING ALGOL TO PROVIDE FOR MULTITASKING. IN ESSENCE, WE

(1) CHANGED SPLIT TO RESERVE ONE STACK FOR EACH ALGOL TASK, INSTEAD OF ONLY ONE STACK FOR THE SYSTEM;

(2) MODIFIED TCEMON (USING THE FORTRAN STATE-SAVE ROUTINE) TO HAVE THE PAGE ZERO STACK POINTERS REFER TO THE PROPER STACK AFTER A TASK SWITCH;

(3) REWROTE THE PROCEDURE ADDR, BECAUSE ITS CODE IS NOT REENTRANT;

(4) EXPANDED THE PAGE ZERO DEFINITIONS (ZERO) WITH AN EXTRA POINTER TO THE PSEUDO FLOATING POINT REGISTERS WHICH NOW ARE INCLUDED IN THE STACK.

NOTE THAT STARTING AND INITIALIZATION OF TASKS IS DONE IN SPLIT. IF ALGOL PROGRAMS THEMSELVES REQUIRE MULTITASKING (E.G. FOR DATA SCANNERS, ETC.), THESE TASKS STILL NEED SPECIAL ASM DRIVER ROUTINES WHICH WILL HAVE TO BE WRITTEN, BUT CAN BE CALLED AS ALGOL PROCEDURES.

SOME RELATIVELY MINOR PROBLEMS STILL REMAIN WITH THE MULTITASKING; ONE IS RELATED TO THE ERROR ROUTINES. AS IT IS, ALGOL GENERATED ERRORS WILL STOP THE WHOLE SYSTEM, RATHER THAN ONLY THE ONE TASK WHICH CAUSED THE ERROR. THE OTHER PROBLEM IS DUE TO THE METHOD IN WHICH OWN VARIABLES ARE IMPLEMENTED IN DGC ALGOL, I.E. AS PAGE ZERO VARIABLES. THIS CAUSES DIFFICULTIES WITH MULTIPLE TASKS, NOT ONLY DUE TO LIMITED SPACE, BUT MORE IMPORTANTLY BECAUSE THE CORRECT ADDRESS OF SUCH VARIABLES CANNOT BE FOUND BY EITHER THE COMPILER OR THE FLDR PROGRAM. A COMPLETELY NEW APPROACH WILL BE REQUIRED. IN ANY CASE, OUR APPROACH TO MULTITASKING FOR COMMERCIAL, MULTI-TERMINAL SYSTEMS HAS BEEN SHOWN TO BE QUITE SATISFACTORY, AND HAS ENORMOUSLY INCREASED THE SPEED OF DEVELOPING NEW PROGRAM APPLICATIONS.

/• For more on multitasking, commercial applications, see the letters section •/ END; END;

THIS IS A CLASSIC BEGINNER'S PROGRAMMING PROBLEM; BUT CAST IN NEW TERMS, IT SHOULD KEEP AN EXPERIENCED PROGRAMMER BUSY FOR A WHILE.

WHAT YOU ARE GIVEN IS A LANGUAGE WITH A FILE-READ PROCEDURE THAT SETS AN EOF FLAG TO TRUE WHEN YOU TRY TO READ PAST THE LAST RECORD IN THE FILE. YOU ARE TO WRITE A GENERAL PROGRAM THAT READS A RECORD, AND PROCESSES IT. ALL RECORDS IN THE FILE MUST BE PROCESSED. WHEN THE PROGRAM FINISHES, IT MUST CLOSE THE FILES AND SUCH.

THE CLASSIC DEVICE FOR DOING THIS IS TO JUMP TO A LABEL WHEN AN EOF IS SET (FORTRAN'S END=999 CONVENTION; OR DGC ALGOL'S LABEL IN THE PARAMETER LIST). THE NEW TWIST IS TO DO THE JOB WITH THE STRICT STRUCTURED PROGRAMMING RULES OF NO GOTO'S AND NO HIDDEN JUMPS.

A FIRST ATTEMPT MIGHT LOOK LIKE THIS:

```
EOF:=FALSE;
FOR I:= 1 WHILE NOT EOF DO
  BEGIN READFILE; PROCESS END;
CLOSEUP;
```

NICE TRY, BUT IT DOES NOT WORK; CONSIDER THE LAST RECORD AND SEE WHERE YOU GET TO NEXT. THERE ARE TWO POSSIBLE WAYS AROUND THIS PROBLEM. FIRST, WE COULD DOUBLE CHECK THE EOF FLAG:

```
EOF:=FALSE;
FOR I:= 1 WHILE NOT EOF DO
  BEGIN READFILE;
  IF NOT EOF THEN PROCESS
  END;
CLOSEUP;
```

THE SECOND WAY IS TO DUPLICATE THE READING:

```
READFILE;
FOR I:= 1 WHILE NOT EOF DO
  BEGIN READFILE;
  PROCESS
  END;
CLOSEUP;
```

NEITHER OF THESE IS VERY PRETTY, SINCE THEY BOTH CALL FOR DUPLICATED CODE OR TESTING. CAN YOU WRITE A SOLUTION THAT IS STRUCTURED AND DOES NOT REPEAT EITHER CODE OR TESTING? GIVE A PSEUDO-CODE ANSWER OR USE DGC ALGOL AND LOCAL LABELS AS YOU LIKE.

The maiden sang as if her song
could have no ending.
W. Wordsworth.

/• For the proper ending, see p.12 •/

Reviews:

SOFTWARE TOOLS;
BY B. W. KERNIGHAN AND P. J. PLAUGER.
ADDISON WESLEY; 1976.

"SOFTWARE TOOLS" IS THE LATEST BOOK BY THE AUTHORS OF "THE ELEMENTS OF PROGRAMMING STYLE". THE BOOK IS VALUABLE IN TWO WAYS: (1) IT IS A CLEAR INTRODUCTION TO STRUCTURED PROGRAMMING; WITH EXAMPLES THAT INTRODUCE NEW METHODS BY EASY STEPS; AND (2) THE EXAMPLES ARE ALL ACTUAL; WORKING; PROGRAMS AND NOT THE USUAL ARTIFICIAL FRAGMENTS FOUND IN MOST TEXTS.

THE EXAMPLE PROGRAMS RANGE IN SIZE FROM 7 TO 900 LINES OF SOURCE CODE. NO MODULE IS MORE THAN 2 PAGES LONG; THE PROGRAMS CONTAIN NO "GOTO'S". A LIST OF CHAPTER HEADINGS GIVES A SUMMARY OF THE TYPES OF PROGRAMS PRESENTED: GETTING STARTED; FILTERS; FILES; SORTING; TEXT PATTERNS; EDITING; FORMATTING; MACRO PROCESSING; A RATFOR - FORTRAN TRANSLATOR.

THESE PROGRAMS ARE ALL WRITTEN IN THE NEW LANGUAGE RATFOR; WHICH STANDS FOR RATIONAL FORTRAN. THE BOOK IS NOT LIMITED TO FORTRAN; RATFOR FEATURES IF THEN ELSE; WHILE AND UNTIL LOOPS; STRING VARIABLES AND SUBROUTINE CALLS. IT DOES NOT INCLUDE CALL BY NAME OR RECURSION. RATFOR PROGRAMS ARE EASILY CONVERTED TO ALGOL.

THE MAIN ADVANTAGE OF THIS BOOK IS THAT IT PROVIDES A SET OF PROGRAM DEVELOPMENT AND DATA MANIPULATION TOOLS THAT ARE MACHINE INDEPENDENT AND PORTABLE. THE BOOK MAY BE VALUABLE TO NUANCE READERS IN AT LEAST TWO WAYS: (1) IT IS THE CLAREST STRUCTURED PROGRAMMING TEXT YET WRITTEN; (2) THE PROGRAMS ARE VALUABLE IN THEMSELVES; AND EASILY EXTENDABLE. INDEED; IT WOULD BE EASY TO CHANGE THE RATFOR PRE-PROCESSOR TO PRODUCE ALGOL CODE. THE SOURCE VERSIONS OF THESE PROGRAMS (WITH SOME POST PUBLICATION ENHANCEMENTS) ARE AVAILABLE FROM ADDISON WESLEY ON AN 800 BPI 9TRACK EBCDIC TAPE FOR \$25. THE FIRST PROGRAM ON THE TAPE IS THE RATFOR TO FORTRAN CONVERTER IN ANSI FORTRAN4. [REVIEWED BY W. J. HUTCHISON; MATHTECH].

SCIENTIFIC AND ENGINEERING PROBLEM SOLVING WITH THE COMPUTER;
BY W. P. BENNETT, JR.
PRENTICE HALL 1976.

THIS IS THE TEXTBOOK FOR AN INTRODUCTORY COMPUTER COURSE AT YALE; THE COURSE MADE THE "TEN BEST" LIST IN THE COURSE CRITIQUE SEVERAL YEARS IN A ROW. NO WONDER; THERE ARE FEW TEXTS WHICH ARE AS APPEALING TO STUDENTS - AND EX-STUDENTS - BOTH IN THE SCIENCES AND IN HUMANITIES; AND STILL HAVE ENOUGH DEPTH OF SUBJECT TO CHALLENGE A MORE EXPERT AUDIENCE.

FROM A VERY SIMPLE INTRODUCTION IN COMPUTER OPERATION AND LANGUAGE; THE

AUTHOR PROCEEDS WITH DESCRIPTIONS OF GRAPHIC DISPLAYS; NOT JUST THE PLAIN \$TIJ PLOTTING VARIETY; BUT THREE DIMENSIONAL GRAPHICS WITH HIDDEN LINES AND STEREOSCOPIC VIEWS. OTHER CHAPTERS ARE GROUPED AROUND OTHER SPECIFIC OPERATIONS SUCH AS LANGUAGE. REMEMBER THE MONKEYS RANDOMLY TYFING SHAKESPEARE? HERE ARE SECOND AND THIRD ORDER APES WITH EXAMPLES OF THEIR OUTPUT. IT WOULD BE DIFFICULT TO FIND A CLEARER EXAMPLE OF THE USE OF STATISTICS AND CORRELATION; WITHOUT USING "HEAVY" MATHEMATICS AND STILL GIVE A CONVINCING DEMONSTRATION THAT MONKEYS WITH HIGHER ORDER CORRELATION WRITE MORE AND MORE LIKE SHAKESPEARE; POE; COUTINHO; OR BAUDELAIRE (IF YOU PREFER OTHER TONGUES).

THE CHAPTER ON DYNAMICS USES A RUNGE-KUTTA METHOD; DEMONSTRATED ON EXAMPLES RANGING FROM SKYDIVERS TO CYCLOTRONS. THE CHOICE OF EXAMPLES MAKES THIS MORE THAN A TEXTBOOK. WHERE ELSE CAN ONE FIND IN ONE PLACE REAL LIFE NUMERICAL DATA ON AIR RESISTANCE OF FALLING BODIES; TRAJECTORIES OF PARTICLES IN MAGNETIC BOTTLES; THE DIFFUSIVE SPREAD OF SYPHILIS IN MEDIAEVAL SPAIN; STATISTICS ON LETTER CORRELATION IN VARIOUS LANGUAGES; ACOUSTICAL SPECTRA OF VARIOUS INSTRUMENTS (INCLUDING A GARDEN HOSE) AND THE PROBABILITY OF FINDING ERASED MESSAGES IN WASHINGTON TAPES? TO TOP IT OFF; THERE IS A DESIGN FOR AN ECOLOGICAL; LASER OPERATED MOTH KILLER MACHINE; WITH FULL MODEL EQUATIONS AND COST EFFECTIVENESS DATA!

THE PURPOSE OF THIS BOOK IS TO TEACH FAMILIARITY WITH COMPUTER USE AND PROGRAMMING TECHNIQUES; TO WARD OFF ANY HESITANCY; THE AUTHOR HAS CHOSEN BASIC AS THE LANGUAGE FOR HIS EXAMPLES. THIS IS A PROPER CHOICE; ONLY 20 MINUTES "INDOCTRINATION" ARE REQUIRED BEFORE THE USER CAN WRITE AND RUN PROGRAMS FROM A TERMINAL. (EVEN FOR ALGOL; THIS TAKES ABOUT AN HOUR; FOR OTHER LANGUAGES; THE TIME IS COUNTED IN DAYS). FOR THE EXAMPLES CHOSEN; WHERE THE PROGRAMMING IS RATHER SIMPLE AND NOT COMPLICATED BY INTERACTION WITH MULTIPLE OUTPUT DEVICES; OTHER PROGRAM BLOCKS; ETC.; THE CHOICE OF LANGUAGE IS OPTIMUM. MOREOVER; THE PROGRAMS ARE READILY CONVERTED TO ALGOL PROCEDURES; IF THEY HAVE TO RUN IN MORE COMPLEX SURROUNDINGS.

EVEN FOR EXPERT COMPUTER USERS; THIS BOOK IS WELL WORTH READING - AND STUDYING. (SEE ALSO THE COVER FEATURE).

THE ELEMENTS OF PROGRAMMING STYLE
BY B. W. KERNIGHAN AND P. J. PLAUGER;
MCGRAW HILL 1974.

EVERY SERIOUS PROGRAMMER SHOULD READ THIS BOOK; WHETHER HE OR SHE WRITES FOR A LIVING; WRITES OCCASIONALLY AS AN AID IN GETTING PROBLEMS SOLVED; OR IS ONE OF THE NEW BREED OF HOBBY MICRO COMPUTERNIKS AT HOME; PLAYING WITH PROGRAMMED GAMES. THE AUTHORS TAKE PUBLISHED PROGRAMS FROM TEXTBOOKS; USE THESE AS EXAMPLES OF GOOD

PROGRAMMING STYLE; AND PROCEED TO SHOW HOW THE EXAMPLE CAN BE IMPROVED. REMARKABLY ENOUGH; THEY HAD NO PROBLEMS FINDING THEIR SOURCE MATERIAL; EVEN THOUGH PART OF THEIR EXAMPLES IS WRITTEN IN PL/I. THIS LANGUAGE HAS THE SAME GENERAL BLOCK STRUCTURE AS ALGOL; AND CAN BE DOCUMENTED SIMILARLY. THE OTHER LANGUAGE USED IS FORTRAN; IN WHICH IT IS MORE DIFFICULT TO WRITE CLEAR PROGRAMS. IT IS INTERESTING TO COMPARE THE "BAD" STYLE THAT OCCURS (OR PERHAPS DEVELOPS NATURALLY) IN THE TWO LANGUAGES. IN MY OPINION; AN ANCIENT LANGUAGE LIKE FORTRAN DOES NOT COMPARE WELL WITH THE MORE MODERN PL/I; AND I WONDER HOW OTHER LANGUAGES WOULD FAIR. THIS IS A QUESTION THAT SHOULD BE ASKED OF ANY LANGUAGE USED FOR SPECIFIC PURPOSE. I REMEMBER SUCH COMPARISONS ON ENGLISH; GERMAN; AND JAPANESE FOR SCIENTIFIC PAPERS; NEW OR POTENTIAL COMPUTER LANGUAGES SHOULD BE SUBJECTED TO THIS SCRUTINY BEFORE THEY BECOME EMBEDDED TOO STRONGLY.

END; END;

J. CELKO'S EOF SOLUTION

ONE POSSIBLE WAY IS TO PUT THE I/O INTO A BOOLEAN PROCEDURE WHICH IS TRUE WHEN THE EOF FLAG IS TRUE; AND FALSE WHEN THAT FLAG IS FALSE. THIS DEPENDS ON THE VARIABLES IN THE RECORD BEING GLOBAL TO THE READ PROCEDURE. THUS THE MAIN LINES LOOK LIKE THIS:

WHILE GETRECORD DO PROCESS; CLOSEUP;

IN DEC ALGOL ONE POSSIBLE IMPLEMENTATION OF THIS APPROACH COULD BE:

```

BOOLEAN PROCEDURE GETRECORD(A);
VALUE A; INTEGER R;
BEGIN BOOLEAN B;
B:=FALSE;
NEXT: READ(A,R1,R2,R3,...,EOF,GOODF);
RECCOUNT:=RECCOUNT+1; B:=TRUE; GOTO EXIT;
EOF: CLOSE(A); B:=FALSE; GOTO EXIT;
GOODF: GOTO NEXT;
/*RE-READ IS SHOWN; BUT ALSO COULD BE A
CALL TO ERROR; OR ANYTHING ELSE*/
EXIT: GETRECORD:=B
END OF GETRECORD;
    
```

HERE THE VARIABLES READ (R1, ...) AND THE RECORD COUNT ARE GLOBALS. IT IS ALSO POSSIBLE TO PUT THE OPEN FOR THIS FILE INTO THIS PROCEDURE; AND THEREBY ISOLATE ALL OF THE I/O IN ONE PROCEDURE BODY. THAT IS LEFT FOR A FURTHER EXERCISE (USE OWN BOOLEAN'S TO PREVENT OPENING THE SAME FILE AFTER THE FIRST CALL).

END; END;



NOTES ON ALGOL DOCUMENTATION AND BUGS

D0 - WHICH VERSION???

IN THE VARIOUS BUGS THAT WERE REPORTED, I FOUND THAT NOT ALL WERE BASED ON MY VERSION OF ALGOL (R1055.0, ALGOL3.2), BUT I HAVE NOT SYSTEMATICALLY TRIED ALL THE SUBMITTED ONES. OF THE 3 BUGS THAT C. R. ROMME SUBMITTED; ONLY ONE SURVIVED. SIMILARLY; D'CARROLL WRITES THAT BUG N2-D18 DID NOT SHOW UP ON HIS SYSTEM. EITHER THERE ARE MANY VERSIONS OF ALGOL STILL AFLOAT (WHY NOT SWITCH TO THE LAST AND BEST ONE?); OR THERE ARE BUGS WHICH RELATE TO THE SYSTEM AS A WHOLE (MAFFED/ FOREGROUND ETC.??). WITH NEW BUGS; PLEASE GIVE THE REVISION NUMBER; ETC. AND; OF COURSE; DON'T FORGET TO SUBMIT THE BUG ALSO ON A STR REPORT TO DEC. THE LATTER IS THE MORE IMPORTANT; BECAUSE THEY ARE THE ONLY ONES THAT CAN APPLY A "UNIFORM" REMEDY TO ALL ALGOL USERS.

D1 - IMBEDDED ASSIGNMENT

THE STATEMENT IF I:=INDEX(S1,S2)=0 THEN ...; DOES NOT WORK AS IT SHOULD. SOLUTION: USE EXTRA PARENTHESIS IF (I:=INDEX(S1,S2))=0 THEN ...;

D2 - MANUAL ERRORS [J. PEARCE]

THE EXAMPLE FOR OPERATOR CAT ON P.7-22 HAS AN ERROR: REPLACE ...+LENGTH(B)):=C BY ...+LENGTH(C)):=C. ON P.9-14; THERE ARE TWO IDENTICAL SENTENCES IN PARAGRAPH 2; "THE READ PROCEDURE IGNORES ...". ONE OF THESE SHOULD BE DELETED.

D3 - GTIME POINTERS

GTIME HAS TWO BUILT-IN POINTERS; DAYPTR AND MONPTR; TO NAMESTRINGS FOR DAYS AND MONTHS. THE INTERNAL DELIMITERS ARE NULLS E.G. MONDAY<NUL>TUESDAY<NUL>.. THESE POINTERS CAN BE USED AS FOLLOWS.
EXTERNAL POINTER DAYPTR,MONPTR;
INTEGER (2) JD; INTEGER MOD; 1;
STRING S; POINTER P; READ STRING BS;
BASED POINTER BP; /*OTHER DECLARATIONS*/
JD:=/*JULIAN DATE; SEE NUANCE2; D3*/
MOD:=JD/7*7; /*= JD MOD 7 */
P:=(DAYPTR+MOD)-BP;
LINEWRITE(3,P); /*NAME TO FILE
OR DEVICE; USES NULL AS DELIMITER*/
S:=P->BS; /*SET UP AS STRING*/
FOR I:=1,I+1 WHILE ASCII(S,I)<>0 DO;
SETCURRENT(S,I-1);
/*FOR USE OF NAME IN PROGRAM*/
NOTE THE DIFFERENCE IN HANDLING FOR OUTPUT AND FOR IN-CODE USE OF THE NAME STRINGS.

D4 - BEGIN - END [P. MAAS]

THE COMPILER DOES NOT CHECK TO SEE WHETHER THE NUMBER OF BEGIN'S AND END'S MATCHES; AND THERE IS NO RUN-TIME ERROR EITHER. PROGRAM DEBUGGING WOULD BE EASIER IF THIS CHECK WERE

D5 - D-ERRORS [P. MAZE]

STATEMENTS LIKE IF (FVE1 RELOP FVE2) THEN ... WHERE FVE ARE FLOATING POINT VARIABLES AND RELOP A RELATIONAL OPERATOR, CAUSE THE GENERATION OF ASSEMBLY STATEMENT FSUB N:M WHICH IS FLAGGED BY THE RSM AS A QUESTIONABLE (Q) ERROR. IN FACT, THE CODE GENERATED IS CORRECT FOR THE ALGOL FLOATING POINT PROCESSOR. IT WOULD HELP NOVICE PROGRAMMERS IF THE ALGOL MNEMONIC WERE CHANGED TO AVOID THIS IRRELEVANT ERROR MESSAGE.

[I AGREE. I HAVE BEEN CALLED BY SEVERAL PEOPLE WHO TRIED ALGOL ONCE, AND DID NOT DARE TO RUN THEIR PROGRAM AFTER THIS MESSAGE, IN FEAR THAT DISASTERS WERE TO BE EXPECTED ON THEIR DISK; THE POWER SUPPLY HAS TO TURN UP, ETC. NO WARNINGS OR MESSAGES ARE LISTED IN THE DOCUMENTATION EITHER. AVR]

D6 - SUBSTR FUNCTION [P. MAZE]

THE 1ST AND 2ND SENTENCE IN THE MANUAL (P.9-8) ARE WRONG. CHANGE THE LINE FROM "DIGITS OF AN INTEGER" TO READ "BYTES OF AN INTEGER" AND REPLACE THE EXAMPLE

```
INTEGER I,J;I:=1776;J:=SUBSTR(I,2,3);
```

```
BY
INTEGER I;STRING A;
I:=65+256*81;
SUBSTR(A,1):=SUBSTR(I,2);
```

THIS TRANSFERS THE LOW ORDER BYTE OF I WITHOUT TRANSLATION; AND IS THE INVERSE OPERATION OF ASCII/BYTE. THE ACTUAL OPERATION OF SUBSTR IS MUCH MORE VALUABLE THAN THE EXAMPLE IN THE MANUAL. OFTEN IT IS USEFUL TO BE ABLE TO MOVE BYTE PATTERNS WITHOUT TRANSLATION.

D7 - PROCEDURE CALLS [P. MAZE]

IN THE MANUAL, P.8-5, CHANGE THE HEADING "CALLING A PROCEDURE BY..." TO "CALLING A PARAMETER BY...". IT IS PARAMETERS, NOT PROCEDURES, WHICH ARE CALLED BY VALUE OR NAME. ALSO CHANGE "BY NAME" TO "BY REFERENCE": CALL BY NAME IS NOT IMPLEMENTED.

D8 - ASCII (BYTE) [P. MAZE]

MANUAL P.9-9: ADD THE SENTENCE "IF V IS AN INTEGER, THEN N#1 RETURNS THE HIGH ORDER BYTE OF INTEGER V; WHILE N#2 RETURNS THE LOW ORDER BYTE OF INTEGER V" IN THE LAST PARAGRAPH BEFORE THE EXAMPLE.

D9 - LINEREAD, -WRITE [P. MAZE]

SEVERAL ADDITIONS IN THE MANUAL, P.9-20 WILL CLARIFY OPERATION. WHEREVER THERE IS A REFERENCE TO THE VARIABLE "POINTER", ALTER THE WORDING TO READ "POINTER TO THE WORD IN MAIN MEMORY AT WHICH READING BEGINS OR FROM WHICH WRITING BEGINS". IN THE SECTION CALLED "PURPOSE", ADD TO THE SECOND PARAGRAPH THE SENTENCE "LINELLENGTH MUST BE LESS THAN 132 CHARACTERS; SEE RDS MANUAL." FINALLY, ADD TO THE COMMENT IN THE EXAMPLE "SETCURRENT MUST BE CALLED BECAUSE LINEREAD DOES NOT UPDATE THE STRING DESCRIPTOR."

D10 - BYTEREAD, -WRITE [P. MAZE]

CORRECT THE MANUAL (P.9-21) SIMILARLY TO THE CORRECTION ABOVE IN D9. ALTER THE COMMENT IN THE EXAMPLE TO READ "ALLOCATE ALLOCATES 100 WORDS IN MAIN MEMORY STARTING AT LOCATION POINTED TO BY POINTER BUFFER. BYTEREAD READS 200 BYTES FROM THE FILE INTO THE ALLOCATED AREA."

D11 - BYTE vs LINE READ AND WRITE

IN THESE FOUR PROCEDURES, IN ADDITION TO THE CORRECTIONS NOTED ABOVE, NOTE THAT THE MANUAL DOES NOT SPECIFY THAT THE PARAMETER COUNT IS CALLED BY VALUE IN THE BYTE PROCEDURES, AND CALLED BY REFERENCE IN THE LINE PROCEDURES. LINEREAD THUS RETURNS UPDATED VALUES FOR COUNT (E.G. AFTER ERROR READING); BUT BYTEREAD MAINTAINS THE INPUT VALUE FOR COUNT, INDEPENDENT OF THE ACTUAL NUMBER OF BYTES TRANSFERRED.

D12 - GLOBAL LABEL [L. M. O'CARROLL]

USE OF A GLOBAL LABEL IN A READ STATEMENT CAUSES LOOPING UNTIL THE DISK HAS FILLED UP WITH RUBBISH; E.G.

```
BEGIN ...
  BEGIN
    READ(1, DATA, READERR);...
```

```
  END;
```

```
READERR: ...
```

```
END;
```

DOES NOT WORK. SOLUTION: USE LABEL ONLY IN INSIDE BLOCK.

D13 - PHASE 2 ERROR [L. M. O'CARROLL]

THE CRYPTIC ERROR MESSAGE "FATAL ERROR FROM PHASE 2: FILE NOT OPEN: ALGOL.SV" MAY BE CAUSED BY TRYING TO SEND AN ERROR MESSAGE TO A LISTING FILE. SOLUTION: USE ALGOL/L.

D14 - PROCEDURE GLOBALS [L. M. O'CARROLL]

GLOBALS IN A PROCEDURE CAN LEAD TO ERRORS:

```
BOOLEAN ECHO; INTEGER J; ...
```

```
INTEGER PROCEDURE CHIN;
```

```
BEGIN INTEGER I;
```

```
  BYTEREAD(5, ADDRESS(I), 1);
```

```
  IF ECHO THEN BYTEWRITE(6, ADDRESS(I), 1);
```

```
  CHIN:=I/256;
```

```
END CHIN;
```

```
...
```

```
J:=CHIN; WRITE(6, " ", J); ...
```

GIVES CORRECT RESULTS IF ECHO= TRUE, BUT WRONG ONES WHEN IT IS FALSE. SOLVE BY USING ECHO AS A PARAMETER.

D15 - STRING PROCEDURES/ALLOCATE [L. M. O'CARROLL]

A PREMATURE JUMP OUT OF A STRING PROCEDURE VIA A LABEL PARAMETER, LEAVES THE TEMPORARY STRING SPACE STILL ALLOCATED:

```
STRING STR; ...
```

```
STR:= STRINGPROC (X,Y,Z,JUMP);
```

```
...
```

```
JUMP: ...
```

SOLUTION: DECLARE JUMP AS EDCLEAN, AND TEST IT ON EXIT FROM THE PROCEDURE. LABELS ARE A NECESSARY EVIL IN ANY CASE!

D16 - UNNOTICED ERRORS [L. M. O'CARROLL]

SEVERAL ERRORS ARE NOT DETECTED BY THE COMPILER. (A). THE "ASSIGNMENT" I:=I+1; IS CORRECTLY COMPILED AS I:=I+1. IS THIS A SOP TO FORTRAN PROGRAMMERS? (B). IF SUFFICIENT "END"'S APPEAR IN A PROGRAM, THE COMPILER APPEARS TO INSERT THEM AFTER REACHING THE END OF FILE. THIS CAN CONCEAL BLOCK ERRORS IN THE PROGRAM. (C). CONVERSELY, TOO MANY "END"'S CAN CAUSE PREMATURE TERMINATION OF COMPILEATION WITHOUT ERROR INDICATIONS. A CHECK THAT EQUAL BEGIN AND END MARKS ARE USED WOULD BE USEFUL. [HOWEVER, IF THERE ARE OTHER MYSTER, ERRORS DURING COMPILEATION, EDITING EXTRA ENDS AND

REMOVING THESE: IS A QUICKER METHOD OF LOCATING THE ERROR. SINGLE STATEMENTS ALWAYS CAN BE MADE NON-OPERATIONAL BY EDITING A COMMENT IN FRONT; WHEN /**/ IS USED FOR THE REGULAR COMMENTS; THIS IS ANOTHER FAST WAY OF LOCATING SUCH STATEMENTS. (AVR)

D17 - COMARG

THIS PROCEDURE HAS AN EDF LABEL AS PARAMETER; HOWEVER; THIS DOES NOT HAVE TO BE USED FOR PROPER STRING PARAMETER ASSIGNMENT. THE PROGRAM

```
TEST: BEGIN STRING ONE, TWO;
TWO:="$TTO";OPEN(0,"COM.COM");
COMARG(0,ONE);COMARG(0,TWO);... END;
CAN BE CALLED (FROM CLI) BY EITHER TEST <CR>,
OR BY TEST XX <CR>. IN BOTH CASES, ONE
CONTAINS "TEST"; IN THE FIRST CALL, TWO
CONTAINS "$TTO"; IN THE SECOND ONE (AS
EXPECTED) "XX". THIS CODE IS SHORTER THAN THE
EQUIVALENT USING EDF LABELS.
```

D18 - FORE/BACKGROUND

THE FOLLOWING SIMPLE CODE SETS A BOOLEAN VARIABLE TRUE WHEN THE PROGRAM RUNS IN THE FOREGROUND:

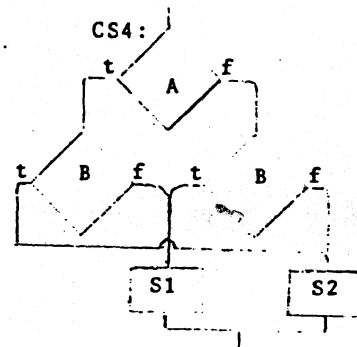
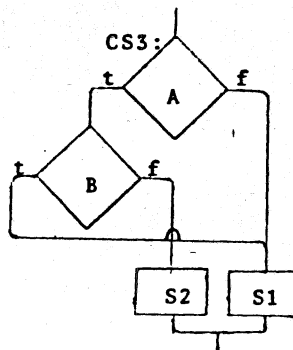
```
BOOLEAN FRG:=BASED BOOLEAN B:POINTER P;
P:=400R8; FRG:=P->B;
FRG IS FALSE IN BACKGROUND, TRUE IN
FOREGROUND. THIS CODE CAN BE USED TO OPEN THE
$TTO1 RATHER THAN THE $TTO AUTOMATICALLY;
MAKING PROGRAMS COMPATIBLE WITH BOTH
FORE/BACKGROUND OPERATIONS. (SEE ALSO CONSOLE
IN THE ALGOL LIBRARY DESCRIPTION).
```

D19 - CONDITIONAL STRING ASSIGNMENT (C. R. ROMME)

THE ASSIGNMENT ST1:= IF BOOL THEN ST2 ELSE ST3; OR THE EQUIVALENT ONE USING LITERALS (THEN "X" ELSE "Y") NEVER WORKS. IT IS ESSENTIAL TO REVERSE THE CODE: IF BOOL THEN ST1:=ST2 ELSE ST1:=ST3.

D20 - CONDITIONAL STATEMENTS

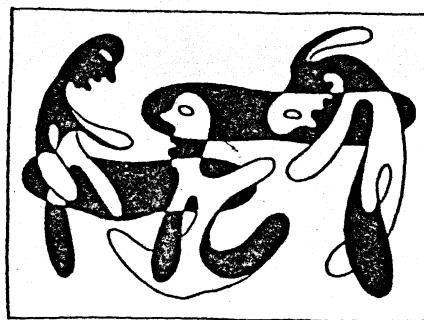
NUANCE1 HAD LOGIC DIAGRAMS FOR "IF A THEN S1 ELSE S2"; AND SOME OTHER ONES. BY REQUEST; HERE ARE THE DIAGRAMS FOR THE BOOLEAN COMBINATORIAL OPERATORS. FOR CLARITY; ALL TRUE BRANCHES GO LEFT. NOTE THE DOUBLE FORM FOR CS5; EXCHANGING ALL TRUE WITH FALSE IN CS4 GIVES THE DIAGRAM OF CS5. THE MANUAL; P.4-7 HAS AN ERROR IN THE LOGIC TABLE. THE LAST COLUMN IS NOT "Y OR Z" BUT "Y XOR Z".



- CS3: IF A IMP B THEN S1 ELSE S2;
- CS4: IF A XOR B THEN S1 ELSE S2;
- CS5: IF A EQV B THEN S1 ELSE S2;
- CS5: IF NOT A XOR NOT B THEN S1 ELSE S2;

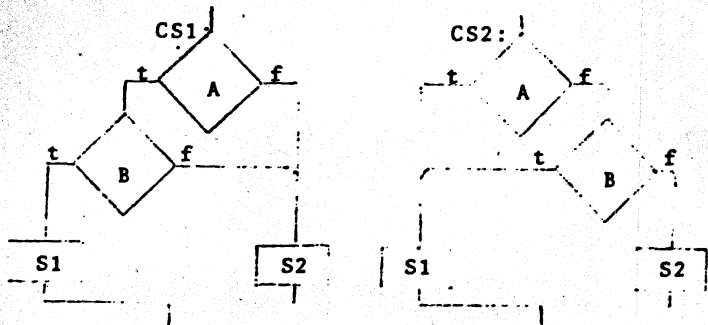
END; END;

THE COVER:



Third-order Monkeys, Typing not-so-random

THE BOOK REVIEW OF BENNETT'S PROBLEM SOLVING WITH THE COMPUTER MENTIONS THE APPLICATION OF STATISTICS AND CORRELATION TO LANGUAGE ANALYSIS. RANDOM TYPING ("ZERO ORDER") FROM THE WELL-KNOWN MONKEYS TRYING TO PRODUCE A WORK OF ART; GIVES COMPLETELY UNRECOGNIZABLE GARBAGE; BENNETT ALSO GIVES SOME INTERESTING DATA ON THE NUMBER OF BOOKS THAT HAVE TO BE READ BY THE CRITICS; BEFORE EVEN A SINGLE SONNET WOULD BE FOUND! FIRST ORDER STATISTICAL CORRELATION CAN BE MADE BY RECOGNIZING THE SPACE AS A VALID CHARACTER (IN ENGLISH IT OCCURS TWICE AS OFTEN AS THE E); AND WEIGHTING THE CHARACTERS AS TO THEIR FREQUENCY OF OCCURRENCE (SPACE 6934, E 3277, D 2578, T 2557, A 2043, ETC.). SECOND-ORDER RANDOM TYPING USES LETTER-PAIRS WITH THE CORRECT FREQUENCY (AA 0, AB 19, AC 63, AD 69, ETC.). AT THIS LEVEL; IT IS POSSIBLE TO RECOGNIZE LANGUAGE ROOTS. NOBODY WILL MISTAKE "ANEINGER HUNSTEIERE DELENINER MESTERUSTEITEINDEPOFOL GSCHIEISS" (GERMAN) OR "DES'UVICILEXINT JONS CENTE DE RETIRE PURS EA SYS DE ENSET LESS GDIRENUS" (FRENCH) ALTHOUGH THE SECOND ORDER SHAKESPEAREAN "FOR'T SESILOPK TITIFOLFELON HELIDRSHIT MY ACT MOUND HARCISTHER K EDMAT Y" MAY GIVE THE IMPRESSION THAT IT COMES FROM CHAUCER. SIMILARLY; THE ITALIAN "VONE SA FA DELIGNDUNID VILE N SESSUE AVA NCHIDIOMFIVORE LITOMO TI" MIGHT WELL BE AN ATTEMPT AT PORTUGUESE. THE COVER QUOTE IS THIRD ORDER. I.E. CHARACTER TRIPLET CORRELATION; IN SHAKESPEAREAN ENGLISH.



- CS1: IF A AND B THEN S1 ELSE S2;
- CS2: IF A OR B THEN S1 ELSE S2;

A. van Roggen

IN THE PREVIOUS NUANCE (2;P.15) A PROCEDURE WAS GIVEN FOR MATRIX INVERSION: DINVERT. THIS "OLD" INVERSION PROCEDURE USES REGULAR ALGOL; THE PROCEDURE INVERT BELOW USES AN IDENTICAL ALGORITHM AND CODING; BUT MAKES USE OF POINTERS AND BASED VARIABLES. THE ONLY DIFFERENCE IS IN THE DIMENSION TEST: BECAUSE OF THE POINTERS; THE REQUIREMENT OF EXACT MATCH OF THE MATRIX INDICES IS NOT REQUIRED. OBVIOUSLY; MATRIX SIZES HAVE TO BE SQUARE TO BE INVERTABLE. DINVERT CAN WORK WITH M(1:5;1:5); BUT NOT WITH M(0:4;0:4) OR P(-2:2;-2:2). FOR INVERT; ANY ONE OF THESE MATRICES CAN BE USED; AND GIVE THE SAME RESULTS IF THE CORRESPONDING ELEMENTS ARE IDENTICAL. THE PROCEDURES USE AN EXTERNAL PROCEDURE; DIM; FOR THE DIMENSION. CURIOUSLY ENOUGH; THIS PROCEDURE IS NOT AVAILABLE FROM DEC; AND IS SHOWN BELOW.

A PRACTICAL EXAMPLE OF THE USE OF BASED VARIABLES AND POINTERS CAN BE SEEN BY A DIRECT COMPARISON OF DINVERT AND INVERT. TOGETHER WITH THE TEXT IN THE ALGOL MANUAL (SECT. 7 AND 9); IS THIS ENOUGH TO GET "THE HANG" OF THE TECHNIQUE; OR IS A FULL ARTICLE NEEDED? COMMENTS PLEASE!

ONE THING NOT EXPLAINABLE AT THE MOMENT; IS THAT THE RUNTIME FOR THESE PROCEDURES IS IDENTICAL (TAKING A MATRIX; PRINTING THE TIME BEFORE AND AFTER 500 CALLS TO INVERT). WHERE DOES THE ARRAY INDEX CHECKING TIME GO? IS IT NEGLECTIBLE WITH THE OVERHEAD IN PIDS? THIS WAS RUN ON NOVAS40; HMPD; BUT NO FLOATING POINT HARDWARE. DOES ANYONE HAVE AN ANSWER?

SWAP with EXEC

NUANCE2 (P.15) PROMISED A PROCEDURE FOR SWAPPING TO ANOTHER .SV PROGRAM; CALLABLE FROM ALGOL. THIS PROCEDURE; EXEC; IS MADE FROM THE ALGOL CODE BELOW (IN EXEC.AL); BY USING ALGOL/S.

```
PROCEDURE EXEC(FNM;ERR);
VALUE FNM;STRING FNM;BOOLEAN ERR;
BEGIN
SUBSTR(FNM,LENGTH(FNM)+1):="<0>";
END EXEC;
```

NOW EDIT THE RESULTING EXEC.SR AND INSERT THE FOLLOWING ASM PATCH IN THE LOCATION SHOWN:

```
000203      ;STRING
LDA         0,S+0,3      ;FNM
SUB         1,1          ;SWAP ONLY
;SYSTEM
;EXEC
MOV         2,0;SKP
SUBD       0,0
STA        0,S+3,3      ;ERR
JSR        @RETURN
```

} insert

FINALLY; DO ASM EXEC; AND DELETE EXEC.AL. THE .SR AND .RE FILES CONTAIN THE SWAPPING PROCEDURE; CALLABLE AS SHOWN IN NUANCE2; SEE ALSO NUANCE1 P.10: ASM PROCEDURES.

```
/*♦♦♦INVERT♦♦♦*/
PROCEDURE INVERT(A);
REAL ARRAY A;
COMMENT INVERT: MATRIX A IN PLACE WITH PIVOT M
ETHOD;
```

```
BEGIN EXTERNAL INTEGER CNSD;
INTEGER N;EXTERNAL INTEGER PROCEDURE DIM;
IF NOT DIM(A)=2 THEN BEGIN WRITE(CNSD,"DIM. ER
FOR");GOTO ZZ END;
IF (N:=1+HEBUND(A,1)-LBOUND(A,1))<>1+HEBUND(A,
2)-LBOUND(A,2) THEN
BEGIN WRITE(CNSD,"MATRIX NOT SQUARE");GOTO ZZ
END;
```

```
BEGIN
LITERAL PROC(Z);POINTER Z;AP;BP;CP;
BASED INTEGER BI;BASED REAL BI;
INTEGER I;J;K;L;P;A1;A2;A3;A4;REAL Y;W;
I:=2*(N+1);ALLOCATE(BP,I);ALLOCATE(CP,I);
ALLOCATE(Z,N+1);AP:=ADDRESS(A);
FOR J:=1 STEP 1 UNTIL N DO (Z+J)->BI:=J;
FOR I:=1 STEP 1 UNTIL N DO
BEGIN
A4:=AP+PROC*(I-1)*(N+1);K:=I;Y:=A4->BR;L:=I-1;
P:=I+1;
```

```
FOR J:=P STEP 1 UNTIL N DO
BEGIN
W:=(AP+PROC*(I-1+(J-1)*N))->BR;
IF ABS(W)>ABS(Y) THEN BEGIN K:=J;Y:=W END;
END;
FOR J:=1 STEP 1 UNTIL N DO
BEGIN
A1:=AP+PROC*(J-1+(K-1)*N);A2:=AP+PROC*(J-1+(I-
1)*N);
A3:=AP+PROC*(I-1+(J-1)*N);
(CP+PROC*(J-1))->BR:=A1->BR;A1->BR:=A2->BR;
A2->BR:=-((CP+PROC*(J-1))->BR)/Y;
(CP+PROC*(J-1))->BR:=A3->BR;A3->BR:=Y
END;
```

```
A4->BR:=1/Y;J:=(Z+I)->BI;
(Z+I)->BI:=(Z+K)->BI;(Z+K)->BI:=J;
FOR K:=1 STEP 1 UNTIL L;P STEP 1 UNTIL N DO
FOR J:=1 STEP 1 UNTIL L;P STEP 1 UNTIL N DO
BEGIN
A1:=AP+PROC*(K-1+(J-1)*N);
A1->BR:=(A1->BR)-((BP+PROC*(J-1))->BR)+((CP+P
RC*(K-1))->BR);
```

```
END;
END;
L:=0;
BACK: L:=L+1;
AGAIN: K:=(Z+L)->BI;IF K=L THEN GOTO TEST;
PERMUTE: FOR I:=1 STEP 1 UNTIL N DO
BEGIN
A1:=AP+PROC*(L-1+(I-1)*N);A2:=AP+PROC*(K-1+(I-1
)*N);
W:=A1->BR;A1->BR:=A2->BR;A2->BR:=W
END;
(Z+L)->BI:=(Z+K)->BI;(Z+K)->BI:=K;GOTO AGAIN;
TEST: IF L<N THEN GOTO BACK;
FREE(Z);FREE(BP);FREE(CP);
END;
ZZ: END INVERT;
```

```
/*♦♦♦DIM♦♦♦*/ /*22 MAY 1977*/
INTEGER PROCEDURE DIM(A);
ARRAY A;
/* RETURNS DIMENSION OF ANY TYPE ARRAY */
BEGIN INTEGER SZ;R;L;H;D;
SZ:=SIZE(A);D:=0;
AA: D:=D+1;L:=LBOUND(A,D);H:=HEBUND(A,D);
R:=H-L+1;IF R=SZ THEN GOTO EXIT;
SZ:=SZ/R;GOTO AA;
ERRDR("ERRDR IN DIM PROCEDURE");
EXIT: DIM:=D;
END DIM;
```