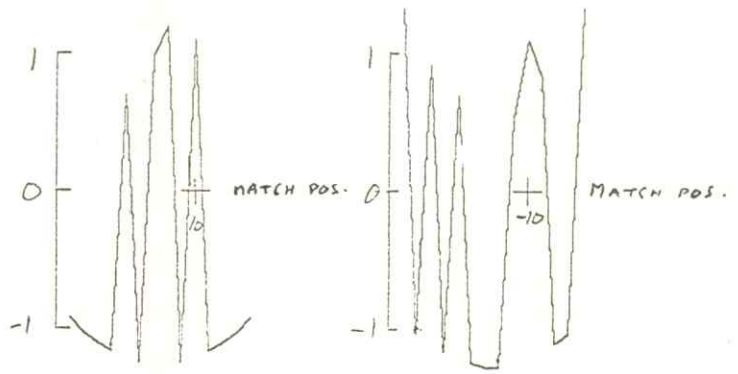
ALGOL Program for Cross-Association of Nonnumeric Sequences Using a Medium-Size Computer

By
Michael J. Sackin
Peter H. A. Sneath
MRC Unit, University of Leicester
and
Daniel F. Merriam
Kansas Geological Survey



SPECIAL DISTRIBUTION PUBLICATION 23

State Geological Survey
The University of Kansas, Lawrence
1965

One of the problems in geology, as well as other scientific disciplines, is matching strings of nonnumeric data. The matching process is loosely known as correlation. The nonnumeric data could be placement in a repetitive sequence, classes in a continuum, coded information of some aspect - such as lithology or color - or even arbitrary classes distinguished from numeric data.

Correlation is very important because it establishes contemporaneity and thereby allows interpretation of historical events. Where numeric data are available, time-trend analysis may be used to good advantage and, indeed, is a useful tool. Where nonnumeric data only are available (and this is often the situation in geology), arbitrary values are assigned. Observations may be plotted in sequence against their relative value, and by connecting the points a "form" curve is created. This curve, likewise, can be analysed by quantitative methods, but interpretation of results is difficult. Many examples of these "form" curves can be cited from the literature.

The program here described was developed originally to compare sequences of amino acids in protein chains. It can be used to find similarities, deletions, insertions, or inversions that are difficult to detect by visual methods and can also indicate cyclic structure if present. The program "slides" one sequence past another step by step and notes the number of matches for each position. The percentage match and statistical significance measures for each position are determined in order to find the best match in "overall similarity".

The geological application reported here is for correlating cyclic sequences of rock strata. The placement of a bed in its proper place in the cyclothem is determined by the investigator from lithology and fossil content, that is, an interpretation about the conditions under which it formed. Each unit is given a coded representation on placement in the sequence and then one string of such nonnumeric data is compared to another string at a different locality. By studying the sequences of environments in different localities a history of the area can be reconstructed. Obviously the final interpretation, even though based on a rigorous quantitative method, is only as reliable as the quality of the original interpretation, which is quite subjective.

This program is another in a series of computer contributions published by the Kansas Geological Survey; other available publications are listed on the inside back cover. A deck of cards for the FORTRAN IV version of the cross-association program may be obtained for a limited time from the Kansas Geological Survey for $10.00.

ALGOL Program for Cross-Association of Nonnumeric Sequences Using a Medium Size Computer

by

Michael J. Sackin, Peter H. A. Sneath, and
Daniel F. Merriam

INTRODUCTION

The problem of matching sequences of rock strata is a common one in geology, but it has parallels in many other disciplines. The matching of tree rings in sections of wood or of rainfall records for successive years are examples that at once come to mind. In many such problems the data are quantitative, or can be ordered at least into a series on a scale of intensity. They must, of course, also be ordered into a definite spatial or temporal sequence. In such instances the correlation coefficient is a useful measure of the degree of matching between two sequences, and by sliding the sequences past each other in small steps, the correlation can be determined for each successive step. This is the familiar technique of auto-correlation and cross-correlation. A method of testing significance of agreement between pairs of time series is described by Burnaby (1953).

There are other problems, however, where the data are not quantitative or are difficult to arrange on a scale of intensity. An example is a list of index fossils for a series of strata; their identity can be expressed in the form of a series of alternatives but not of intensities. In principle some material could be arranged on intensity scales, but a large number of scales would be required to express all the facts. A geological example is the sequence of rock types in a cyclothem, where in theory it would be possible to express the alternatives of, say, limestone, shale, coal, sandstone in the form of four quantitative chemical scales (content of calcium, aluminum, carbon, silica). But both for technical reasons and for reasons of computing efficiency, it is impracticable with present facilities to work with multivariate auto- and cross-correlation.

The computer program presented here was devised for comparing amino acid sequences of proteins. Description of its use for this purpose is given in Sackin and Sneath (1965), but a brief note may be of use here in explaining the principle. The program is one for auto- and cross-association, that is, the two sequences are slid past one another but the measure of concordance is an association coefficient rather than a correlation coefficient. Association coefficients are those where nonquantitative data are used, nearly always Yes-No data or data reducible to this form (for a discussion and listing, see, Sokal and Sneath, 1963, p. 125). Here, the association coefficient is the proportion of matches. It is impossible in practice, using association coefficients, to slide the two sequences in infinitely small steps, so that the method uses unit steps (or multiples of these) at each of which a Yes-No property is recorded.

Proteins are made up of chains of amino acids linked in a uniform manner, so that a protein chain can be described by a single unique sequence of amino acids just like the links of a chain. Each unit step is therefore an amino acid, and about twenty different alternatives are possible at each position. Using the common abbreviations for amino acids*, a chain might start as:

Amino acid: Val-Leu-Ileu-Tyr-Gly-His- .....

Position in chain 1 2 3 4 5 6 .....

and end as:

.....Phe-Val-Val-Arg-Leu

.....128 129 130 131 132

In searching for meaningful structure in these sequences we can slide two chains past each other one position at a time and calculate an association coefficient for the overlapping segments. Thus, suppose one had two chains of five amino acids, Ala-Phe-Ala-Leu-Tyr, and Ala-Phe-Ala-Val-Tyr; it is evident at once by inspection that they are identical except in position 4. With two long chains, however, it is not easy to see such obvious relations, but they can be detected by the sliding process. Two match positions are shown below:

Match position 1 Ala-Phe-Ala-Leu-Tyr

Ala-Phe-Ala-Val-Tyr

Agreement (matches) No

Match position 5 Ala-Phe-Ala-Leu-Tyr

Ala-Phe-Ala-Val-Tyr

Agreement (matches) Yes Yes Yes No Yes

In the first match position the agreements are 0 out of 1; in the fifth match position they are 4 out of 5. One can now estimate the probability that an agreement of 4 out of 5 would occur by chance, and thus obtain both a measure of resemblance and an estimate of significance. The association coefficient used is the simple matching coefficient, or proportion of agreements.

This method is obviously suited to matching sequences whose relative positions are uncertain; they are slid past each other and the position of best match is found. But other information can also be obtained. If there is a deletion in one chain (or an insertion in the other), one cannot match the chains perfectly at any single position, but instead one finds two positions of good match. In the following example the top chain has His inserted, otherwise the two chains are the same.

* It should be mentioned that ALGOL only handles numerical data, and that although words and letters are used here as illustrations, the actual input data consist of a series of numbers, according to some coding scheme such as 1=Ala, 2=Arg, etc. Automatic translation programs, however, can be written to make the conversion, and one for amino acids is available.

Match position 4	Tyr-Phe-His-Ala-Gly
	Tyr-Phe-Ala-Gly
Agreement (matches)	Yes Yes No No
Match position 5	Tyr-Phe-His-Ala-Gly
	Tyr-Phe-Ala-Gly
Agreement (matches)	No No Yes Yes

If a graph of the degree of matching is made for each successive match position, two peaks are found, indicating a deletion or insertion, and the distance between the peaks indicates the size of the deletion or insertion. A series of peaks indicates a cyclic structure (i.e. a series of similar subsequences). A reduplication shows as a minor peak. An inversion can be detected by repeating the sliding process with one chain reversed. The inverted section then gives a peak when it is opposite its noninverted analog.

With sequences which do not show a single position of excellent matching, some measure of "overall" resemblance of the sequences besides the one mentioned above (i.e. the proportion of matches at the best-fitting position) is useful. This can be obtained by cumulative statistics summed over all match positions, and several are given by this program. One can also print out the sequences in any desired position of overlap for further study of details of the matches. The geological analogs of deletions, insertions, inversions, reduplications, cyclic subsequences, and overall resemblance will come readily to mind.

Finally, one need not slide the chains one position at a time. One could slide them, for example, in steps of three, so that only the 3rd, 6th, 9th...match positions are studied. While not of interest in protein studies, this has an important use in other applications, for it allows more than one property to be scored for each position of primary data. Taking a short illustrative sequence of four stratigraphic units, one could for example, allocate to positions 1, 4, 7, 10 the alternatives limy - nonlimy, to positions 2, 5, 8, 11, the alternatives fossils present - fossils absent, and to positions 3, 6, 9, 12 the alternatives red - white. Then by comparing the sequences only at match positions 3, 6, 9 and 12, all three variables are used simultaneously, e.g. for match position 3:

		Bed I	Bed II
Match position	...nonlimy fossils red	limy fossils white	limy no fossils white...
	Bed III	limy fossils white	
		Bed IV	
Agreement (matches)		yes yes yes	

In general one can use T positions per rock unit and slide the sequences in steps of T. This then gives an association coefficient analagous to a multiple correlation coefficient. Although not such a precise statistic as the latter, it is simpler and quicker to compute and should be adequate for a wide range of applications. It is possible to introduce a measure of quantitation if necessary by the following device, analagous to the coding technique in taxonomy (Sokal and

Sneath, 1963, p. 76). Suppose the attribute is divided into three levels (besides "absent"), i.e. "weak", "medium", "strong", one can use three positions and code them as below:

Level	Positions	1	2	3
absent		1	1	1
weak		2	1	1
medium		2	2	1
strong		2	2	2

The effect of this coding scheme is to give more matches when the comparison is between adjacent levels than widely separated levels, thus making the matches reflect the quantitative values of the attribute. Note that 1 is used instead of the usual zero, because in this ALGOL program zero is reserved for "unknown".

We have shown with random numbers that the program gives results very close to those expected statistically. Although the probability distribution has been simplified (from the multinomial one to the binomial), we believe that this will make little practical difference. Provided that fairly stringent levels of significance are chosen, the results should prove very reliable. One point requires emphasis; with long sequences, giving say 400 match positions, little weight can be put on any individual value of probability of 1:400 for example, because one would expect about one of these in four hundred. In nearly perfect matches the probabilities are not in question, often being astronomical. Thus, with two typical protein chains each of 100 amino acids, one would expect about 7 matches in a hundred by chance. Even 50 matches in a hundred gives a nominal probability of over $1:10^{50}$. These extreme probability values are not very useful, because the basic assumptions may not be realistic; the number of standard deviations from the mean is a more convenient statistic (in this example it is 16.9).

NOTE ON ELLIOTT ALGOL

The program description is that of the program as written in Elliott's implementation of ALGOL 60 (Elliott, Ltd., 1965). To assist in comprehension and to enable the user to modify the program for his machine if required, some explanation of Elliott ALGOL is necessary. If an introduction to ALGOL 60 is required see Dijkstra (1962). This book also contains the official ALGOL 60 report.

Hardware Representation

The program was punched onto 5-hole paper tape, so rendering some of the standard ALGOL characters not available in view of the restricted character set of the 5-hole telecode. We consider it unnecessary to detail fully the differences in hardware representation and trust that they will be clear from the context (Table 1).

Title

All Elliott ALGOL programs start with a title which consists of all characters up to the first semicolon in the program. The compiler ignores these characters.

Table 1.- Program listing (version with plotter procedures).

```
ALGOL PROGRAM FOR CROSS-ASSOCIATION OF NON-NUMERIC SEQUENCES
USING AN ELLIOTT 803 COMPUTER - VERSION WITH GRAPH PLOTTER.
M.J.SACKIN, MEDICAL RESEARCH COUNCIL UNIT, LEICESTER UNIVERSITY,
ENGLAND: OCTOBER 1965'
BEGIN REAL LAMBDA,MU'
INTEGER AMATCH,ACOMPS,APROP,ADEVS,ACHISQ,
ACHSQY,AGRAPH,AREV,ASIM,T,L,M,ABSC,ORD'
COMMENT IF NUMBER OF STD DEVS OF NO. OF MATCHES FROM THE MEAN
LIES BETWEEN LAMBDA AND MU THEN THE TABLE-ROW FOR THE
CORRESPONDING MATCH POSITION WILL NOT BE PUNCHED.
      AMATCH,ACOMPS,.....,ACHSQY ARE THE DIRECTIVES
FOR PUNCHING OR NOT PUNCHING THE VARIOUS TABLE COLUMNS,
TOGETHER WITH THEIR COLUMN HEADINGS. PUNCHING WILL BE
SUPPRESSED IF DIRECTIVE=0. LIKEWISE AGRAPH=0
WILL SUPPRESS OUTPUT OF GRAPH.
      AREV=0 SIMILARLY SUPPRESSES ALL COMPUTATION
RELATING TO REVERSE MATCHES, INCLUDING PART OF THE
COMPUTATION OF THE SIMILARITY COEFFICIENT S(L).
      ASIM=0 SUPPRESSES COMPUTATION OF SIMILARITY
COEFFICIENT S(L).
      T=SLIDING STEP.
      L,M ARE THE LENGTHS OF THE TWO CHAINS'
PROCEDURE LINE(A,B)' VALUE A,B' INTEGER A,B'
COMMENT PLOTS BEST LINE FROM CURRENT VALUES ( IN LINE-SEGMENTS)
      OF (ABSC,ORD) TO (A,B).  SETS ABSC:=A,  ORD:=B.
      PEN STAYS ON OR OFF THE PAPER.
      +ABSC DIRECTION IS TOWARDS RIGHT-HAND EDGE OF PAPER.
      +ORD IS TOWARDS END OF ROLL, I.E. TOWARDS TOP OF PAPER'
BEGIN INTEGER MODA,MODB,QUA,SEG,AX1,AX2,COUNT,
      MOVE1, MOVE2,I,U,V,S'
      SWITCH SS:=L1,L2,L3,L4,L5,L6,L7,L8,L9'
      A:=A-ABSC' ABSC:=ABSC+A' MODA:=ABS(A)'
      B:=B-ORD' ORD:=ORD+B' MODB:=ABS(B)'
      IF A GREQ 0 THEN
      BEGIN IF B GREQ 0 THEN QUA:=0 ELSE QUA:=6
      END ELSE
      BEGIN IF B GREQ 0 THEN QUA:=2 ELSE QUA:=4
      END'
      IF MODA GREQ MODB THEN
      BEGIN SEG:=1' MOVE1:=ABS(MODA-MODB)' MOVE2:=MODB' S:=MODA
      END ELSE
      BEGIN SEG:=2' MOVE1:=ABS(MODB-MODA)' MOVE2:=MODA' S:=MODB
      END'
      GOTO SS(QUA+SEG)'
L1:   AX1:=5' AX2:=1' GOTO L9'
L2:   AX1:=5' AX2:=4' GOTO L9'
L3:   AX1:=6' AX2:=2' GOTO L9'
L4:   AX1:=6' AX2:=4' GOTO L9'
L5:   AX1:=10' AX2:=2' GOTO L9'
L6:   AX1:=10' AX2:=8' GOTO L9'
L7:   AX1:=9' AX2:=1' GOTO L9'
L8:   AX1:=9' AX2:=8' GOTO L9'
L9:   COUNT:=0'
      FOR I:=1 STEP 1 UNTIL S DO
      BEGIN U:=COUNT+MOVE2'
            V:=COUNT-MOVE1'
            IF ABS(U) LESSEQ ABS(V) THEN
            BEGIN COUNT:=U' ELLIOTT(0,0,AX2,1,7,2,7168)
            END ELSE
            BEGIN COUNT:=V' ELLIOTT(0,0,AX1,1,7,2,7168)
            END
      END
END'
PROCEDURE PENRAISE' ELLIOTT(7,2,7184,0,7,2,7184)'
PROCEDURE PENLOWER' ELLIOTT(7,2,7200,0,7,2,7200)'
READ LAMBDA' READ MU'
IF LAMBDA-.0000001*ABS(LAMBDA)  GR MU THEN
BEGIN PRINT  ££L?ERROR IN BOUNDS OF STD DEVS?' STOP
END'
```

```
COMMENT £ ? ARE STRING QUOTES.  ££L3?? = 3 NEW LINES, ££S5?? = 5
      SPACES, ETC.'
READ AMATCH,ACOMPS,APROP,ADEVS,ACHISQ,ACHSQY,
AGRAPH,AREV,ASIM,T'
IF T LESSEQ 0 THEN
BEGIN PRINT ££L?NON-POSITIVE SLIDING-STEP?' STOP
END'
WAIT' READ L'
BEGIN INTEGER I,J'
      INTEGER ARRAY PRO1(1:L)'
      FOR I:=1 STEP 1 UNTIL L DO READ PRO1(I)'
      WAIT' READ M'
      BEGIN INTEGER ACID,PR1ACID,PR2ACID,TOTALMATCHES,ACIDTYPES,
            MATCHES,COMPS,BASE1,BASE2,K,DEST,X,Y,OVERLAP,
            W,FDM,SIM,ZERO,Z1,Z2,SUMZ1,SUMZ2,SUMZ1Z2'
            REAL P,MATCHPROP,STDDEVS,CHISQ,CHSQY,SUMCHISQ,Z'
            BOOLEAN SUPPRESS,NOPT'
            SWITCH SW:=SLIDE,SIMCOEFF,DELETE,TEST,RETURN,
            SETI,SETJ,TESTREV' INTEGER ARRAY PRO2(1:M)'
            INTEGER PROCEDURE MAX(A,N)' VALUE N' INTEGER N'
                  INTEGER ARRAY A'
            BEGIN INTEGER I,P'
                  P:=A(1)'
                  FOR I:=1 STEP 1 UNTIL N DO
                  BEGIN IF A(I) GR P THEN P:=A(I) END' MAX:=P
            END MAX'
            FOR I:=1 STEP 1 UNTIL M DO READ PRO2(I)'
            NOPT:=AMATCH=0 AND ACOMPS=0 AND APROP=0 AND ADEVS=0 AND
            ACHISQ=0 AND ACHSQY=0'
            IF NOPT AND AGRAPH=0 THEN GOTO SIMCOEFF'
            PRINT ££L?SLIDING STEP =?,SAMELINE,DIGITS(3),T'
            COMMENT DIGITS(3) = UP TO 3 DIGITS PRINTED.
                  USED FOR INTEGERS'
            COMMENT CALCULATE PROB. P OF A MATCH'
            TOTALMATCHES:=0' ACIDTYPES:=IF MAX(PRO2,M) GR
                  MAX(PRO1,L) THEN MAX(PRO2,M) ELSE MAX(PRO1,L)'
            SUMZ1Z2:=0'
            FOR J:=1 STEP 1 UNTIL T DO
            BEGIN FOR ACID:=1 STEP 1 UNTIL ACIDTYPES DO
                  BEGIN PR1ACID:=PR2ACID:=0'
                        FOR I:=J STEP T UNTIL L DO
                        BEGIN IF PRO1(I)=ACID THEN PR1ACID:=PR1ACID+1
                        END'
                        FOR I:=J STEP T UNTIL M DO
                        BEGIN IF PRO2(I)=ACID THEN PR2ACID:=PR2ACID+1
                        END'
                        TOTALMATCHES:=TOTALMATCHES+PR1ACID*PR2ACID
                  END ACID'
                  Z1:=L DIV T' Z2:=M DIV T'
                  FOR I:=J STEP T UNTIL L DO
                        Z1:=Z1 - SIGN(PRO1(I))'
                  FOR I:=J STEP T UNTIL M DO
                        Z2:=Z2 - SIGN(PRO2(I))'
                  SUMZ1Z2:=SUMZ1Z2 + Z1*Z2
            END J'
            SUMZ1:=L'
            FOR I:=1 STEP 1 UNTIL L DO
                  SUMZ1:=SUMZ1 - SIGN(PRO1(I))'
            SUMZ2:=M'
            FOR I:=1 STEP 1 UNTIL M DO
                  SUMZ2:=SUMZ2 - SIGN(PRO2(I))'
            ZERO:=(M*SUMZ1 + L*SUMZ2) DIV T - SUMZ1Z2'
            COMMENT ZERO = NO. OF INVALID COMPARISONS, DUE TO
                  UNKNOWN (ZERO) ELEMENTS'
            P:=ABS(TOTALMATCHES*T/(L*M - ZERO*T))'
            IF P GR 1 THEN P:=1'
            PRINT ££L?PROB(MATCH) =?, SAMELINE,P'
            DEST:=1'
            PRINT ££L2S25?FORWARD£S?MATCHES£L2??'
            COMMENT NOW FOLLOWS THE MATCHING, STARTING WITH THE
            OUTPUT OF THE COLUMN HEADINGS'
      SLIDE:IF NOT NOPT THEN
            BEGIN PRINT ££S?MATCH£S??'
```

```
      IF AMATCH NOTEQ 0 THEN PRINT ££S?NO.£S?OF£S2??'
      IF ACOMPS NOTEQ 0 THEN PRINT ££S?NO.£S?OF£S??'
      IF APROP NOTEQ 0 THEN PRINT ££S?MATCHES£S??'
      IF ADEVS NOTEQ 0 THEN PRINT ££S2?STD£S2??'
      IF ACHISQ NOTEQ 0 THEN PRINT ££S3?CHI-SQ£S3??'
      IF ACHSQY NOTEQ 0 THEN PRINT ££S3?CHI-SQ£S3??'
      PRINT ££LS2?POS£S2??'
      IF AMATCH NOTEQ 0 THEN PRINT ££S?MATCHES£S??'
      IF ACOMPS NOTEQ 0 THEN PRINT ££S?COMPS£S2??'
      IF APROP NOTEQ 0 THEN PRINT ££S?/COMPS£S2??'
      IF ADEVS NOTEQ 0 THEN PRINT ££S?DEVS £S??'
      IF ACHISQ NOTEQ 0 THEN PRINT ££S?UNCORRECTED?'
      IF ACHSQY NOTEQ 0 THEN PRINT ££S3?(YATES)£S2??'
      PRINT ££L??'
END HEADINGS'
SUMCHISQ:=0' ABSC:=-520' ORD:=0' FDM:=0'
COMMENT PLOT Y-AXIS OF GRAPH IF REQUIRED'
IF AGRAPH NOTEQ 0 THEN
BEGIN PENLOWER'
      FOR I:=-500 STEP 100 UNTIL +500 DO
      BEGIN LINE(I,0)' LINE(I,10)' LINE(I,0)
      END'
      LINE(520,0)' PENRAISE' LINE(0,-10)'
      PENLOWER' LINE(0,0)' PENRAISE
END'
FOR K:=T STEP T UNTIL L+M-T DO
BEGIN COMPS:=MATCHES:=0'
      BASE1:=IF K LESS M THEN 0 ELSE (K-M)'
      BASE2:=IF K LESS M THEN (M-K) ELSE 0'
      OVERLAP:=IF K LESS M THEN
            (IF K LESS L THEN K ELSE L) ELSE
            (IF K LESS L THEN M ELSE L+M-K)'
      COMMENT OVERLAP = MIN(K,M,L,L+M-K)'
      FOR I:=1 STEP 1 UNTIL OVERLAP DO
      BEGIN IF PRO1(BASE1+I)=PRO2(BASE2+I)
            AND PRO1(BASE1+I) NOTEQ 0 THEN
            MATCHES:=MATCHES+1'
           COMPS:=COMPS+SIGN(PRO1(BASE1+I)*PRO2(BASE2+I))
     END COUNTING UP THE MATCHES AND COMPARISONS'
      COMMENT COMPS = MIN(K,M,L,L+M-K)-NO. OF
            UNKNOWN COMPARISONS'
      FDM:=FDM+SIGN(COMPS)'
      IF COMPS NOTEQ 0 THEN
      BEGIN MATCHPROP:=
                  (MATCHES*1.0000001+COMPS*0.0000001)/
                  (COMPS*1.0000001)'
            STDDEVS:=2*SQRT(COMPS)*
                  (ARCSIN(SQRT(MATCHPROP))-ARCSIN(SQRT(P)))
      END ELSE
      STDDEVS:=0'
      SUPPRESS:=LAMBDA LESS STDDEVS AND STDDEVS LESS MU'
      IF NOT SUPPRESS AND NOT NOPT THEN
      BEGIN PRINT PREFIX(££LS??),DIGITS(4),K*DEST,£££??'
           IF AMATCH NOTEQ 0 THEN PRINT ££S2??,DIGITS(4),
                  SAMELINE,MATCHES,£ES2??'
            IF ACOMPS NOTEQ 0 THEN PRINT ££S??,DIGITS(4),
                  SAMELINE,COMPS,£ES2??'
            IF COMPS NOTEQ 0 THEN
            BEGIN IF APROP NOTEQ 0 THEN PRINT FREEPOINT
                  (4),PREFIX(£ES??), MATCHPROP,£ES2??'
                  IF ADEVS NOTEQ 0 THEN PRINT
                  FREEPOINT(5),SAMELINE,STDDEVS
            END
      END'
      COMMENT FREEPOINT(4) = UP TO 4 DIGITS PRINTED.
            USED FOR REALS'
      IF AGRAPH NOTEQ 0 THEN
      BEGIN IF STDDEVS=0 THEN  Z:=0
            ELSE Z:=-EXP(LN(ABS(STDDEVS))/3)*100
                  *SIGN(STDDEVS)' X:=K*10 DIV T'
            LINE(Z,X)'
            IF K=T THEN PENLOWER'
            IF K=10*T*(K DIV (10*T)) THEN
            BEGIN PENRAISE' LINE(-10,X)'
```

```
                        PENLOWER' LINE(10,X)'
                        PENRAISE' LINE(0,X+10)'
                        PENLOWER' LINE(0,X-10)'
                        PENRAISE' LINE(Z,X)'
                        PENLOWER
                  END
            END'
            IF ACHISQ NOTEQ 0 AND COMPS NOTEQ 0 THEN
            BEGIN CHISQ:=(MATCHES-P*COMPS)**2/(P*COMPS*(1-P))'
                  SUMCHISQ:=SUMCHISQ+CHISQ'
                  IF NOT SUPPRESS  THEN PRINT ALIGNED(5,4),
                        SAMELINE,CHISQ,ΣΣS??
            END'
            IF ACHSQY NOTEQ 0 AND COMPS NOTEQ 0 THEN
            BEGIN CHSQY:=(ABS(MATCHES-P*COMPS)-0.5)**2
                        /(P*COMPS*(1-P))'
                  IF NOT SUPPRESS THEN PRINT ALIGNED(5,4),
                        SAMELINE,CHSQY,ΣΣS??
            END
      END MATCHING'
      COMMENT ALIGNED (5,4) = IN FORM DDDDD.DDDD'
      IF ACHISQ NOTEQ 0 THEN
      BEGIN PRINT PREFIX(ΣΣL2?SUM CHI-SQ (UNCORRECTED) =?),
            ALIGNED(5,4),SUMCHISQ,PREFIX(Σ, WHICH ISΣLS5??),
            FREEPOINT(8), SQRT(2*SUMCHISQ)-SQRT(2*FDM-1),
            Σ STD DEVS FROM THE MEAN (NORMAL APPROX)?
      END'
      IF AREV=0 THEN GOTO SIMCOEFF'
      COMMENT REVERSE SECOND CHAIN'
      K:=M DIV (2*T)'
      FOR I:=1 STEP 1 UNTIL K DO
      BEGIN FOR J:=1 STEP 1 UNTIL T DO
            BEGIN W:=PRO2((I-1)*T+J)'
                  PRO2((I-1)*T+J):=PRO2(M-I*T+J)'
                  PRO2(M-I*T+J):=W
            END
      END REVERSING SECOND CHAIN'
      IF DEST=-1 THEN GOTO SIMCOEFF'
      DEST:=-1'
      IF AGRAPH NOTEQ 0 THEN
      BEGIN PENRAISE' LINE(-520,ORD+100)' ORD:=0
      END RESETTING GRAPH'
      PRINT ΣΣR200L3S25?REVERSEΣS?MATCHESΣL2??'
      COMMENT ΣΣR200?? = 200 BLANKS'
      GOTO SLIDE'
SIMCOEFF: IF ASIM=0 THEN STOP'
      DEST:=1' SIM:=0'
      COMMENT FINAL SIM/(MAX(L,M)-1)=SIM COEFF'
   SETI: I:=1'
   SETJ: J:=1'
   TEST: IF PRO1(I)=PRO2(J) AND PRO1(I+1)=PRO2(J+1)
      AND PRO1(I) NOTEQ 0 AND PRO1(I+1) NOTEQ 0 THEN
            GOTO DELETE'
      IF J LESS M-1 THEN
      BEGIN J:=J+1' GOTO TEST
      END'
      IF I LESS L-1 THEN
      BEGIN I:=I+1' GOTO SETJ
      END'
TESTREV: IF DEST=1 AND AREV NOTEQ 0 THEN
      BEGIN DEST:=-1'
            FOR K:=1 STEP 1 UNTIL M DIV 2 DO
            BEGIN W:=PRO2(K)' PRO2(K):=PRO2(M+1-K)'
                  PRO2(M+1-K):=W
            END REVERSING CHAIN 2'
            GOTO SETI
      END'
      PRINT PREFIX(ΣΣL2?SIMILARITY INDEX S(L) =?),
            SIM/((IF L GR M THEN L ELSE M)-1)' STOP'
      COMMENT REMAINDER OF PROGRAM DEALS WITH CASE WHERE A
      MATCH OF 2 OR MORE PAIRS OF RESIDUES HAS BEEN FOUND'
   DELETE: PRO1(I):=-1' PRO2(J):=-2' COMMENT DELETES
      MATCHED ACIDS'
      K:=1'
```

```
        RETURN: PRO1(I+K):=-1' PRO2(J+K):=-2'
              COMMENT THUS CONTINUING DELETIONS'
              SIM:=SIM+1'
              IF I+K=L THEN GOTO TESTREV'
              IF J+K=M THEN
              BEGIN I:=I+K' GOTO SETJ
              END'
              K:=K+1'
              IF PRO1(I+K)=PRO2(J+K) AND PRO1(I+K) NOTEQ 0
                    THEN GOTO RETURN'
              I:=I+K-1' J:=J+K-1' GOTO TEST
        END
    END
    END PROGRAM'
```

Plotter Procedures

The procedures, LINE, PENRAISE, and PENLOWER make use of an online digital plotter. To program for this device some Elliott 803 machine-code is required in the procedures. These machine orders are contained in the standard procedures ELLIOTT and occur nowhere else in the program. We have therefore prepared a version of the program in which these procedures have been replaced by dummy procedures. In this version all running options other than those which involve the use of the digital plotter will still be available. Note that the variables ABSC and ORD must still be declared.

If desired, it should be possible for the user to rewrite the plotter procedures for a line printer, i.e. producing graphs similar in form to those of Fox (1964).

The procedures READ, PRINT, STOP, and WAIT, which are among the standard procedures of Elliott ALGOL and are thus not explicitly declared in the program, are briefly described.

READ statements

The numbers to be read should conform to the ALGOL definition of a number. Spaces, letters, and some nonnumeric characters may be used as number-separators. Input is from paper tape.

PRINT statements

The output device to which the print statements refer is the paper tape punch which may be fed with 5-hole or 8-hole tape. The meanings of the format settings used are explained in outline in the comments in the body of the program (Table 1), as are the means of outputting strings. Further clarification may be obtained by examination of the sample output.

The procedures STOP and WAIT

The procedure STOP causes an immediate end to the program.

The procedure WAIT causes the program to be held up until the operator makes a change to the keyboard setting allowing time to feed in new data tapes during the course of a run.

Miscellaneous

All labels must be declared as switch-elements in the heads of the blocks in which they are found.

INPUT DATA TO THE PROGRAM

Input is on paper tape, which may be either 5- or 8-hole tape.

For each run there will normally be three data tapes. Tape 1 is the parameter tape. Discussion of this tape is best deferred until the output has been described.

Tapes 2 and 3 consist of the two chains to be compared.

Each of these two tapes comprises the chain-length followed by the actual sequence (Fig. 1), the elements of which are coded as positive integers. Zeros represent unknown elements and do not enter into comparisons.

7 1 2 3 9 8 4 7 8 1 2 3 4 7 8 9 7

Figure 1.- Sample input.

PROGRAM METHOD AND OUTPUT

The program offers several running options, most of which can be suppressed in any combination in an actual run. The description here will be geared to an example using the input data of Figure 1 and the corresponding output (Table 2), in which all the options have been included.

Sliding process

The sliding process forms the bulk of the program.

After the parameter tape has been read, the program reads the two chains (of length L, M) and places them side by side in as many positions (viz. L+M-1) as will give an overlap.

For example:

```
Match position 1                        1 2 3 9 8 4 7
                          1 2 3 4 7 8 9 7

Match position 2                        1 2 3 9 8 4 7
                            1 2 3 4 7 8 9 7

Match position 14                       1 2 3 9 8 4 7
                                                  1 2 3 4 7 8 9 7
```

Forward matches

For each position the program calculates and records the following values in tabular form, any row of the resulting table containing:

(i) the current match position (a number in the range 1 to L+M-1);

(ii) the number of matches, excluding matches of zero (unknown) elements;

(iii) the number of comparisons, i.e. size of the overlap, not counting any comparison with a zero element in either chain;

(iv) the ratio of matches to comparisons, i.e. (ii)/(iii);

(v) the number of standard deviations of this ratio from its mean. A binomial distribution is here assumed, and the mean, P, is given by

$$P = \frac{\text{total number of matches summed over all match positions}}{\text{total number of comparisons summed over all match positions}}$$

$$= \frac{\sum_{j=1}^{T} \sum_{s=1}^{S} A_{1sj} \cdot A_{2sj}}{LM/T - ZERO},$$

where

A_{rsj} = no. of occurrences of element numbered s (= 1,2,...S) in jth class (j = 1,2,...T) of chain r (= 1,2),

L = length of chain 1,
M = length of chain 2,
T = sliding step,
ZERO = no. of comparisons involving zero (unknown) elements,

or

$$ZERO = \sum_{j=1}^{T} \left[(MA_{10j} + LA_{20j})/T - A_{10j} A_{20j} \right]$$

The number of standard deviations is then given by the formula:

$$\frac{h-Pc}{\sqrt{P(1-P)c}},$$

where h = number of matches for current match position and c = corresponding number of comparisons.

This formula, however, is not used by the program, which instead calculates

$$\sqrt{c}(2\arcsin\sqrt{h/c} - 2\arcsin\sqrt{P})$$

as the number of standard deviations. This has been done so that the user may be able to obtain a fair approximation to the cumulative probability, i.e. the probability that the ratio of matches to comparisons has the value h/c or less, by treating the standard deviations as a standardized normal variate.

The transformation used is the arcsin transformation (Owen, 1962, p. 293), and we consider that the transformed variate (i.e. the number of standard deviations as calculated in the program) approximates somewhat better to the standardized normal variate than does the untransformed variate. Table 4 gives some values of the normal distribution function.

The number of standard deviations is a useful measure for picking out positions of high (and low) matching. In a simple illustration (Table 2), the "best" match is at the 8th match position. The probability that such a high match would occur in a comparison of random permutations of the two sequences is only about 1/15. Table 4 gives a value slightly under 1/20, a fair approximation.

Table 2.- Sample tabular output.

SLIDING STEP = 1
PROB(MATCH) = .14285714

FORWARD MATCHES

MATCH POS	NO. OF MATCHES	NO. OF COMPS	MATCHES /COMPS	STD DEVS	CHI-SQ UNCORRECTED	CHI-SQ (YATES)
1	0	1	.0000	-.77456	0.1667	1.0417
2	0	2	.0000	-1.0954	0.3333	0.1875
3	0	3	.0000	-1.3416	0.5000	0.0139
4	0	4	.0000	-1.5491	0.6667	0.0104
5	1	5	.2000	.34011	0.1333	0.0750
6	0	6	.0000	-1.8973	1.0000	0.1736
7	2	7	.2857	.93313	1.1667	0.2917
8	3	7	.4286	1.7257	4.6667	2.6250
9	0	6	.0000	-1.8973	1.0000	0.1736
10	2	5	.4000	1.3288	2.7000	1.0083
11	0	4	.0000	-1.5491	0.6667	0.0104
12	0	3	.0000	-1.3416	0.5000	0.0139
13	0	2	.0000	-1.0954	0.3333	0.1875
14	0	1	.0000	-.77456	0.1667	1.0417

SUM CHI-SQ (UNCORRECTED) = 14.0000, WHICH IS
.09535022 STD DEVS FROM THE MEAN (NORMAL APPROX)

REVERSE MATCHES

MATCH POS	NO. OF MATCHES	NO. OF COMPS	MATCHES /COMPS	STD DEVS	CHI-SQ UNCORRECTED	CHI-SQ (YATES)
-1	1	1	1.000	2.3646	6.0000	1.0417
-2	0	2	.0000	-1.0954	0.3333	0.1875
-3	1	3	.3333	.78941	0.8889	0.0139
-4	0	4	.0000	-1.5491	0.6667	0.0104
-5	1	5	.2000	.34011	0.1333	0.0750
-6	0	6	.0000	-1.8973	1.0000	0.1736
-7	0	7	.0000	-2.0493	1.1667	0.2917
-8	0	7	.0000	-2.0493	1.1667	0.2917
-9	1	6	.1667	.16136	0.0278	0.1736
-10	2	5	.4000	1.3288	2.7000	1.0083
-11	1	4	.2500	.54401	0.3750	0.0104
-12	0	3	.0000	-1.3416	0.5000	0.0139
-13	0	2	.0000	-1.0954	0.3333	0.1875
-14	1	1	1.000	2.3646	6.0000	1.0417

SUM CHI-SQ (UNCORRECTED) = 21.2917, WHICH IS
1.3294382 STD DEVS FROM THE MEAN (NORMAL APPROX)

SIMILARITY INDEX S(L) = .57142857

END OF PROGRAM

Note that the untransformed value of the number of standard deviations is given by $\pm\sqrt{\chi^2}$, the sign being that of the excess of the proportion of matches over the mean;

(vi) χ^2, calculated from the 2 x 2 table

O	E
O'	E'

by the formula:

$$\chi^2 = \frac{(O-E)^2}{E} + \frac{(O' - E')^2}{E'},$$

where O = observed (actual) number of matches

O' = observed (actual) number of mismatches

E,E' = corresponding expected values;

(vii) χ^2 with Yates' continuity correction, according to the formula:

$$\chi^2 \text{ (Yates' correction)} = \frac{(|O-E| - \frac{1}{2})^2}{E} + \frac{(|O'-E'| - \frac{1}{2})^2}{E'},$$

using the same notation as in (vi).

The values of χ^2 (each with one degree of freedom) are approximate two-tailed tests for the significance (nonrandomness) of the degree of matching.

The version of the program which includes the plotter procedures will in addition plot a graph of the standard deviations against the match position using an online digital plotter. For convenience the standard deviations are plotted on a cube-root scale. Figure 2 shows the graph corresponding to the output of Table 2.

After all the match positions have been examined, the program will sum the values of χ^2 (uncorrected) and compute the number of standard deviations of this sum from the mean, taking χ^2 with L+M-1 degrees of freedom as the distribution function. In doing so it simply calculates $\sqrt{2\Sigma\chi^2} - \sqrt{2L+2M-3}$ which is the standard normal approximation to χ^2 with L+M-1 degrees of freedom. This function measures the overall nonrandomness of the matching.

Reverse matches

The whole sliding process is then repeated with the second chain reversed in direction, in order to test for subsequences which appear in reverse order in one of the chains (inversions). The printout follows the details given for the forward matches.

Similarity index

The remaining part of the program consists of the computation of a non-probabilistic SIMILARITY INDEX, S_L, between the two chains. It is essentially a measure of the proportion

Table 3.- Sample tabular output of same data with different parameters.

SLIDING STEP = 1
PROB(MATCH) = .14285714

FORWARD MATCHES

MATCH POS	NO. OF MATCHES	NO. OF COMPS	MATCHES /COMPS	STD DEVS
3	0	3	.0000	-1.3416
4	0	4	.0000	-1.5491
6	0	6	.0000	-1.8973
8	3	7	.4286	1.7257
9	0	6	.0000	-1.8973
10	2	5	.4000	1.3288
11	0	4	.0000	-1.5491
12	0	3	.0000	-1.3416

END OF PROGRAM

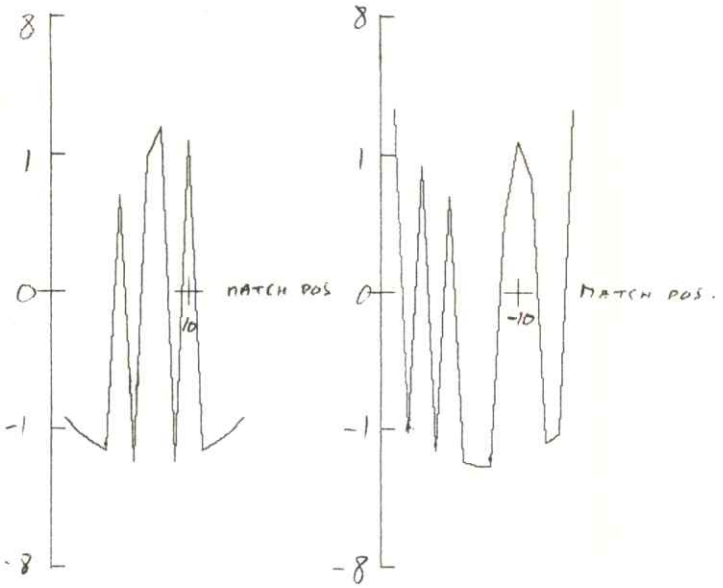


Figure 2.- Sample graphical output.

of the two chains which can be paired off as matching subsequences. We give here a full definition of S_L.

Denote the chains by

$A_1 \quad A_2 \quad \ldots\ldots A_r \ldots\ldots A_L$

and $B_1 \quad B_2 \quad \ldots\ldots B_s \ldots\ldots B_M$

where $L > 2$, $M > 2$, and A_r, B_s represent typical chain elements.

Suppose for some r,s that

$A_{r-1} \neq B_{s-1}$ (or $r = 1$ or $s = 1$),

$A_{r+i} = B_{s+i} \neq 0 \quad (i = 0,1,2,\ldots,n_{r,s}-1)$

$A_{r+n_{r,s}} \neq B_{s+n_{r,s}}$ (or $r+n_{r,s} = L$ or $s+n_{r,s} = M$),

i.e. the two subchains of length $n_{r,s}$, one in each chain, match perfectly.

Then S_L (L stands for "link") is defined by the relation

$$S_L = \frac{\sum_{r=1}^{L} \sum_{s=1}^{M} (n_{r,s}-1)}{\max (L,M) - 1}$$

subject to the conditions that

(i) $n_{r,s} \geqslant 2$;

(ii) Once any element has been matched in a subchain it cannot be matched again. Thus, once an element has been paired off it is "deleted". Cf. example (iii) below;

(iii) If AREV $\neq 0$ on the parameter tape then the summation will cover the cases of the second chain being reversed as well as in normal sequence.

Examples:

(i) A B C D E
A B C D E $\quad S_L = \frac{4}{4} = 1$

(ii) A B C D E
E D C B A $\quad S_L = 1$

(iii) A B C D E F G
A B C A B $\quad S_L = \frac{1}{3}$

(iv) A X B X C X D
A Y B Y C Y D $\quad S_L = 0.$

Table 4.- Extreme values of the normal distribution.*

Probabilities of falling outside (one tailed)	Standard deviations	Cube roots
1/2.5	0.25	0.63
1/5	0.84	0.94
1/10	1.28	1.09
1/20	1.65	1.18
1/40	1.96	1.25
1/100	2.33	1.32
1/200	2.58	1.37
1/1000	3.09	1.46
$1/10^4$	3.72	1.55
$1/10^5$	4.26	1.62
$1/10^6$	4.75	1.68
$1/10^7$	5.20	1.73
$1/10^8$	5.61	1.78
$1/10^9$	6.00	1.82
$1/10^{10}$	6.29	1.85
$1/10^{20}$	9.26	2.10
$1/10^{30}$	11.47	2.25
$1/10^{40}$	13.13	2.36
$1/10^{50}$	14.93	2.46
$1/10^{100}$	21.15	2.77
$1/10^{200}$	30.10	3.11
$1/10^{1000}$	67.61	4.07

*/ Table gives probabilities for extreme values of normal distribution. For present application these are acceptably close to values for binomial distribution under arcsin transformation; however, they should not be taken too literally, since assumptions of normality are likely to be only roughly satisfied. Main purpose of table is to indicate comparative significance; for example 10 standard deviations from mean are many times more significant than 6 standard deviations. Cube root column will assist in examination of graphical output.

In these examples reversed pairings are counted. Otherwise the respective values of S_L are 1,0, $\frac{1}{3}$ and 0.

The subtraction of unity from each $n_{r,s}$ (and hence also from max (L,M)) in the definition has been done in order to decrease the value of S_L for each break in sequence. Thus for the pair:

A B C D E
C D E A B

$S_L = \frac{3}{4}$ and not 1, which would otherwise by the case.

The similarity index would normally be of little value for data in which the sliding step is greater than unity, because its definition remains unchanged.

Unknown comparisons

Suppose the two sequences:

1 0 0 1 2 3 0 2
0 1 0 1 2 0 3 4 5

overlap as written. The program will then count two matches and only three (not eight) comparisons, since a comparison in which either element (or both elements) is zero ("unknown") is not counted. In the (rare) instance in which not a single valid comparison is found for a given overlap position, the values of functions (iv), (v), (vi) and (vii) are all indeterminate, and the program does not punch out any values for them. Consequently, for each overlap position there will be one degree of freedom fewer in $\Sigma \chi^2$ than in the usual case, i.e. in which each overlap position produces at least one valid comparison.

In the graphical output any indeterminate value of the ordinate is assigned the value zero.

The parameter tape

This tape, which is read in before the two sequences, contains twelve numbers of which all but the first two must be integers.

The numbers will be denoted by λ, μ, AMATCH, ACOMPS, APROP, ADEVS, ACHISQ, ACHSQY, AGRAPH, AREV, ASIM, and T respectively.

If for any match position the number of standard deviations of the proportion of matches from the mean (item (v) above) lies between λ and μ (λ should not exceed μ) then the complete row of tabular output corresponding to this match position will not be punched. Reverse matches are treated in the same way.

Zero values of AMATCH, ACOMPS, APROP, ADEVS, ACHISQ, ACHSQY will cause punching of columns (ii)-(vii) respectively to be suppressed. If all six values are zero then no tabular output will appear. Zero values of ACHISQ will also cause suppression of punching of the corresponding cumulative function. Reverse matches are treated similarly.

If AGRAPH, AREV, or ASIM are zero then no graphical output or reversed sliding process or computation of the similarity index, respectively, will take place.

Note: in the version of the program in which dummy procedures have replaced the plotter procedures a value for AGRAPH must still be given. Any integer or zero will do.

Note further that AREV = 0 will suppress that part of the computation of the similarity index which involves reversed pairings, as described above.

Finally T is the required value of the sliding step. In the illustrative example just given, T was unity, but the use of higher values has been described earlier and one such value will be used in the geological example to follow.

In the output shown in Table 2 and Figure 2 the parameter tape consisted of the numbers:

-100 -100 1 1 1 1 1 1 1 1 1 1

In contrast, the parameters

-1.2 1.2 1 1 1 1 0 0 0 0 0 1

produce the output shown in Table 3 if the same short chain-pair is used as data.

Data checks

The program checks whether $\lambda \leqslant \mu$ and $T \geqslant 1$. Any errors will cause a fault indication to be printed and an immediate termination of the program.

There are no other data checks. Thus the user must take care not to omit any elements in the chain-pair or to insert spurious ones. It is recommended that the data are punched in rows of fixed length (e.g. in rows of ten) with the numbers forming neat columns.

OPERATION OF THE PROGRAM

The program calls in succession for three paper tapes to be read - the parameter tape, the tape containing the first chain, and that containing the second chain. As written in Elliott ALGOL the program contains a WAIT between these READ statements.

An error on the parameter tape will cause one of the messages:

ERROR IN BOUNDS OF STD DEVS or
NON-POSITIVE SLIDING-STEP

to be punched, followed by immediate termination of the run.

Output is on the paper tape punch and/or the graph plotter depending on certain values on the parameter tape and the version of the program used. If used, the plotter should be set initially with the pen near the left hand edge of the paper.

Inadmissible characters on the input tapes cause the message:

READ ERROR

to be displayed, followed by a WAIT. Re-entry to the program will be at the READ statement at which the error occurred.

Computer storage considerations

The compiled program takes up about 3000 40-bit machine locations in the Elliott 803. Each location contains two single-address instructions. This space includes all the necessary

standard procedures and most variables, but it does not include space for the chain-pair, which is thus stored elsewhere in the machine.

Timing estimates

On the Elliott 803C, average instruction time about 4 μ sec, the time taken for a run is very nearly the same as the punching time alone for most options. The punch operates at about 100 characters per second at full speed. The similarity index, however, takes about a minute to compute for sequences each of about 100 elements in length. This time increases approximately with the square of the mean of the sequence lengths.

On the Elliott 803B, which is the machine used at Leicester, the run-times are determined mainly by the amount of internal computing. The machine instructions take on average about 400 μ sec. to be executed. The forward and reverse sliding processes with unit sliding step and chain-lengths each of 100 elements in length take about an hour. This time is virtually independent of the degree of row and column suppression, since most of the time is taken up in counting up the matches.

Using an Elliott 803B the plotting-time averages about 2 1/2 seconds per comparison.

Machine adaptability

It is hoped that the note on Elliott ALGOL above, together with the use of identifiers that are often mnemonic in form, will render the adaptation of the program to another machine with an ALGOL compiler a reasonably practical proposition*.

The authors know of at least two preprocessors which will convert programs from Elliott ALGOL to other (British) versions of ALGOL 60.

SAMPLE PROBLEMS

Two examples using the cross-association program for processing certain types of non-numeric sequences of geologic data are described here. Various other applications are presently being tested. It should be emphasized that the results presented in this report are preliminary.

Only a very brief discussion of the Kansas section of Upper Pennsylvanian and Lower Permian rock units and their cyclic nature is included here (Fig. 3). For further information the reader is referred to publications by Moore (1935, 1949, 1957) and Merriam (1963).

Stratigraphic summary

Pennsylvanian deposits in Kansas are divided into five stages (in ascending order): Morrowan, Atokan, Desmoinesian, Missourian, and Virgilian. The lower contact with Mississippian rocks is unconformable, and at the upper contact Pennsylvanian rocks are gradational with Permian rocks. Deposits of Pennsylvanian age cover all of Kansas except the extreme southeastern corner.

---

* A FORTRAN IV version of the program is included in the Appendix.

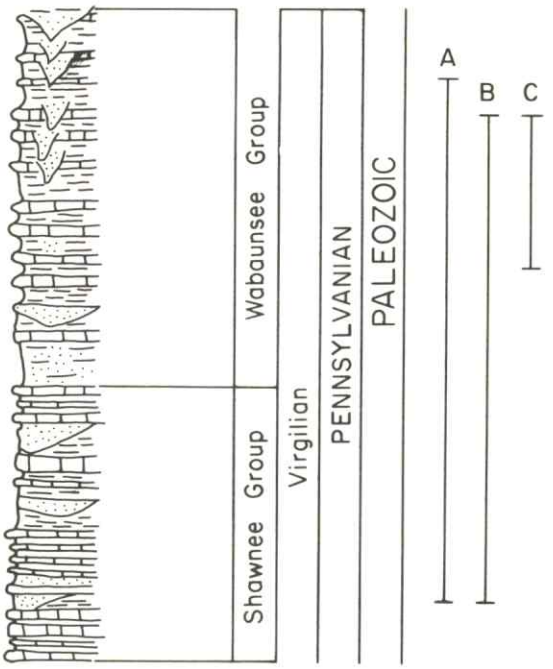


Figure 3.- Generalized section of Upper Pennsylvanian rocks in Kansas showing stratigraphic position of three composite sections studied (A, Kansas River Valley; B, southern Kansas, Chautauqua and Cowley Counties; C, Osage County, northern Oklahoma).

In eastern Kansas Pennsylvanian beds dip gently westward, forming cuestas with very gentle dip slopes to the west and relatively steep east faces, and form a wide outcrop band from north to south. Pennsylvanian deposits in Kansas are a succession of thin, laterally persistent, cyclic beds, which could be and have been described as monotonous.

Although much has been written on the origin and development of cyclothems, they are little understood. Several explanations of their formation have been suggested; generally these include eustatic changes of sea level, tectonic movements, complex environmental changes, or a combination of factors.

Cyclothems, for the most part consisting of marine limestone and shale, alternating with nonmarine clastic deposits, may be symmetrical or asymmetrical, depending on the arrangement of the marine and nonmarine components. A nonmarine to marine to nonmarine arrangement is symmetrical and gives rise to units of a cyclic nature, whereas nonmarine to marine followed by nonmarine to marine is asymmetrical or hemicyclic. Inasmuch as the transgressive phase of the cyclothem is usually better represented (or better preserved) than the regressive part, most cyclothems exhibit asymmetrical aspects. Seldom are all members of a cyclothem represented at a single locality; either they were not developed or they were developed and subsequently destroyed.

Lower Permian rocks in Kansas belong to the Wolfcampian Stage (in ascending order): Admire, Council Grove, and Chase Group. Because they are similar in character, it is convenient to treat them together. The overall thickness of the three groups is about 800 feet.

The units consist chiefly of shale but include some thin limestones; the sequence is similar to that of underlying Pennsylvanian rocks. In the upper part the shales are mostly red and green or varicolored, and most of the limestones are cherty. Of special interest is the random distribution of sandstone bodies in various stratigraphic positions, indicating minor local unconformities.

Sample problem 1

One set of data comprises three composite surface sections located in the Kansas River Valley, Cowley and Chautauqua Counties in southern Kansas, and Osage County in northern Oklahoma (Fig. 4). From the lithologic descriptions, each unit was assigned to a category based on Moore's ideal cyclothem (Table 5). Some descriptions were inadequate, and undoubtedly the sections were measured with different degrees of detail and accuracy. Unfortunately also the three sections are of unequal stratigraphic length. These factors account for some discrepancies in the results, but at this point they are unaccessed.

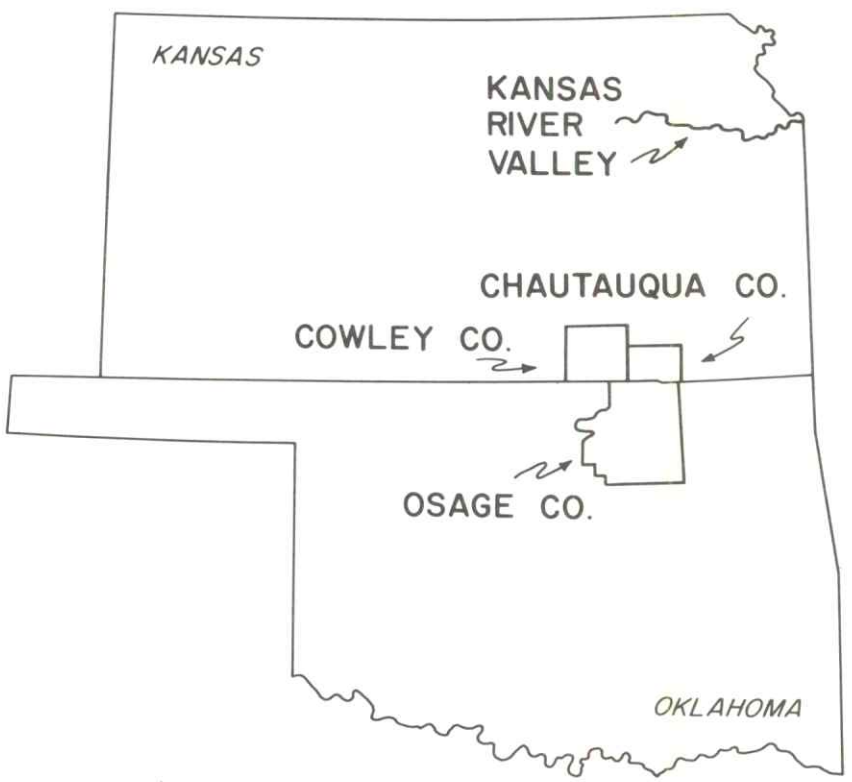


Figure 4.- Index map showing location of composite surface sections.

Table 5.- Members of an ideal cyclothem, designated upwards (from Moore, 1935).

.9	Shale (and coal).
.8	Shale, typically with molluscan fauna.
.7	Limestone, algal, molluscan, or with mixed molluscan and molluscoid fauna.
.6	Shale, molluscoids dominant.
.5	Limestone, contains fusulinids, associated commonly with molluscoids.
.4	Shale, molluscoids dominant.
.3	Limestone, molluscan, or with mixed molluscan and molluscoid fauna.
.2	Shale, typically with molluscan fauna.
.1c	Coal.
.1b	Underclay.
.1a	Shale, may contain land-plant fossils.
.0	Sandstone.

Stratigraphically the Kansas River Valley section extends from the Jim Creek Limestone (top) to the Kanwaka Shale (bottom) in which 106 successively different rock lithologies were recognized. The section in southern Kansas extends from the Dover Limestone (top) to Kanwaka Shale (bottom) and 86 units were recognized; the northern Oklahoma section has 36 different units in a section extending from the Dover Limestone down into the Cedar Vale Shale. The coded information showing the successive categories as interpreted from the raw data for the three sections are shown in Table 6, and the key to the code is shown in Table 7.

Table 6.- Coded data for first sample problem involving three composite surface sections (location shown in Figure 4).

Northern Oklahoma	36 7 6 1 9 6 8 6 9 6 8 6 1 6 7 6 7 6 7 6 7 6 7 6 7 6 8 6 7 6 9 6 7 6 5 4 6
Southern Kansas	86 7 6 5 6 9 6 7 6 8 3 9 6 8 6 7 1 9 7 6 2 1 7 6 9 8 2 8 6 5 2 1 2 9 8 6 8 6 9 6 3 1 2 6 9 6 9 3 2 4 6 9 2 1 2 1 2 9 6 8 7 8 6 9 6 8 7 6 9 6 8 6 9 2 1 2 9 6 9 6 8 6 7 6 3 1 2
Kansas River Valley	106 8 2 9 2 6 7 6 8 6 9 6 8 2 1 8 2 8 2 8 6 9 2 6 7 8 6 7 6 7 2 1 2 7 2 5 2 1 6 7 3 7 6 7 6 3 2 7 6 8 6 2 7 6 7 6 5 6 2 8 6 7 6 7 2 7 6 9 6 8 3 7 6 5 2 1 7 6 8 6 8 6 9 7 8 2 1 2 8 6 7 6 8 6 9 6 8 6 5 1 2 9 6 8 6 7 2

Table 7.- Key to lithologic code used in compiling data for Table 6.

9	algal limestone
8	fusulinid limestone
7	molluscan limestone
6	molluscan shale
5	coal
4	underclay
3	shale with land plants
2	siltstone
1	sandstone

Results of matching the three sections together in pairs are shown in Table 8. Overall, the best match is between the sections in southern Kansas and northern Oklahoma; this is to be expected as they are close together geographically and are lithologically similar. The Kansas River Valley section shows similarity to the one in southern Kansas but is less like the one in northern Oklahoma. The "best" match between any two sections, however, does not necessarily correspond with the "best" geological correlation and indeed in this example does not. It should be noted that good matches were obtained at several positions emphasizing, of course, the cyclic nature of these sequences. Of geological interest is the fact that good matches were obtained where older units (Shawnee Group) of the north were compared with younger units (Wabaunsee Group) of the south. The reverse matches indicate some symmetry in the megacycles of the Shawnee Group.

Sample problem 2

The second example involves two shorter more detailed portions of the stratigraphic sections located in southern Kansas and northern Oklahoma (Fig. 4) in the Wabaunsee Group (Fig. 3) and extending from the Dry Shale (top) down into the Auburn Shale (bottom). Information regarding the lithology and fossil content of each bed was compiled from the written description so that more than one property of each unit could be used in comparing one sequence to another for each position of primary data. In this way comparisons are based on the raw data rather than the subjective interpretation of the investigator as described in the first example. There are objections obviously to both methods, and it is for the investigator to decide which one best suits his needs.

The data for this example is shown in Table 9. If a particular characteristic was present in any unit, it was so indicated by a 2 or if absent by a 1; thus, it was possible to note the lithology and any of nine parameters for each unit. The matching of the two sequences in this example then is for every thirteenth position.

Again as in the first example the "best" match is not the "best" geological match but is very close (Table 10). It would seem that this technique has merit for correlating stratigraphic sequences or for suggesting a possible position or positions for "best" correlation. Another merit is that it draws critical attention toward the quantity of exact information that is needed for making confident statements about geological matches between rock sequences. Additional work, however, needs to be done.

Table 8.- Output for matching three stratigraphic sections together in pairs. Forward and reverse matches are shown along with statistical information.

Northern Oklahoma - southern Kansas

SLIDING STEP = 1
PROB(MATCH) = .19186046

FORWARD MATCHES

MATCH POS	NO. OF MATCHES	NO. OF COMPS	MATCHES /COMPS	STD DEVS	CHI-SQ UNCORRECTED	CHI-SQ (YATES)
3	0	3	.0000	-1.5695	0.7122	0.0123
6	0	6	.0000	-2.2196	1.4245	0.4558
12	5	12	.4167	1.7201	3.9113	2.5958
16	6	16	.3750	1.6453	3.4611	2.3607
17	1	17	.0588	-1.7186	1.9405	1.1774
18	7	18	.3889	1.8664	4.5067	3.3255
19	1	19	.0526	-1.9346	2.3754	1.5623
22	10	22	.4545	2.6675	9.7509	8.1700
23	0	23	.0000	-4.3456	5.4604	4.2931
24	10	24	.4167	2.4326	7.8227	6.4400
25	0	25	.0000	-4.5306	5.9353	4.7623
26	11	26	.4231	2.5962	8.9648	7.5355
29	2	29	.0690	-2.0212	2.8248	2.0876
33	3	36	.0833	-1.9266	2.7347	2.0795
61	3	36	.0833	-1.9266	2.7347	2.0795
63	2	36	.0556	-2.5654	4.3137	3.4794
71	2	36	.0556	-2.5654	4.3137	3.4794
72	11	36	.3056	1.5675	3.0013	2.3126
73	3	36	.0833	-1.9266	2.7347	2.0795
75	1	36	.0278	-3.4313	6.2511	5.2376
81	3	36	.0833	-1.9266	2.7347	2.0795
90	3	32	.0938	-1.8665	1.9066	1.4042
96	1	24	.0417	-2.4262	3.4516	2.5903
99	12	23	.5217	3.3931	16.1423	14.0646
101	6	21	.3810	1.9412	4.6426	3.7000
102	0	20	.0000	-4.0524	4.7462	3.5914
103	11	19	.5789	3.5655	16.3611	15.5494
104	0	18	.0000	-3.8445	4.2734	3.1255
105	8	17	.4706	2.4951	8.5160	6.8152
106	0	16	.0000	-3.6246	3.7506	2.6619
107	7	15	.4667	2.3133	7.3059	5.6410
109	5	13	.3846	1.5545	3.1152	1.9960
111	6	11	.5455	2.5042	6.6701	6.7362
112	0	10	.0000	-2.8655	2.3741	1.2979
113	4	9	.4444	1.6560	3.7032	2.2533
117	3	5	.6000	1.9350	5.3717	3.0619
118	0	3	.0000	-1.5695	0.7122	0.0123

SUM CHI-SQ (UNCORRECTED) = 241.5279, WHICH IS
6.4543576 STD DEVS FROM THE MEAN (NORMAL APPROX)

REVERSE MATCHES

MATCH POS	NO. OF MATCHES	NO. OF COMPS	MATCHES /COMPS	STD DEVS	CHI-SQ UNCORRECTED	CHI-SQ (YATES)
-1	1	1	1.000	2.2330	4.2121	0.6124
-4	0	4	.0000	-1.8123	0.9496	0.1153
-11	0	11	.0000	-3.0054	2.6115	1.5207
-12	5	12	.4167	1.7201	3.9113	2.5958
-14	7	14	.5000	2.4045	6.5734	6.7012
-16	6	16	.3750	1.6453	3.4611	2.3607
-17	0	17	.0000	-3.7362	4.0360	2.8534
-18	12	18	.6667	4.2590	26.1716	23.1991
-19	0	19	.0000	-3.9498	4.5106	3.3592
-20	10	20	.5000	2.5695	12.2477	10.3405
-21	0	21	.0000	-4.1525	4.9656	3.6250
-22	10	22	.4545	2.6675	9.7509	8.1700

-24	10	24	.4167	2.4326	7.6227	6.4400
-25	1	25	.0400	-2.5204	3.7184	2.8035
-26	5	26	.3214	1.5611	3.0317	2.2536
-32	11	32	.3437	1.9583	4.7614	3.6322
-34	12	34	.3529	2.1311	5.6898	4.6903
-36	11	36	.3056	1.5875	3.0013	2.3126
-40	1	36	.0278	-3.4313	6.2511	5.2376
-45	11	36	.3056	1.5875	3.0013	2.3126
-50	2	36	.0556	-2.5854	4.3137	3.4794
-52	3	36	.0833	-1.9266	2.7347	2.0795
-56	2	36	.0556	-2.5854	4.3137	3.4794
-65	11	36	.3056	1.5875	3.0013	2.3126
-66	3	34	.0882	-1.7703	2.3547	1.7330
-94	2	28	.0714	-1.9350	2.6192	1.9001
-95	9	27	.3333	1.6844	3.4853	2.6326
-96	2	26	.0769	-1.7577	2.2153	1.5360
-97	11	25	.4400	2.7186	9.9280	6.3921
-98	0	24	.0000	-4.4392	5.6978	4.5276
-99	9	23	.3913	2.1335	5.9006	4.6844
-100	0	22	.0000	-4.2502	5.2230	4.0589
-101	8	21	.3810	1.9412	4.8428	3.7000
-102	1	20	.0500	-2.0382	2.5959	1.7615
-104	1	18	.0556	-1.8262	2.1569	1.3673
-105	7	17	.4118	2.0063	5.3021	3.9786
-106	1	16	.0625	-1.6057	1.7268	0.9933
-108	0	14	.0000	-3.3905	3.3237	2.2015
-109	5	13	.3846	1.5545	3.1152	1.9960
-110	0	12	.0000	-3.1390	2.6489	1.7459
-117	0	5	.0000	-2.0262	1.1871	0.2721
-119	0	3	.0000	-1.5695	0.7122	0.0123

SUM CHI-SQ (UNCORRECTED) = 259.2734, WHICH IS
7.2474478 STD DEVS FROM THE MEAN (NORMAL APPROX)

Northern Oklahoma - Kansas River Valley

SLIDING STEP = 1
PROB(MATCH) = .20597464

FORWARD MATCHES

MATCH POS	NO. OF MATCHES	NO. OF COMPS	MATCHES /COMPS	STD DEVS	CHI-SQ UNCORRECTED	CHI-SQ (YATES)
3	0	3	.0000	-1.6308	0.7782	0.0283
5	0	5	.0000	-2.1053	1.2970	0.3433
8	0	8	.0000	-2.6630	2.0752	1.0069
9	4	9	.4444	1.5519	3.1294	1.8411
12	0	12	.0000	-3.2615	3.1129	1.9606
14	1	14	.0714	-1.5006	1.5496	0.8361
32	12	32	.3750	2.1268	5.5699	4.6042
33	3	33	.0909	-1.8934	2.6715	2.0143
34	11	34	.3235	1.5623	2.8728	2.1990
35	3	35	.0857	-2.0562	3.0950	2.4034
36	13	36	.3611	2.0830	5.2976	4.3915
37	2	36	.0556	-2.7976	4.9804	4.1031
39	2	36	.0556	-2.7976	4.9804	4.1031
40	12	36	.3333	1.7329	3.5703	2.6341
41	2	36	.0556	-2.7976	4.9804	4.1031
42	12	36	.3333	1.7329	3.5703	2.6341
63	14	36	.3889	2.4273	7.3646	6.2886
67	13	36	.3611	2.0830	5.2976	4.3915
73	12	36	.3333	1.7329	3.5703	2.6341
83	12	36	.3333	1.7329	3.5703	2.6341
84	4	36	.1111	-1.5749	1.9809	1.4433
85	12	36	.3333	1.7329	3.5703	2.6341
86	2	36	.0556	-2.7976	4.9804	4.1031
87	13	36	.4167	2.7672	9.7712	8.5254
88	2	36	.0556	-2.7976	4.9804	4.1031
89	12	36	.3333	1.7329	3.5703	2.6341
90	1	36	.0278	-3.6435	6.9696	5.9429
91	13	36	.3611	2.0830	5.2976	4.3915
92	2	36	.0556	-2.7976	4.9804	4.1031
93	12	36	.3333	1.7329	3.5703	2.6341

94	2	36	.0556	-2.7976	4.9804	4.1031
95	15	36	.3611	2.0830	5.2976	4.3915
96	2	36	.0556	-2.7976	4.9804	4.1031
97	12	36	.3333	1.7329	3.5703	2.8341
98	3	36	.0833	-2.1368	3.3108	2.6034
102	4	36	.1111	-1.5749	1.9809	1.4433
113	3	29	.1034	-1.5468	1.8639	1.2697
117	2	25	.0800	-1.6432	2.4258	1.7186
119	2	23	.0870	-1.6473	1.9921	1.3306
121	0	21	.0000	-4.3146	5.4475	4.2689
123	0	19	.0000	-4.1040	4.9287	3.7490
125	0	17	.0000	-3.8820	4.4099	3.2404
127	0	15	.0000	-3.6465	3.8911	2.7336
129	0	13	.0000	-3.3947	3.3723	2.2305
136	0	6	.0000	-2.3062	1.5564	0.9516
138	0	4	.0000	-1.8830	1.0376	0.1604
139	0	3	.0000	-1.6308	0.7762	0.0283

SUM CHI-SQ (UNCORRECTED) = 241.2016, WHICH IS
5.2000229 STD DEVS FROM THE MEAN (NORMAL APPROX)

REVERSE MATCHES

MATCH POS	NO. OF MATCHES	NO. OF COMPS	MATCHES /COMPS	STD DEVS	CHI-SQ UNCORRECTED	CHI-SQ (YATES)
-3	0	3	.0000	-1.6308	0.7762	0.0283
-4	0	4	.0000	-1.8830	1.0376	0.1604
-5	0	5	.0000	-2.1053	1.2970	0.3433
-6	4	6	.6667	2.3723	7.7862	5.2241
-7	0	7	.0000	-2.4910	1.8158	0.7746
-14	0	14	.0000	-3.5228	3.8317	2.4815
-16	1	16	.0625	-1.7472	2.0130	1.2321
-17	7	17	.4118	1.8605	4.4020	3.2330
-18	0	18	.0000	-3.9945	4.6693	3.4948
-20	1	20	.0500	-2.1964	2.9750	2.0970
-22	1	22	.0455	-2.4036	3.4660	2.5540
-30	3	30	.1000	-1.6358	2.0601	1.4630
-36	3	36	.0833	-2.1368	3.3108	2.6034
-38	3	36	.0833	-2.1368	3.3108	2.6034
-45	3	36	.0833	-2.1368	3.3108	2.6034
-47	3	36	.0833	-2.1368	3.3108	2.6034
-49	2	36	.0556	-2.7976	4.9804	4.1031
-51	3	36	.0833	-2.1368	3.3108	2.6034
-53	1	36	.0278	-3.6435	6.9696	5.9425
-54	14	36	.3889	2.4273	7.3646	6.2886
-55	3	36	.0833	-2.1368	3.3108	2.6034
-56	13	36	.3611	2.0830	5.2976	4.3915
-57	1	36	.0278	-3.6435	6.9696	5.9425
-58	14	36	.3889	2.4273	7.3646	6.2886
-59	2	36	.0556	-2.7976	4.9804	4.1031
-60	12	36	.3333	1.7329	3.5703	2.8341
-70	12	36	.3333	1.7329	3.5703	2.8341
-74	14	36	.3889	2.4273	7.3646	6.2886
-78	13	36	.3611	2.0830	5.2976	4.3915
-91	4	36	.1111	-1.5749	1.9809	1.4433
-97	12	36	.3333	1.7329	3.5703	2.8341
-100	3	36	.0833	-2.1368	3.3108	2.6034
-101	13	36	.3611	2.0830	5.2976	4.3915
-102	3	36	.0833	-2.1368	3.3108	2.6034
-103	13	36	.3611	2.0830	5.2976	4.3915
-104	3	36	.0833	-2.1368	3.3108	2.6034
-105	13	36	.3611	2.0830	5.2976	4.3915
-106	3	36	.0833	-2.1368	3.3108	2.6034
-133	0	9	.0000	-2.8246	2.3347	1.2451
-135	0	7	.0000	-2.4910	1.8158	0.7746
-138	0	4	.0000	-1.8830	1.0376	0.1604

SUM CHI-SQ (UNCORRECTED) = 241.7985, WHICH IS
5.2277546 STD DEVS FROM THE MEAN (NORMAL APPROX)

SLIDING STEP = 1
PROB(MATCH) = .16619131

FORWARD MATCHES

MATCH POS	NO. OF MATCHES	NO. OF COMPS	MATCHES /COMPS	STD DEVS	CHI-SQ UNCORRECTED	CHI-SQ (YATES)
5	0	5	.0000	-1.8764	0.9966	0.1581
7	0	7	.0000	-2.2202	1.3952	0.4536
9	0	9	.0000	-2.5175	1.7938	0.7950
17	6	17	.3529	1.7831	4.2785	3.0370
18	1	18	.0556	-1.5439	1.5900	0.8918
19	6	19	.3158	1.5417	3.0685	2.0839
20	1	20	.0500	-1.7386	1.9485	1.2002
22	1	22	.0455	-1.9235	2.3143	1.5251
23	11	23	.4783	3.2972	16.1643	13.9907
24	1	24	.0417	-2.1000	2.6856	1.8622
40	3	40	.0750	-1.8024	2.4004	1.7875
41	11	41	.2683	1.5954	3.0844	2.3916
48	16	48	.3333	2.7101	9.6769	8.5083
49	4	49	.0816	-1.8220	2.5284	1.9550
50	13	50	.2600	1.6288	3.1753	2.5344
52	16	52	.3077	2.4245	7.5136	6.5271
62	19	62	.3065	2.6262	8.8021	7.8190
65	5	65	.0769	-2.2391	3.7379	3.1215
78	6	78	.0769	-2.4528	4.4855	3.8645
89	9	86	.1047	-1.6780	2.3504	1.9273
90	20	86	.2326	1.5453	2.7335	2.2756
92	20	86	.2326	1.5453	2.7335	2.2756
93	5	86	.0581	-3.2713	7.2458	6.4870
95	6	86	.0698	-2.8301	5.7702	5.0954
96	25	86	.2907	2.7737	9.6207	8.7432
97	8	86	.0930	-2.0395	3.3225	2.8155
99	8	86	.0930	-2.0395	3.3225	2.8155
110	6	82	.0732	-2.6439	5.1203	4.4710
112	8	80	.1000	-1.7557	2.5294	2.0743
119	6	73	.0822	-2.2065	3.7171	3.1356
123	17	69	.2464	1.6532	3.2016	2.6491
131	5	61	.0820	-2.0234	3.1227	2.5445
137	5	55	.0909	-1.6852	2.2494	1.7390
145	13	47	.2766	1.8360	4.1342	3.3759
147	13	45	.2889	1.9796	4.8889	4.0435
159	11	33	.3333	2.2471	6.6529	5.5014
161	9	31	.2903	1.6607	3.4471	2.6095
166	9	26	.3462	2.1326	6.0766	4.8473
173	0	19	.0000	-3.6578	3.7870	2.6826
178	5	14	.3571	1.6510	3.6838	2.4347
183	0	9	.0000	-2.5175	1.7938	0.7950
186	0	6	.0000	-2.0555	1.1959	0.2973
187	0	5	.0000	-1.8764	0.9966	0.1581

SUM CHI-SQ (UNCORRECTED) = 280.6202, WHICH IS
4.1712924 STD DEVS FROM THE MEAN (NORMAL APPROX)

REVERSE MATCHES

MATCH POS	NO. OF MATCHES	NO. OF COMPS	MATCHES /COMPS	STD DEVS	CHI-SQ UNCORRECTED	CHI-SQ (YATES)
-4	0	4	.0000	-1.6783	0.7973	0.0490
-5	0	5	.0000	-1.8764	0.9966	0.1581

-11	0	11	.0000	-2.7832	2.1925	1.1572
-12	5	12	.4167	1.9522	5.4330	3.7758
-15	0	15	.0000	-3.2501	2.9897	1.9107
-17	0	17	.0000	-3.4599	3.3884	2.2952
-19	1	19	.0526	-1.6426	1.7682	1.0436
-27	10	27	.3704	2.4357	8.1229	6.7163
-28	1	28	.0357	-2.4317	3.4399	2.5628
-30	2	30	.0667	-1.7389	2.1444	1.4863
-33	10	33	.3030	1.8732	4.4592	3.5264
-40	1	40	.0250	-3.3029	5.7544	4.7806
-41	13	41	.3171	2.2824	6.7357	5.6909
-42	3	42	.0714	-1.9358	2.7218	2.0809
-43	12	43	.2791	1.7924	3.9538	3.1812
-45	13	45	.2889	1.9796	4.8889	4.0435
-46	4	46	.0870	-1.6354	2.0841	1.5515
-47	14	47	.2979	2.1584	5.8812	4.9694
-48	4	48	.0833	-1.7605	2.3781	1.8178
-49	13	49	.2653	1.6968	3.4737	2.7953
-52	4	52	.0769	-2.0027	2.9904	2.3808
-53	14	53	.2642	1.7457	3.6702	2.9974
-71	18	71	.2535	1.8160	3.9076	3.3028
-74	4	74	.0541	-3.1872	6.7152	5.9303
-75	19	75	.2533	1.8627	4.1100	3.5052
-76	7	76	.0921	-1.9448	3.0103	2.4994
-77	24	77	.3117	3.0261	11.7632	10.7366
-78	7	78	.0897	-2.0428	3.2896	2.7611
-95	8	86	.0930	-2.0395	3.3225	2.8155
-97	7	86	.0814	-2.4219	4.4625	3.8715
-101	9	86	.1047	-1.6780	2.3504	1.9273
-103	6	86	.0698	-2.8301	5.7702	5.0954
-104	28	86	.3256	3.4746	15.7669	14.6376
-107	8	85	.0941	-1.9929	3.1864	2.6875
-109	8	83	.0964	-1.8990	2.9187	2.4367
-112	8	80	.1000	-1.7557	2.5294	2.0743
-129	16	63	.2540	1.7188	3.5029	2.8981
-132	6	60	.1000	-1.5205	1.8978	1.4494
-134	5	58	.0862	-1.8566	2.6777	2.1316
-138	21	54	.3889	3.7250	19.3264	17.7527
-139	5	53	.0943	-1.5682	1.9746	1.4901
-141	3	51	.0588	-2.4983	4.2427	3.5033
-142	14	50	.2800	1.9474	4.6735	3.8883
-143	2	49	.0408	-3.0305	5.5583	4.6904
-144	13	48	.2708	1.7659	3.7930	3.0754
-145	2	47	.0426	-2.9084	5.1848	4.3309
-157	12	35	.3429	2.4333	7.8831	6.6598
-158	2	34	.0588	-2.0399	2.8285	2.1067
-159	15	33	.4545	3.6763	19.8012	17.7750
-160	2	32	.0625	-1.8918	2.4829	1.7910
-161	11	31	.3548	2.4300	7.9614	6.6582
-162	0	30	.0000	-4.5963	5.9795	4.8403
-164	2	28	.0714	-1.5805	1.8145	1.1951
-166	0	26	.0000	-4.2789	5.1822	4.0523
-171	7	21	.3333	1.7926	4.2337	3.1134
-172	7	20	.3500	1.9065	4.8763	3.6400
-181	0	11	.0000	-2.7832	2.1925	1.1572
-187	0	5	.0000	-1.8764	0.9966	0.1581
-188	0	4	.0000	-1.6783	0.7973	0.0490
-191	1	1	1.000	2.3000	5.0172	0.8041

SUM CHI-SQ (UNCORRECTED) = 350.6273, WHICH IS
6.9619903 STD DEVS FROM THE MEAN (NORMAL APPROX)

Table 9.- Data used for input for second sample problem using multiple descriptors for each unit (T=13); 2=present; 1=absent. Columns from left to right - limestone, shale, sandstone, coal, calcareous, argillaceous, sandy, thick-bedded, thin-bedded, fusulines, algae, mollusks, mixed fauna.

```
-100  -100
 1  1  1  1  1  1  0  1  0  13


247
 1  1  2  1  1  2  1  1  2  1  1  1  1
 2  1  1  1  1  1  1  2  1  1  1  1  2
 1  2  1  1  2  1  2  1  1  1  1  1  2
 2  1  1  1  1  1  1  1  2  1  2  1  2

 1  2  1  1  1  1  1  1  1  1  1  1  1
 2  1  1  1  1  1  1  1  1  2  2  1  2
 1  2  1  1  1  1  1  1  1  1  1  1  1
 2  1  1  1  1  1  1  1  2  1  2  1  2

 1  2  1  1  1  1  1  1  1  1  1  1  1
 2  1  1  1  1  1  1  1 2    2  2  1  2
 1  2  1  1  1  1  1  1  1  1  1  1  1
 1  1  2  1  2  2  1  1  1  1  1  2  1
 1  2  1  1  1  1  1  1  1  1  1  1  1

 2  1  1  1  1  2  1  1  1  1  1  1  2
 1  2  1  1  1  1  1  1  1  1  1  1  1
 2  1  1  1  1  1  1  1  2  1  1  2  1

 1  2  1  1  1  1  1  1  1  1  1  1  2
 2  1  1  1  1  2  2  1  1  1  1  2  1
 1  2  1  1  1  1  1  1  1  1  1  1  1


 208
2  1  1  1  1  1  1  1  1  1  1  1  2
1  2  1  2  2  1  1  1  1  1  1  2  1
2  1  1  1  1  1  1  1  2  1  2  2  1
1  2  1  1  2  1  1  1  1  1  1  1  2

2  1  1  1  1  1  1  2  1  1  1  1  2
1  2  1  1  1  1  1  1    1  1  1  1  2
2  1  1  1  1  1  1  1  2  2  2  1  1
1  2  1  2  1  1  1  1  1  1  1  1  1

2  1  1  1  1  1  1  1  1  1  2  1  1
1  2  1  1  1  1  1  1  1  1  1  1  2
2  1  1  1  1  1  1  2  1  1  1  1  2
1  2  1  1  2  1  2  1  1  1  1  1  2

2  1  1  1  1  1  2  2  1  1  1  2  1
1  1  2 1    2  1  1  2  1  1  1  1  2
1  2  1  1  2  1  1  1  1  1  1  2  1
1  1  2  1  2  1  1  1  1  1  1  1  1
```

Table 10.- Output for matching detailed stratigraphic sections along with statistical information.

SLIDING STEP = 13
PROB(MATCH) = .68977733

FORWARD MATCHES

MATCH POS	NO. OF MATCHES	NO. OF COMPS	MATCHES /COMPS	STD DEVS	CHI-SQ UNCORRECTED	CHI-SQ (YATES)
13	10	13	.7692	.64645	0.3835	0.1021
26	15	26	.5769	-1.1975	1.5475	1.0650
39	24	39	.6154	-.97702	1.0087	0.6910
52	33	52	.6346	-.84169	0.7394	0.5041
65	41	65	.6308	-1.0054	1.0577	0.7999
78	47	78	.6026	-1.6137	2.7725	2.3799
91	65	91	.7143	.51126	0.2554	0.1537
104	66	104	.6346	-1.1903	1.4789	1.2323
117	80	117	.6838	-.14033	0.0198	0.0017
130	80	130	.6154	-1.7838	3.3622	3.0235
143	98	143	.6853	-.11515	0.0133	0.0006
156	100	156	.6410	-1.2913	1.7327	1.5124
169	131	169	.7751	2.5142	5.7560	5.3640
182	119	182	.6538	-1.0327	1.0981	0.9366
195	145	195	.7436	1.6693	2.6389	2.3934
208	129	208	.6202	-2.1136	4.7067	4.3871
221	160	208	.7692	2.5858	6.1363	5.7706
234	127	208	.6106	-2.3987	6.0973	5.7328
247	164	208	.7885	3.2542	9.4662	9.0107
260	121	195	.6205	-2.0373	4.3719	4.0542
273	142	182	.7802	2.7737	6.9572	6.5409
286	105	169	.6213	-1.8755	3.7032	3.3901
299	121	156	.7756	2.4304	5.3748	4.9810
312	90	143	.6294	-1.5258	2.4385	2.1644
325	103	130	.7923	2.6804	6.3865	5.9164
338	77	117	.6581	-.73070	0.5480	0.4100
351	77	104	.7404	1.1444	1.2447	1.0195
364	56	91	.6154	-1.4924	2.3535	2.0187
377	66	78	.8462	3.3151	8.9136	8.1978
390	40	65	.6154	-1.2613	1.6811	1.3514
403	39	52	.7500	.96833	0.8813	0.6224
416	26	39	.6667	-.30900	0.0973	0.0193
429	19	26	.7308	.46102	0.2042	0.0575
442	10	13	.7692	.64645	0.3835	0.1021

SUM CHI-SQ (UNCORRECTED) = 95.8105, WHICH IS
5.6573680 STD DEVS FROM THE MEAN (NORMAL APPROX)

REVERSE MATCHES

MATCH POS	NO. OF MATCHES	NO. OF COMPS	MATCHES /COMPS	STD DEVS	CHI-SQ UNCORRECTED	CHI-SQ (YATES)
-13	8	13	.6154	-.56408	0.3362	0.0784
-26	18	26	.6923	.02792	0.0008	0.0339
-39	23	39	.5897	-1.3043	1.8238	1.3863
-52	35	52	.6731	-.25850	0.0678	0.0122
-65	36	65	.5538	-2.2688	5.6127	4.9954
-78	65	78	.8333	3.0064	7.5120	6.8561
-91	57	91	.6264	-1.2763	1.7096	1.4261
-104	78	104	.7500	1.3694	1.7627	1.4925
-117	70	117	.5983	-2.0708	4.5764	4.1588
-130	102	130	.7846	2.4657	5.4642	5.0300
-143	87	143	.6084	-2.0424	4.4264	4.0542
-156	124	156	.7949	3.0154	8.0520	7.5683
-169	108	169	.6391	-1.3974	2.0320	1.8019
-182	148	182	.8132	3.8796	12.9535	12.3832
-195	123	195	.6308	-1.7413	3.1730	2.9033
-208	157	208	.7548	2.0973	4.1107	3.8124
-221	134	208	.6442	-1.3946	2.0165	1.8092
-234	153	208	.7356	1.4607	2.0389	1.8305

-247	132	208	.6346	-1.6834	2.9577	2.7056
-260	141	195	.7231	1.0214	1.0105	0.8609
-273	113	182	.6209	-1.9580	4.0374	3.7219
-286	122	169	.7219	.91655	0.8146	0.6714
-299	99	156	.6346	-1.4579	2.2183	1.9680
-312	103	143	.7203	.80008	0.6218	0.4874
-325	83	130	.6385	-1.2397	1.5998	1.3690
-338	89	117	.7607	1.7213	2.7490	2.4276
-351	71	104	.6827	-.15572	0.0244	0.0025
-364	66	91	.7253	.74469	0.5359	0.3828
-377	48	78	.6154	-1.3817	2.0173	1.6846
-390	49	65	.7538	1.1544	1.2469	0.9654
-403	31	52	.5962	-1.4120	2.1301	1.7150
-416	25	39	.6410	-.64564	0.4332	0.2353
-429	18	26	.6923	.02792	0.0008	0.0339
-442	10	13	.7692	.64645	0.3835	0.1021

SUM CHI-SQ (UNCORRECTED) = 90.4501, WHICH IS
5.2645611 STD DEVS FROM THE MEAN (NORMAL APPROX)

ACKNOWLEDGMENTS

The manuscript was written while one of the authors (D. F. Merriam) was a Senior Fulbright-Hay Research Fellow at the University of Leicester (U. K.), 1964-65. Computer time was made available through the Computation Centers at the University of Leicester, Dr. A. J. Cole, Director; and Brush Electrical Engineering Co. Ltd., Loughborough, Mr. W. S. Blaschke, Director. The manuscript was typed by Miss Dawn Evans and Mrs. Carol Roper.

REFERENCES

Burnaby, T. P., 1953, A suggested alternative to the correlation coefficient for testing the significance of agreement between pairs of time series and its application to geological data: Nature, v. 172, p. 210.

Dijkstra, E. W., 1962, A primer of ALGOL 60 programming: Academic Press, New York, 114 p.

Elliott, Ltd., 1965, 803 ALGOL: description of 803 library program A 104: Issue no. 4, Sci. Comp. Div., Elliott Bros. Ltd., Borehamwood, Herts., England, 47 p.

Fox, W. T., 1964, FORTRAN and FAP program for calculating and plotting time-trend curves using an IBM 7090 or 7094/1401 computer system: Kansas Geol. Survey Sp. Dist. Publ. 12, 22 p.

Merriam, D. F., 1963, The geologic history of Kansas: Kansas Geol. Survey Bull. 162, 317 p.

Moore, R. C., 1935, Stratigraphic classification of the Pennsylvanian rocks of Kansas: Kansas Geol. Survey Bull. 22, 256 p.

_____, 1949, Divisions of the Pennsylvanian System in Kansas: Kansas Geol. Survey Bull. 83, 203 p.

_____, 1957, Geological understanding of cyclic sedimentation represented by Pennsylvanian and Permian rocks of northern Midcontinent region: Kansas Geol. Soc. 21st Field Conference Guidebook, p. 77-84.

Owen, D. B., 1962, Handbook of statistical tables: Pergamon Press, London, 580 p.

Sackin, M. J., and Sneath, P. H. A., 1965, Amino acid sequences of proteins: a computer study: Biochem. Jour., v. 96, p. 70P-71P.

Sokal, R. R., and Sneath, P. H. A., 1963, Principles of numerical taxonomy: W. H. Freeman and Co., San Francisco and London, 359 p.

KANSAS GEOLOGICAL SURVEY COMPUTER PROGRAM
THE UNIVERSITY OF KANSAS, LAWRENCE

PROGRAM ABSTRACT

Title (If subroutine state in title):

ALGOL Program for Cross-Association of Nonnumeric Sequences using an Elliott 803 Computer

Computer: Elliott 803 Date: July 1965

Programming language: Elliott ALGOL 60

Author, organization: Michael J. Sackin
Medical Research Council Microbial Systematics Research Unit, University of Leicester, Leicester, U.K.

Direct inquiries to: Author or

Name: Daniel F. Merriam Address: Kansas Geological Survey
University of Kansas, Lawrence, Kansas

Purpose/description: Program reads a pair of sequences whose elements belong to a nonordered set, e.g. limestone, shale, sandstone, etc. The data are read in a numeric code. The program "slides" the sequences past each other one or more steps at a time and for each match position counts the number of comparisons (size of overlap). Various significance measures and overall similarity estimates are made.

Mathematical method:

Restrictions, range: Sum of sequence lengths restricted only by size of store

Storage requirements: Compiled program takes about 3000 locations, i.e. 6000 single address instructions

Equipment specifications:

Memory 20K_____ 40K_____ 60K_____ K_____

Automatic divide: Yes_____ No_____ Indirect addressing: Yes_____ No_____

Other special features required Paper tape station, suitable ALGOL compiler.

Additional remarks (include at author's discretion: fixed/float, relocatability; optional: running time, approximate number of times run successfully, programming hours) On a fast Elliott 803C, the run time is normally the output time. Program needs some modification to run at other ALGOL installations.

APPENDIX

```
$IBFTC CRSASS
C                        FORTRAN IV PROGRAM FOR CROSS ASSOCIATION OF NONNUMERIC
C                             SEQUENCES USING AN IBM 7040/7044 COMPUTER
C                                                BY PAUL F. SMITH, JR.
C                                                   BASIC GEOLOGY
C                                                   KANSAS GEOLOGICAL SURVEY
C                                                   KANSAS UNIVERSITY
C                                                   LAWRENCE, KANSAS
C
C                        $$$$$$$ FORM FOR INPUT DATA $$$$$$$
C     FIRST CARD CONTAINS ANY DESCRIPTIVE INFORMATION ABOUT THE FOLLOWING DATA
C
C     SECOND CARD CONTAINS THE CONTROL PARAMETERS AS LISTED BELOW
C         COLUMN  1-2  A T IN THIS FIELD CAUSES THE PROGRAM TO EXIT FOLLOWING
C                        THIS SET OF DATA
C         COLUMN 3-7  LENGTH OF THE LONGER CHAIN (RIGHT JUSTIFIED)
C         COLUMN 8-12 LENGTH OF THE SHORTER CHAIN (RIGHT JUSTIFIED)
C         COLUMN 13-17 LENGTH OF THE SLIDING STEP (RIGHT JUSTIFIED)
C         COLUMN 18-22 AN IDENTIFICATION NUMBER FOR THIS SET OF DATA
C         COLUMN 23-27 NUMBER OF POSSIBLE VALUES AT EACH MATCH POSITION
C     (RIGHT JUSTIFIED)
C         COLUMN 28-35 LOWER LIMIT OF THE RANGE FOR WHICH THE STANDARD DEVIATION
C                        WILL NOT SUPPRESS OUTPUT FOR THE CURRENT MATCH
C                        POSITION.  (DECIMAL POINT IS CONSIDERED TO BE BETWEEN
C                        COLUMNS 31 AND 32 IF NO POINT IS PUNCHED)
C         COLUMN 36-43 UPPER LIMIT OF RANGE DESCRIBED ABOVE. (DECIMAL POINT IS
C                        CONSIDERED TO BE BETWEEN COLUMNS 39 AND 40 IF NO POINT
C                        IS PUNCHED)
C
C     THIRD CARD CONTAINS THE POSSIBLE VALUES FOR EACH MATCH POSITION WITHIN THE
C         SLIDING STEP (MAY BE CONTINUED ON SUCCEEDING CARDS) THIS IS READ IN A
C         FORMAT OF THIRTEEN, SIX CHARACTER ALPHA-NUMERIC WORDS.
C     NEXT CARD CONTAINS THE FORMAT SPECIFICATION FOR THE FOLLOWING DATA. THE
C         CHAINS MUST BE READ IN IN AN A6 FORMAT.
C                $$$$$$$ REPEAT PROCEDURE FOR MULTIPLE SETS OF DATA $$$$$$$
C                        ------- DEFINITIONS -------
C      ALFA IS THE NAME OF THE LONGER CHAIN, BETA IS THE NAME OF THE SHORTER
C      CHAIN, I AND K ARE THEIR RESPECTIVE LENGTHS, L IS THE SLIDING STEP, XMU
C      AND XLAMDA ARE THE LIMITS OF THE OUTPUT SUPPRESSION RANGE, PROMAT IS
C      PROB(MATCH), STDDEV IS THE STANDARD DEVIATION, ENPLUS IS THE MATCH
C      COUNTER, MCOMPS IS THE COMPARISON COUNTER, COMPMA IS MATCHES/COMPARISONS,
C      CHISQU IS THE CHI-SQUARE, CHISQY IS THE YATES CORRECTED CHI-SQUARE,
C      CHISUV IS THE CHI-SQUARE SUM, CHISUY IS THE YATES CHI-SQUARE SUM, NOMAPU
C      IS THE NORMAL APPROXIMATION TO THE STANDARD DEVIATION, NOMAPY IS THE
C      NORMAL APPROXIMATION TO THE STANDARD DEVIATION FOR THE YATES CHI-SQUARE,
C      MLAP IS THE LOW ALFA POSITION, MHAP IS THE HIGH ALFA POSITION
C             ADDRESS ANY QUESTIONS TO THE AUTHOR OR DR. DANIEL F. MERRIAM
C                                                   218 LINDLEY HALL
C                                                   KANSAS GEOLOGICAL SURVEY
C                                                   KANSAS UNIVERSITY
C                                                   LAWRENCE, KANSAS 66044
C     ALL MATHEMATICAL FORMULAE USED IN THIS PROGRAM ARE FROM THE        CRSAS 51
C     KANSAS GEOLOGICAL SURVEY SPECIAL DISTRIBUTION PUBLICATION NO. 23   CRSAS 52
      DIMENSION ALFA(1000), BETA(1000), DIFGRP(25,25),FMT(14),INFO(14)   CRSAS 53
      INTEGER ALFA,BETA,ZILCH                                            CRSAS 54
      REAL NOMAPU,NOMAPY                                                 CRSAS 55
      LOGICAL LAST,XEXIT, PAST                                           CRSAS 56
    1 IF (LAST) CALL EXIT                                                CRSAS 57
      DATA ZILCH/6H      /                                               CRSAS 58
      READ(5,2) INFO                                                     CRSAS 59
      READ (5,10)LAST,I,K,L,NUMDAT,NDIFF,XLAMDA,XMU,MM                   CRSAS 60
      CALL ARDVRK(I,K,L,NDIFF,XMU,XLAMDA)                                CRSAS 61
   10 FORMAT (L2,5I5,2F8.4,I2)                                           CRSAS 62
      READ (5,2) FMT                                                     CRSAS 63
      READ (5,FMT) ((DIFGRP(J,M),M=1,NDIFF),J=1,L)                       CRSAS 64
      READ(5,2)FMT                                                       CRSAS 65
    2 FORMAT(13A6,A2)                                                    CRSAS 66
      READ(5,FMT)(ALFA(J),J = 1,I)                                       CRSAS 67
      READ(5,FMT)(BETA(J),J = 1,K)                                       CRSAS 68
      XEXIT = .FALSE.                                                    CRSAS 69
      WRITE(6,21)INFO                                                    CRSAS 70
   21 FORMAT(1H1,13A6,A2)                                                CRSAS 71
```

```
      WRITE (6,40) NUMDAT                                                    CRSAS 72
   40 FORMAT (1X ,45X,40HFORWARD MATCHES SEQUENCE IDENTIFICATION ,I4///)CRSAS 73
      GO TO 42                                                               CRSAS 74
   41 IF(XEXIT) GO TO 39                                                     CRSAS 75
      GO TO 38                                                               CRSAS 76
   39 CALL SIMIND(ALFA,BETA,I,K,L,SSUBL)                                     CRSAS 77
      WRITE(6,1001) PROMAT                                                   CRSAS 78
 1001 FORMAT(1X,13HPROB(MATCH) =,F8.4)                                       CRSAS 79
      WRITE (6,1000) SSUBL                                                   CRSAS 80
 1000 FORMAT (1X//1X,18HSIMILARITY INDEX =,F8.4)                             CRSAS 81
      GO TO 1                                                                CRSAS 82
   38 XEXIT = .TRUE.                                                         CRSAS 83
      WRITE(6,21)INFO                                                        CRSAS 84
      WRITE (6,43) NUMDAT                                                    CRSAS 85
   43 FORMAT      (46X,40HREVERSE MATCHES SEQUENCE IDENTIFICATION ,I4///)CRSAS 86
   42 WRITE (6,45)                                                           CRSAS 87
   45 FORMAT (1X,7HLO ALFA,5X,7HHI ALFA,11X,6HNUMBER,10X,9HNUMBER OF,9X,CRSAS 88
     18HMATCHES/,9X,8HSTANDARD,7X,10HCHI-SQUARE,8X,10HCHI-SQUARE/1X,8HPOCRSAS 89
     2SITION,4X,8HPOSITION,7X,10HOF MATCHES,7X,11HCOMPARISONS,7X,11HCOMPCRSAS 90
     3ARISONS,8X,10HDEVIATIONS,4X,11HUNCORRECTED,8X,8H(YATES) //)          CRSAS 91
      CHISUV = 0.                                                            CRSAS 92
      CHISUY = 0.                                                            CRSAS 93
      DO 110 J=L,I,L                                                         CRSAS 94
      NZERO = 0                                                              CRSAS 95
      NPLUS = 0                                                              CRSAS 96
      J1 = J                                                                 CRSAS 97
      K1 = K                                                                 CRSAS 98
   50 IF(ALFA(J1).EQ.BETA(K1).AND.ALFA(J1).NE.ZILCH.AND.BETA(K1).NE.ZILCCRSAS 99
     1H) NPLUS = NPLUS + 1                                                   CRSAS100
      IF (ALFA(J1).EQ.ZILCH.OR.BETA(K1).EQ.ZILCH) NZERO = NZERO + 1          CRSAS101
      K1 = K1 - 1                                                            CRSAS102
      J1 = J1 - 1                                                            CRSAS103
      IF(J1.LT.1.OR.K1.LT.1) GO TO 55                                        CRSAS104
      GO TO 50                                                               CRSAS105
   55 MN = K-J                                                               CRSAS106
      MLAP = J-K+1                                                           CRSAS107
      IF (MN.GE.0) MLAP = 1                                                  CRSAS108
      MHAP = J                                                               CRSAS109
      MCOMPS = K                                                             CRSAS110
      IF (MN.GT.0) MCOMPS = J                                                CRSAS111
      MCOMPS = MCOMPS - NZERO                                                CRSAS112
      CALL PROBAB(ALFA,BETA,DIFGRP,I,K,L,ZILCH,NDIFF,PROMAT)                 CRSAS113
      EMCOMP = MCOMPS                                                        CRSAS114
      ENPLUS = NPLUS                                                         CRSAS115
      IF (NPLUS.GT.MCOMPS)GO TO 105                                          CRSAS116
      COMPMA = ENPLUS / EMCOMP                                               CRSAS117
      STDDEV = SQRT(EMCOMP)*(2.*(ARSIN(SQRT(COMPMA)))-2.*(ARSIN(SQRT(PROCRSAS118
     1MAT))))                                                                CRSAS119
      IF(STDDEV.GT.XMU.OR.STDDEV.LT.XLAMDA)GO TO 110                         CRSAS120
      PROEMM = (1. - PROMAT)                                                 CRSAS121
      PROEMC = PROMAT * EMCOMP                                               CRSAS122
      X = ENPLUS - PROEMC                                                    CRSAS123
      XX =(EMCOMP-ENPLUS)- (PROEMM * EMCOMP)                                 CRSAS124
      CHISQU = (X**2/PROEMC) + (XX**2/(PROEMM * EMCOMP))                     CRSAS125
      CHISUV = CHISUV + CHISQU                                               CRSAS126
      CHISQY=((ABS(X)-.5)**2)/PROEMC+((ABS(XX)-.5)**2)/(PROEMM*EMCOMP)       CRSAS127
      WRITE (6,100) MLAP,MHAP,NPLUS,MCOMPS,COMPMA,STDDEV,CHISQU,CHISQY       CRSAS128
  100 FORMAT (3X,I4,8X,I4,12X,I4,13X,I4,12X,F8.4,8X,F8.4,9X,F8.4,10X,F8.CRSAS129
     14)                                                                     CRSAS130
      GO TO 110                                                              CRSAS131
  105 WRITE(6,201)                                                           CRSAS132
  110 CONTINUE                                                               CRSAS133
      DO 210  J=L,K,L                                                        CRSAS134
      NZERO = 0                                                              CRSAS135
      NPLUS = 0                                                              CRSAS136
      KI = K - J                                                             CRSAS137
      JI = I                                                                 CRSAS138
  130 IF(ALFA(JI).EQ.BETA(KI).AND.ALFA(JI).NE.ZILCH.AND.BETA(KI).NE.ZILCCRSAS139
     1H) NPLUS = NPLUS + 1                                                   CRSAS140
      IF(ALFA(JI).EQ.ZILCH.OR.BETA(KI).EQ.ZILCH) NZERO = NZERO + 1           CRSAS141
      KI = KI-1                                                              CRSAS142
      JI = JI-1                                                              CRSAS143
      IF(JI.LT.1.OR.KI.LT.1) GO TO 150                                       CRSAS144
```

```
      GO TO 130                                                          CRSAS145
  150 MHAP = I                                                           CRSAS146
      MLAP = I-K+J+1                                                     CRSAS147
      MCOMPS = I - MLAP +1                                               CRSAS148
      MCOMPS = MCOMPS - NZERO                                            CRSAS149
      ENDIFF = NDIFF                                                     CRSAS150
      EMCOMP = MCOMPS                                                    CRSAS151
      ENPLUS = NPLUS                                                     CRSAS152
      COMPMA =ENPLUS /EMCOMP                                             CRSAS153
      IF (NPLUS.GT.MCOMPS) GO TO 200                                     CRSAS154
      STDDEV = SQRT(EMCOMP)*(2.*(ARSIN(SQRT(COMPMA)))-2.*(ARSIN(SQRT(PROCRSAS155
     1MAT))))                                                            CRSAS156
      IF(STDDEV.GT.XMU.OR.STDDEV.LT.XLAMDA)GO TO 210                     CRSAS157
      PROEMM = (1. - PROMAT)                                             CRSAS158
      PROEMC = PROMAT * EMCOMP                                           CRSAS159
      X = ENPLUS - PROEMC                                                CRSAS160
      XX =(EMCOMP-ENPLUS)- (PROEMM * EMCOMP)                             CRSAS161
      CHISQU = (X**2/PROEMC) + (XX**2/(PROEMM * EMCOMP))                 CRSAS162
      CHISUV = CHISUV + CHISQU                                           CRSAS163
      CHISQY=((ABS(X)-.5)**2)/PROEMC+((ABS(XX)-.5)**2)/(PROEMM*EMCOMP)   CRSAS164
      IF (MCOMPS.LE.0) GO TO 210                                         CRSAS165
      WRITE (6,100) MLAP,MHAP,NPLUS,MCOMPS,COMPMA,STDDEV,CHISQU,CHISQY   CRSAS166
      GO TO 210                                                          CRSAS167
  200 WRITE (6,201)                                                      CRSAS168
  201 FORMAT(1X,29HMORE MATCHES THAN COMPARISONS )                       CRSAS169
  210 CONTINUE                                                           CRSAS170
      EYEK2 = 2 * (I+K)/L-3                                              CRSAS171
      NOMAPU = SQRT(2.* CHISUV)-SQRT (EYEK2)                             CRSAS172
      WRITE(6,220)CHISUV,NOMAPU,L                                        CRSAS173
  220 FORMAT (1X///1X,20HTHE CHI-SQUARE SUM =,F10.4,1H.2X,8HWHICH IS,F8.CRSAS174
     14,2X,50HSTANDARD DEVIATIONS FROM THE MEAN (NORMAL APPROX.)//1X,14HCRSAS175
     2SLIDING STEP =,I3)                                                 CRSAS176
      CALL CHNREV(BETA,L,K)                                              CRSAS177
      GO TO 41                                                           CRSAS178
      END                                                                CRSAS179
$IBFTC ARDVRK                                                            CRSAS180
C     ERROR ROUTINE FOR INPUT PARAMETERS                                 CRSAS181
      SUBROUTINE ARDVRK(I,K,L,NDIFF,XMU,XLAMDA)                          CRSAS182
      DIMENSION KX(6)                                                    CRSAS183
      IF(XLAMDA.GE.XMU) KX(1)  = 1                                       CRSAS184
      IF(K.GT.I) KX(2) = 2                                               CRSAS185
      IF(L.GT.I.OR.L.GT.K) KX(3) = 3                                     CRSAS186
      KDLXL = (K/L)*L                                                    CRSAS187
      IDLXL = (I/L)*L                                                    CRSAS188
      IF(KDLXL.NE.K) KX(4) = 4                                           CRSAS189
      IF(IDLXL.NE.I) KX(5) = 5                                           CRSAS190
      IF(NDIFF.LE.0)KX(6)  = 6                                           CRSAS191
      KSUM = 0                                                           CRSAS192
      IF(L.LT.0)KSUM = 2                                                 CRSAS193
      DO 100 M = 1,6                                                     CRSAS194
  100 KSUM = KSUM + KX(M)                                                CRSAS195
      IF(KSUM.GT.0) GO TO 200                                            CRSAS196
      RETURN                                                             CRSAS197
  200 WRITE(6,25)                                                        CRSAS198
      WRITE(6,10)                                                        CRSAS199
   25 FORMAT (1H1)                                                       CRSAS200
   10 FORMAT(30X,17HAARDVARK ASSEMBLY///35X,14HERROR MESSAGES//)         CRSAS201
    7 FORMAT(20X,25HNON-POSITIVE SLIDING STEP/)                          CRSAS202
      IF(XLAMDA.GE.XMU)WRITE(6,1)                                        CRSAS203
      IF(K.GT.I) WRITE(6,2)                                              CRSAS204
      IF(L.GT.I.OR.L.GT.K)WRITE(6,3)                                     CRSAS205
      IF(KDLXL.NE.K)WRITE(6,4)                                           CRSAS206
      IF(IDLXL.NE.I)WRITE(6,5)                                           CRSAS207
      IF(NDIFF.LE.0)WRITE(6,6)                                           CRSAS208
    1 FORMAT(20X,31HCHECK XLAMDA AND XMU FOR ERRORS/)                    CRSAS209
    2 FORMAT(20X,46HBETA IS LONGER THAN ALFA.CHANGE THIS SITUATION/)     CRSAS210
    3 FORMAT(20X,46HCHECK L, I, ANDK FOR POSSIBLE INCORRECT INPUT/)      CRSAS211
    4 FORMAT(20X,65HLENGTH OF BETA CHAIN IS NOT EQUALLY DIVISIBLE BY THECRSAS212
     1 SLIDING STEP/)                                                    CRSAS213
    5 FORMAT(20X,65HLENGTH OF ALFA CHAIN IS NOT EQUALLY DIVISIBLE BY THECRSAS214
     1 SLIDING STEP/)                                                    CRSAS215
    6 FORMAT(20X,36HYOU HAVE NOT ENTERED NDIFF CORRECTLY/)               CRSAS216
      CALL EXIT                                                          CRSAS217
      END                                                                CRSAS218
```

```
$IBFTC TWNT5                                                            CRSAS219
C       THIS SUBROUTINE IS USED IN COMPUTING THE SIMILARITY INDEX       CRSAS220
        SUBROUTINE TWNT5(K1,J1,ALFA,BETA,ZILCH,MSUBL)                   CRSAS221
        DIMENSION ALFA(50), BETA(50)                                    CRSAS222
        INTEGER ALFA , BETA, ZILCH                                      CRSAS223
        MSUBL = MSUBL + 1                                               CRSAS224
        KII = K1 + 2 * L                                                CRSAS225
        JII = J1 + 2 * L                                                CRSAS226
        ALFA(JII) = ZILCH                                               CRSAS227
        BETA(KII) = ZILCH                                               CRSAS228
   20 CONTINUE                                                          CRSAS229
        RETURN                                                          CRSAS230
        END                                                             CRSAS231
$IBFTC SIMIND  NODECK                                                   CRSAS232
C       THIS SUBROUTINE COMPUTES THE SIMILARITY INDEX                   CRSAS233
        SUBROUTINE SIMIND(ALFA,BETA,I,K,L,SSUBL)                        CRSAS234
        DIMENSION ALFA(1), BETA(1)                                      CRSAS235
        INTEGER ALFA, BETA, ZILCH                                       CRSAS236
        DATA ZILCH/6H      /                                            CRSAS237
        LOGICAL MATCH, FINISH                                           CRSAS238
        FINISH = .FALSE.                                                CRSAS239
        MSUBL = 0                                                       CRSAS240
        L2 = L * 2                                                      CRSAS241
   10 DO 100 J = L2,I,L                                                 CRSAS242
        MATCH = .FALSE.                                                 CRSAS243
        NSUM = 0                                                        CRSAS244
        DO 20 JJ = 1,L                                                  CRSAS245
        J1 = J - JJ + 1                                                 CRSAS246
        K1 = K - JJ + 1                                                 CRSAS247
   20 IF(ALFA(J1).EQ.BETA(K1).AND.ALFA(J1).NE.ZILCH.AND.BETA(K1).NE.ZILCCRSAS248
     1H) NSUM = NSUM + 1                                                CRSAS249
        MATCH = .FALSE.                                                 CRSAS250
        IF(NSUM.EQ.L) MATCH = .TRUE.                                    CRSAS251
   30 IF(J1.LT.1) GO TO 100                                             CRSAS252
        NSUM = 0                                                        CRSAS253
        DO 50 JJ= 1,L                                                   CRSAS254
        K1 = K1 - 1                                                     CRSAS255
        J1 = J1 - 1                                                     CRSAS256
   50 IF(ALFA(J1).EQ.BETA(K1).AND.ALFA(J1).NE.ZILCH.AND.BETA(K1).NE.ZILCCRSAS257
     1H) NSUM = NSUM + 1                                                CRSAS258
        IF(NSUM.EQ.L.AND.MATCH)CALL TWNT5(K1,J1,ALFA,BETA,ZILCH,MSUBL)  CRSAS259
        MATCH =    .FALSE.                                              CRSAS260
        IF(NSUM.EQ.L) MATCH = .TRUE.                                    CRSAS261
        GO TO 30                                                        CRSAS262
  100 CONTINUE                                                          CRSAS263
        KML2 = K - L2                                                   CRSAS264
        DO 200 J = L,KML2,L                                             CRSAS265
        NSUM = 0                                                        CRSAS266
        DO 120 JJ = 1,L                                                 CRSAS267
        J1 = I                                                          CRSAS268
        K1 = K - JJ                                                     CRSAS269
  120 IF(ALFA(J1).EQ.BETA(K1).AND.ALFA(J1).NE.ZILCH.AND.BETA(K1).NE.ZILCCRSAS270
     1H) NSUM = NSUM + 1                                                CRSAS271
        IF(NSUM.EQ.L.AND.MATCH) CALL TWNT5(K1,J1,ALFA,BETA,ZILCH,MSUBL)  CRSAS272
        MATCH = .FALSE.                                                 CRSAS273
        IF(NSUM.EQ.L) MATCH = .TRUE.                                    CRSAS274
  130 IF(K1.LT.1) GO TO 200                                             CRSAS275
        NSUM = 0                                                        CRSAS276
        DO 150 JJ = 1,L                                                 CRSAS277
        K1 = K1 - 1                                                     CRSAS278
        J1 = J1 - 1                                                     CRSAS279
  150 IF(ALFA(J1).EQ.BETA(K1).AND.ALFA(J1).NE.ZILCH.AND.BETA(K1).NE.ZILCCRSAS280
     1H) NSUM = NSUM + 1                                                CRSAS281
        IF(NSUM.EQ.L.AND.MATCH) CALL TWNT5(K1,J1,ALFA,BETA,ZILCH,MSUBL)  CRSAS282
        MATCH = .FALSE.                                                 CRSAS283
        IF(NSUM.EQ.L) MATCH = .TRUE.                                    CRSAS284
        GO TO 130                                                       CRSAS285
  200 CONTINUE                                                          CRSAS286
        IF(FINISH) GO TO 250                                            CRSAS287
        CALL CHNREV(BETA,L,K)                                           CRSAS288
        FINISH =.TRUE.                                                  CRSAS289
        GO TO 10                                                        CRSAS290
  250 SSUBL = MSUBL                                                     CRSAS291
        EYMU = (I - L) / L                                              CRSAS292
```

```
      SSUBL = SSUBL / EYMU                                              CRSAS293
      RETURN                                                            CRSAS294
      END                                                               CRSAS295
$IBFTC CHNREV                                                           CRSAS296
C     THIS SUBROUTINE REVERSES THE CHAINS                               CRSAS297
      SUBROUTINE CHNREV(B,L,K)                                          CRSAS298
      DIMENSION B(1000)                                                 CRSAS299
      K2 =  K / 2                                                       CRSAS300
      DO 100 J = 1,K2                                                   CRSAS301
      JJ = K - J + 1                                                    CRSAS302
      B1 = B(J)                                                         CRSAS303
      B(J) = B(JJ)                                                      CRSAS304
      B(JJ)= B1                                                         CRSAS305
  100 CONTINUE                                                          CRSAS306
      IF(L.LE.1) GO TO 210                                              CRSAS307
      LD2 = L / 2                                                       CRSAS308
      KML = K - L                                                       CRSAS309
      JCOUNT = 0                                                        CRSAS310
      DO 200JJ=1,K,L                                                    CRSAS311
      JCOUNT = JCOUNT + 1                                               CRSAS312
      DO 200 JLI = 1,LD2                                                CRSAS313
      JL = JCOUNT * L - JLI + 1                                         CRSAS314
      J = JJ + JLI - 1                                                  CRSAS315
      B1 = B(J)                                                         CRSAS316
      B(J) = B(JL)                                                      CRSAS317
      B(JL) = B1                                                        CRSAS318
  200 CONTINUE                                                          CRSAS319
  210 CONTINUE                                                          CRSAS320
      RETURN                                                            CRSAS321
      END                                                               CRSAS322
$IBFTC PROBAB                                                           CRSAS323
C     THIS SUBROUTINE COMPUTES PROB(MATCH)                              CRSAS324
      SUBROUTINE PROBAB(ALFA,BETA,DIFGRP,I,K,L,ZILCH,NDIFF,PROMAT)      CRSAS325
      DIMENSION ALFA(1000), BETA(1000), DIFGRP(25,25)                   CRSAS326
      NSUM = 0                                                          CRSAS327
      DO 100  N = 1,L                                                   CRSAS328
      DO 100  M = 1,NDIFF                                               CRSAS329
      NBESUM = 0                                                        CRSAS330
      NALSUM = 0                                                        CRSAS331
      DO 30  J = N,I,L                                                  CRSAS332
   30 IF (ALFA(J).EQ.DIFGRP(N,M))NALSUM = NALSUM+1                      CRSAS333
      DO 50  J = N,K,L                                                  CRSAS334
   50 IF(BETA(J).EQ.DIFGRP(N,M))NBESUM = NBESUM + 1                     CRSAS335
      NSUM = NSUM + NALSUM * NBESUM                                     CRSAS336
  100 CONTINUE                                                          CRSAS337
      SUMZ1 = 0.                                                        CRSAS338
      SUMZ2 = 0.                                                        CRSAS339
      ZERO  = 0.                                                        CRSAS340
      DO 130  J = 1,I                                                   CRSAS341
  130 IF(ALFA(J).EQ.ZILCH)SUMZ1 = SUMZ1 + 1.                            CRSAS342
      AI = I                                                            CRSAS343
      ZERO = ZERO + AI * SUMZ1                                          CRSAS344
      DO 150 J = 1,K                                                    CRSAS345
  150 IF(BETA(J).EQ.ZILCH) SUMZ2 = SUMZ2 + 1.                           CRSAS346
      AK = K                                                            CRSAS347
      ZERO = ZERO + AK * SUMZ2                                          CRSAS348
      EYK = I * K                                                       CRSAS349
      EL = L                                                            CRSAS350
      EYKDL = EYK / EL                                                  CRSAS351
      ZERO =(ZERO -SUMZ1 * SUMZ2) / EL                                  CRSAS352
      ENSUM = NSUM                                                      CRSAS353
      PROMAT = ENSUM / (EYKDL - ZERO)                                   CRSAS354
      RETURN                                                            CRSAS355
      END                                                               CRSAS356
$ENTRY                                                                  CRSAS357
```

COMPUTER CONTRIBUTIONS
Kansas Geological Survey
University of Kansas
Lawrence, Kansas

Daniel F. Merriam, Editor

Special Distribution Publication

3. BALGOL program for trend-surface mapping using an IBM 7090 computer, by J. W. Harbaugh, 1963. . . . . $0.50
4. FORTRAN II program for coefficient of association (Match-Coeff) using an IBM 1620 computer, by R. L. Kaesler, F. W. Preston, and D. I. Good, 1963. . . . . $0.25
9. BALGOL programs for calculation of distance coefficients and correlation coefficients using an IBM 7090 computer, by J. W. Harbaugh, 1964. . . . . $0.50
11. Trend-surface analysis of regional and residual components of geologic structure in Kansas, by D. F. Merriam and J. W. Harbaugh, 1964. . . . . $0.50
12. FORTRAN and FAP program for calculating and plotting time-trend curves using an IBM 7094/1401 computer system, by W. T. Fox, 1964. . . . . $0.50
13. FORTRAN program for factor and vector analysis of geologic data using an IBM 7090 or 7094/1401 computer system, by Vincent Manson and John Imbrie, 1964. . . . $0.50
14. FORTRAN II trend-surface program for the IBM 1620, by D. I. Good, 1964. . . . . $0.50
15. Application of factor analysis to petrologic variations of Americus Limestone (Lower Permian), Kansas and Oklahoma, by J. W. Harbaugh and Ferruh Demirmen, 1964 . . $0.50
23. ALGOL program for cross-association of nonnumeric sequences using a medium size computer, by M. J. Sackin, P. H. A. Sneath, and D. F. Merriam, 1965 . . . . . $0.50

Report of Studies

170-3 Mathematical conversion of section, township, and range notation to Cartesian Coordinates, by D. I. Good, 1964. . . . . $0.25

Bulletin

171 A computer method for four-variable trend analysis illustrated by a study of oil-gravity variations in southeastern Kansas, by J. W. Harbaugh, 1964. . . . . $0.75

Reprints (available for limited time)

Computer helps map oil structures, by D. F. Merriam and J. W. Harbaugh (reprinted from the Oil and Gas Journal, v. 61, no. 47, 1963). . . . . no charge

Use of trend-surface residuals in interpreting geologic structures, by D. F. Merriam (reprinted from Stanford University Publications, Geological Sciences, v. 9, no. 2, 1964). . . . . no charge

Use of asymmetric frequency distribution curves of core analysis data in calculating oil reserves, by F. W. Preston and J. S. Van Scoyoc (reprinted from Stanford University Publications, Geological Sciences, v. 9, no. 2, 1964). . . . . no charge

Pattern recognition studies of geologic structure using trend-surface analysis, by D. F. Merriam and R. H. Lippert (reprinted from Colorado School Mines Quarterly, v. 59, no. 4, 1964). . . . . no charge

Trend-surface mapping of hydrodynamic oil traps with the IBM 7090/94 computer, by J. W. Harbaugh (reprinted from Colorado School Mines Quarterly, v. 59, no. 4, 1964) . . . . . no charge

Fourier series analysis in geology, by J. W. Harbaugh and F. W. Preston (reprinted from College of Mines, Arizona University, v. 1, 1965). . . . . no charge

Geology and the computer, by D. F. Merriam (reprinted from New Scientist, v. 26, no. 444, 1965). . . . . no charge