Routine to provide a random order for counterbalanced variables

C. JAMES SCHEIRER

State University of New York, Binghamton, New York 13901

One of the most common exercises in psychology is the randomization of a set of counterbalanced variables. While the problem is a logically simple one, the task itself can be laborious and time consuming. The routine described here accepts up to five variables, produces all combinations of levels of these variables, and prints a random order of this counterbalancing. After initialization, dimensioning and data input, the program is only 27 code statements long.

Table 1
Program to Provide Random Orders: Five Variables

```
10 REM
20 DIM AS(1)
30 PRINT "HOW MANY VARIABLES, LIMIT IS 5"
40 INPUT C
50 L1=1
60 L2=1
70 L3=1
80 L4=1
90 L5=1
100 REM - L1 TO L5 ARE THE NUMBER OF LEVELS OF EACH VARIABLE
110 ;"LEVELS OF FIRST"
120 INPUT L1
130 IF C=1 GO TO 250
140 ;"SECOND"
150 INPUT L2
160 IF C=2 GO TO 250
170 ;"THIRD"
180 INPUT L3
190 IF C=3 GO TO 250
200 ;"FOURTH"
210 INPUT L4
220 IF C=4 GO TO 250
230 ;"FIFTH"
240 INPUT L5
250 LET T = L1*L2*L3*L4*L5
260 REM
270 DIM X(T),Y(T)
280 R=0
290 FOR J=1 TO L1
300 FOR J=1 TO L2
310 FOR K=1 TO L3
320 FOR L=1 TO L4
330 FOR M=1 TO L5
340 R=R+1
350 IF C=1 THEN LET Y(R)=I
360 IF C=2 THEN LET Y(R)=I+J
370 IF C=3 THEN LET Y(R)=I+J+K
380 IF C=4 THEN LET Y(R)=I+J+K+L
390 IF C=5 THEN LET Y(R)=I+J+K+L+M
400 REM - FOR CONVENIENCE IN OUTPUT
410 NEXT M
420 NEXT L
430 NEXT K
440 NEXT J
450 NEXT I
460 REM - Y NOW CONTAINS ALL POSSIBLE ARRANGEMENTS, IN ORDER
470 X(0)=0
480 FOR I=1 TO T
490 X(I)=INT(T*RND(0))+1
500 FOR J=1 TO I
510 IF X(I)=X(J-1) GO TO 490
520 NEXT J
530 NEXT I
540 REM - X(P) IS A RANDOM ORDER OF THE INTEGERS 1 TO T.
550 ;
560 FOR P=1 TO T
570 PRINT Y(X(P))
580 NEXT P
590 ;
600 ;"DO YOU WANT ANOTHER BLOCK?"
610 INPUT AS
620 IF AS="Y" GO TO 470
630 STOP
640 END
```

The program produces a systematic arrangement of all combinations of levels of variables and stores it in a Vector Y. It then randomly arranges the integers from 1 to the length of Y(R) and stores each integer as a value of X(P). The statement "PRINT Y(X(P))" then produces the desired output.

It might be noted that the random ordering of a sequence (or generation of a set of random sequences) of a specified length is a fairly common problem. In this program, this is accomplished in a subroutine of only seven code statements (470-530) and is easily adaptable as a stand-alone program. In fact, if the user specifies only one variable with L1 levels, the output of this program is a random order of the sequence 1 to L1.

Support. This program was developed on an Interdata Model 70 minicomputer. It is written in INTERACTIVE BASIC but would be relatively simple to convert to BASIC or FORTRAN. With the Interdata Model 70 machine, the program uses 1,080 bytes of core before dimensioning.

Availability. Requests for copies of the program should be sent to C. James Scheirer, Department of Psychology, State University of New York at Binghamton, Binghamton, New York 13901.

A BASIC subroutine to generate random Latin squares

C. JAMES SCHEIRER

*State University of New York at Binghamton
Binghamton, New York 13901*

Latin squares are useful in providing experimental arrangements for certain types of experimental designs. Tables exist for small squares from which it is possible to generate randomly all possible

Table 1
Generation of a Set of Random Latin Squares

```
10 REM
20 ;"ENTER RANDOM NUMBER"
30 REM INITIALIZE RANDOM NUMBER
40 REM GENERATOR
50 INPUT R2
60 FOR I=1 TO R2
70 X=RND(0)
80 NEXT I
90 ;"WHAT SIZE SQUARE?"
100 INPUT N
110 ;"HOW MANY SQUARES DO YOU WANT?"
120 INPUT K
130 DIM A(N,N),B(N,N)
140 FOR Q=1 TO K
150 ;
160 MAT A=(0)
170 FOR J=1 TO N
180 FOR I=1 TO N
190 A(I,J)=0
200 NEXT I
210 FOR I=1 TO N
220 T=0
230 REM T IS NUMBER OF TIMES A SPECIFIC
240 REM ELEMENT HAS BEEN SAMPLED
250 T=T+1
260 IF T > 2*N GO TO 180
270 REM
280 REM IF HAVEN'T FOUND AN ACCEPTABLE
290 REM ELEMENT IN 2*N TRIES, RESET
300 REM ENTIRE COLUMN AND TRY AGAIN
310 REM
320 REM SAMPLE A NUMBER
330 FOR Y=1 TO (I-1)
340 IF Y=I GO TO 370
350 REM
360 REM IF THIS NUMBER HAS APPEARED BEFORE
370 REM IN THIS ROW, RESAMPLE
380 NEXT Y
390 IF Z=J GO TO 430
400 IF A(I,J)=A(I,Z) GO TO 250
410 REM
420 REM IF THIS NUMBER HAS APPEARED BEFORE
430 REM IN THIS COLUMN, RESAMPLE
440 NEXT I
450 NEXT J
460 REM
470 MAT PRINT A;
480 NEXT Q
490 STOP
500 END
BASIC
```