REMARK ON ALGORITHM 352 [S22] CHARACTERISTIC VALUES AND ASSOCIATED SOLUTIONS OF MATHIEU'S DIFFERENTIAL EQUATION [D. S. Clemm, *Comm. ACM 12* (July 1969), 399–407]

ARTHUR H. J. SALE (Recd. 4 May 1970 and 28 May 1970)
University of Sydney, Sydney, NSW, Australia

KEY WORDS AND PHRASES: Mathieu's differential equation, Mathieu function, characteristic value, periodic solution, radial solution
CR CATEGORIES: 5.12

This algorithm contains a number of syntactically incorrect *FORMAT* statements: labeled 901, 911, 921, 941, and 951 in subroutine *MFCVAL*, and 99 in the functions *FJ*, *FY*, *DJ*, and *DY*. The error consists of omitting a comma separating the Hollerith field descriptor and the integer field descriptor, as required by Sections 7.2.3 and 7.2.3.2 of the Fortran standard [1, 2]. In all cases this may be corrected by inserting a comma immediately preceding the field descriptor *I*3 in these statements.

It has also been pointed out by the referee and the Algorithms Editor that the two *FORMAT* statements in functions *DJ* and *DY* contain a character not in the standard Fortran character set. The standard is somewhat ambiguous on this point: any representable character is permitted in a Hollerith constant in a *CALL* or a *DATA* statement, and also in data to be read in with an *Aw* field descriptor (Sections 4.2.6, 5.1.1.6), but since Hollerith field descriptors are not Hollerith constants, it must be presumed that the prohibition of Section 3.1 applies. The "at" symbol (@) in these two statements should therefore be replaced by a blank or some other character in the standard set.

There is another, more serious, error: subroutines *BOUNDS* and *MFITR*8 both reference a named common block which is not referenced by the routine that calls them (*MFCVAL*). According to Section 10.2.5 of the standard, the contents of this block will therefore become undefined at the moment either of these two routines executes a *RETURN*, unless this common block is referenced by a routine which is directly or indirectly calling *MFCVAL*. This undefinition permits named common blocks to be overlaid, and since it is not the author's intention to allow this block to become undefined, the following two statements should be added to *MFCVAL* immediately following the existing *DOUBLE PRECISION* and *COMMON* statements respectively:

```
DOUBLE PRECISION FILL(3)
COMMON /MF2/ FILL
```

REFERENCES:

1. ANSI Standard Fortran ANSI (USASI) X3.9-1966. American National Standards Institute, New York, 1966.
2. FORTRAN vs Basic FORTRAN. *Comm. ACM 7* (Oct. 1964), 591–625.

REMARK ON ALGORITHM 384 [F2] EIGENVALUES AND EIGENVECTORS OF A REAL SYMMETRIC MATRIX [G. W. Stewart, *Comm. ACM 6* (June 1970), 369–371]

G. W. STEWART
Department of Computer Sciences, The University of Texas at Austin, Austin, TX 78712

KEY WORDS AND PHRASES: real symmetric matrix, eigenvalues, eigenvectors, *QR* algorithm
CR CATEGORIES: 5.14

The following changes should be made in the subroutine *SYMQR*. Change the statement:

```
 REAL
1A(NA,1),D(1),E(1),K0,D1,D2,...
```

to:

```
 REAL
1A(NA,1),D(1),E(1),K0,K1,K2,...
```

After statement number 230 delete the statements:

```
     IF(E(200).NE.0.) PRINT 2000,(D(I),E(I),I=1,NU)
2000 FORMAT(1H010E12.4/(1H 10E12.4))
```

Replace the statements:

```
240 E(NU) = E(NU)+FLOAT(NUM1-NL)
    IF(1. .EQ. 1.) GO TO 250
    IF(0. .EQ. 1.) GO TO 250
```

by:

```
240 E(NU) = E(NU)+1.
    IF(E(NU) .LE. TITTER) GO TO 250
```

Replace the statements:

```
300 IF(E(200).NE.0. .AND. K.LE.1.E-14*ABS(D(NL)))K=0.
    P = D(NL) - K
```

by:

```
300 P = D(NL) - K
```

REMARK ON ALGORITHM 385 [S13] EXPONENTIAL INTEGRAL *Ei(x)* [Kathleen A. Paciorek, *Comm. ACM 13* (July 1970), 446–447]

K. A. REDISH (Recd. 3 Aug. 1970)
Department of Applied Mathematics, Hamilton, McMaster University, Ontario, Canada

KEYWORDS AND PHRASES: ANSI Fortran standard
CR CATEGORIES: 4.0, 4.22

(a) This algorithm does not conform to the standard in that the DATA statements contain array names. Section 7.2.2 of ANSI Fortran standard [*Comm. ACM 7* (Oct. 1964), 590–625] (1) states that the list(s) of a data statement contain "names of variables and array elements." It is therefore necessary to list the elements singly. (A more readable layout can be obtained in one of the following ways:

```
  DATA
1 A(1)/-5.772156649015328 63D-1/, A(2)/7.5416431366301662 0D-1/,
2 A(3)/ 1.29849232927373234D-1/, A(4)/2.40681355683977413D-2/,
3 A(5)/ 1.32084309209609371D-3/, A(6)/6.57739399753264501D-5/
```

or

```
  DATA        A(1)                   ,        A(2)
1      /-5.77215664901532863D-1 ,    7.54164313663016620D-1/,
2             A(3)                   ,        A(4)
3      /  1.29849232927373234D-1 ,   2.40681355683977413D-2/,
4             A(5)                   ,        A(6)
5      /  1.32084309209609371D-3 ,   6.57739399753264501D-5/
```

The latter example might well be broken into three separate data statements.)

(b) In the discussion of *Machine dependent features* it is noted, in particular, that references are made to the largest positive real number and (in effect) its natural logarithm. These references are buried in the code, at the statement numbered 80, 2 lines later, and 2 lines after the statement numbered 100. I feel that these should, at least, be defined by DATA statements at the head of the program. In fact, perhaps the time is now ripe for standard names and definitions of these and other environmental entities.

Index by Subject to Algorithms, 1970

[Algorithms published elsewhere are included, when known to us.]

A1		REAL ARITHMETIC, NUMBER THEORY	
A1	310	PRIME NUMBER GENERATOR 1	9-67(569),9-67(570),
A1	310	3-70(192)	
A1	371	PARTITIONS IN NATURAL ORDER	1-70(52)
A1	372	COMPLEX PRIMES	1-70(52),11-70(695)
A1	373	DOUBLY RESTRICTED PARTITIONS	2-70(120)
A1	374	RESTRICTED PARTITIONS GEN.	2-70(120)
A1	386	GCD OF N INTEGERS AND MLTPLRS	7-70(447)
A1	401	COMPLEX PRIMES	11-70(693),11-70(695)
A1		PARTITION FCNS.(MODULO D)	BIT 1969(83)
A1		OUTER PRODUCT OF SYMM. GROUP	BIT 1970(106)
A1		RATIONAL APPROX. OF REAL NO.	COMP.J.V11(347),
A1		V12(293)	
C2		ZEROS OF POLYNOMIALS	
C2		PL/I PROG FOR LEHMERS METHOD	MOC 23-69(829)
C5		ZEROS OF ONE OR MORE TRANSCENDENTAL EQUATIONS	
C5	378	SOLVE NONLINEAR SYSTEM OF EQS.	4-70(259)
C5		SOLN. OF NONLINEAR EQUATIONS	COMP.J.V12(406),
C5		V13(219)	
C6		SUMMATION OF SERIES, CONVERGENCE ACCELERATION	
C6	393	SUMMATION WITH ARB. PRECISION	9-70(570)
D1		QUADRATURE	
D1	233	MULTIPLE INTEG.-SIMPSONS RULE	6-64(348),8-70(512)
D1	331	GAUSSIAN QUADRATURE FORMULAS	6-68(432),5-69(280),
D1	331	8-70(512)	
D1	351	MODIFIED ROMBERG QUADRATURE	6-69(324),4-70(263),
D1	351	6-70(374),7-70(449)	
D1	353	FILON QUADRATURE	8-69(457),4-70(263)
D1	379	ADAPTIVE SIMPSON QUADRATURE	4-70(260)
D1	400	MODIFIED HAVIE INTEGRATON	10-70(622)
D1		MSQ RESPONSE ANALYSIS INTEGRAL	COMP.J.V13(207)
D1		CALC. OF GAUSS QUAD. RULES	MOC 23-69(221)
D2		ORDINARY DIFFERENTIAL EQUATIONS	
D2		1ST ORDER, AUTO. STEP CHANGE	NUM.MATH.V14(317)
D3		PARTIAL DIFFERENTIAL EQUATIONS	
D3	392	SYSTEMS OF HYPERBOLIC PDE	9-70(567)
D5		INTEGRAL EQUATIONS	
D5	368	INVERSION OF LAPLACE TRANSFORM	1-70(47),10-70(624)
E1		INTERPOLATION	
E1		SPLINE INTERPOLN OF DEGREE 3	COMP.J.V12(198),
E1		V12(409)	
E1		QUINTIC SPLINES INTERPOLATION	COMP.J.V12(292),
E1		V13(115)	
E2		CURVE AND SURFACE FITTING	
E2	375	FIT DATA TO A*EXP(-B*X)	2-70(120)
E2	376	FIT DATA TO A*COS(B*X+C)	2-70(121)
E4		MINIMIZING OR MAXIMIZING A FUNCTION	
E4	387	FCN. MINIMIZATION	8-70(509)
F1		MATRIX OPERATIONS, INCLUDING INVERSION	
F1	380	TRANSPOSE RECTANGULAR MTRX.	5-70(324),5-70(327)
F1		SINGULAR VALUE DECOMPOSITION	NUM.MATH.V14(403)
F2		EIGENVALUES AND EIGENVECTORS OF MATRICES	
F2	343	EIVALS-VECS.OF REAL GEN.MTRX	12-68(820),2-70(122),
F2	343	11-70(694)	
F2	384	EIVAL-EIVEC OF A SYMM. MTRX	6-70(369),12-70(750)
F2		EIVECS FOR GENL COMPLEX MTRX	MOC 22-68(785)
F2		QR FOR REAL HESSENBERG MTRX	NUM.MATH.V14(219)
F5		ORTHOGONALIZATION	
F5	127	ORTHONORMALIZATION	10-62(511),2-70(122)
G1		SIMPLE CALCULATIONS ON STATISTICAL DATA	
G1	359	FACTORIAL ANALYSIS OF VARIANCE	11-69(631),7-70(449)
G5		RANDOM NUMBER GENERATORS	
G5	369	POISSON RANDOM NUMBERS	1-70(49)
G5	370	RANDOM NUMBER GENERATOR	1-70(49)
G5	381	RANDOM VECTORS IN SOLID ANGLE	5-70(326)
G6		PERMUTATIONS AND COMBINATIONS	
G6	361	PERMANENT FNC.OF SQUARE MATRIX	11-69(634),6-70(376)
G6	382	COMBINATIONS OF M OUT OF N	6-70(368),6-70(376)
G6	383	PERMUTATIONS WITH REPETITIONS	6-70(368),6-70(376)
G7		SUBSET GENERATORS	
G7		A CLUSTERING ALGORITHM	COMP.J.V13(113)
G7		INDEXING SUBARRAYS	COMP.J.V13(208),
G7		V13(219)	
H		OPERATIONS RESEARCH, GRAPH STRUCTURES	
H	263	INTEGER PROGRAMMING-GOMORY1	10-65(601),5-70(326)
H	336	NETFLOW	9-68(631),3-70(192)
H	341	LINEAR PGMS.IN 0-1 VARIABLES	11-68(782),12-69(692),
H	341	4-70(263)	
H	394	DECISION TABLE TRANSLATION	9-70(571)
H	397	INTEGER PROGRAMMING PROBLEM	10-70(620)
H	399	SPANNING TREE	10-70(621)
H		MIN LENGTH FULL STEINER TREES	MOC 23-69(521)
M1		SORTING	
M1	201	SHELLSORT	8-63(445),6-64(349),
M1	201	6-70(373)	
M1	245	TREESORT 3	12-64(701),7-65(445),
M1	245	6-70(371)	
M1	347	SORT WITH MINIMAL STORAGE	3-69(185),1-70(54),
M1	347	10-70(624)	
M1	402	IMPROVED QUICKSORT	11-70(693)
M1		SORT BY RANKING ELEMENTS	COMP.J.V10(308),
M1		V12(408)	
M1		ORDER SUBSCRIPTS BY ELEMNT SIZE	COMP.J.V10(309),
M1		V12(408)	
M1		LISTED RADIX SORT	COMP.J.V12(406)
M1		INTERNAL SORT BY TWO-WAY MERGE	COMP.J.V13(110)
M1		MODIFIED DAVIDON FOR FCN. MIN.	COMP.J.V13(111)
R2		SYMBOL MANIPULATION	
R2	377	SYMB. EXPANSION OF EXPRESSIONS	3-70(191)
R2		FCN. VALUE FROM CHARAC. STRING	BIT 1969(283)
S		APPROXIMATION OF SPECIAL FUNCTIONS	
S		FUNCTIONS ARE CLASSIFIED S01 TO S22, FOLLOWING	
S		FLETCHER-MILLER-ROSENHEAD, INDEX OF MATH. TABLES	
S04		DIRICHLET L-SERIES	MOC 23-69(489)
S13	385	EXPONENTIAL INTEGRAL EI(X)	7-70(446),7-70(448),
S13	385	12-70(750)	
S14	321	T-TEST PROBABILITIES	2-68(115),2-70(124)
S14	344	STUDENT*S T-DISTRIBUTION	1-69(37),2-70(124),
S14	344	7-70(449)	
S14	395	STUDENTS T-DISTRIBUTION	10-70(617)
S14	396	STUDENTS T-QUANTILES	10-70(619)
S15	304	NORMAL CURVE INTEGRAL	6-67(374),6-67(377),
S15	304	4-68(271),10-69(565),10-70(624)	
S17	22	RICATTI-BESSEL FCNS. KIND 1,2	11-60(600),7-70(448)
S22	282	D/DX OF EXP(X)/X,COS(X)/X,	
S22	282	AND SIN(X)/X	4-69(272),1-70(53)
S22	292	REGULAR COULOMB WAVE FCNS.	11-66(793),5-69(278),
S22	292	5-69(280),9-70(573)	
S22	332	JACOBI POLYNOMIALS	6-68(436),7-70(449)
S22	352	CHAR.VALS,SLNS OF MATHIEU*S DE.	7-69(399),12-70(750)
S22	388	RADEMACHER FCN.	8-70(510)
S22	389	BINARY ORDERED WALSH FCNS.	8-70(511)
S22	390	SEQUENCY ORDERED WALSH FCNS.	8-70(511)
Z		ALL OTHERS	
Z	391	UNITARY SYMMETRIC POLYNOMIALS	8-70(512)
Z	398	TABLELESS DATE CONVERSION	10-70(621)
Z		COMP. OF GALOIS GROUP ELEMENTS	MOC 23-69(425)
Z		PLAY LEGAL CHESS	COMP.J.V13(208)

Key. The first column, A1, C2, C5, is the key to the classification system categories; second column is the number of the algorithm if in *CACM*, third column, algorithm title; fourth column, month, year, pages (in parentheses) in *CACM* or reference elsewhere.

This 1970 index is the second supplement to the "Index by Subject to Algorithms, 1960–1968" *Comm. ACM 11*, 12 (Dec. 1968), 827–830; the first supplement was published in *Comm. ACM 12*, 12 (Dec. 1969), 693.