DECUS
PROGRAM LIBRARY

DECUS NO. FOCAL® 8-217

TITLE HAMMING ALGORITHM TO SOLVE TWO COUPLED ORDINARY FIRST ORDER DIFFERENTIAL EQUATIONS WITH GIVEN INITIAL CONDITIONS

AUTHOR U. K. Shivadev

COMPANY Harvard University
Cambridge, Massachusetts

DATE February 24, 1972

SOURCE LANGUAGE FOCL.S (DECUS NO. FOCAL8-148)

HAMMING ALGORITHM TO SOLVE TWO COUPLED ORDINARY FIRST ORDER DIFFERENTIAL EQUATIONS WITH GIVEN INITIAL CONDITIONS

BACKGROUND:

Given a system of two coupled ordinary differential equations

$$\frac{dy}{dx} = f(x,y,M) \quad \text{and} \quad \frac{dM}{dx} = g(x,y,M) \qquad \ldots(1)$$

with given initial conditions, $y(x_o) = y_o$ and $M(x_o) = M_o$, the solutions in the form $y=y(x)$ and $M=M(x)$ are sought. The procedure given here is extracted from Ref(1). The main algorithm used is the Hamming's modified predictor-corrector algorithm. This method is a stable fourth order integration procedure that requires the evaluation of the right side of the system only twice per step. As a comparison, the Runge-Kutta method requires four such evaluations per step. Another advantage is the estimation of the local truncation error at each step. As such, this method indicates the point at which the step size can be altered for greater speed and accuracy depending on the local truncation error. The only drawback of this method is that it needs the first four starting values. In order to obtain these starting values, a special Runge-Kutta procedure, as suggested by Ralston is used.

Any such system of ordinary differential equations can be represented by

$$Y' = \frac{dY}{dx} = F(x,Y) \quad \text{with} \quad Y(x_o) = Y_o \qquad \ldots(2)$$

where Y', $F(x,Y)$ and Y_o are column vectors with the number of components equal to the number of equations and each component equation representing one differential equation. Supposing the results at the four equidistant points x_{j-3}, x_{j-2}, x_{j-1}, and x_j are known. Then the following procedure gives the results at x_{j+1}:

Predictor: $P_{j+1} = Y_{j-3} + (4h/3)(2Y'_j - Y'_{j-1} - 2Y'_{j-2})$

Modifier: $M_{j+1} = P_{j+1} - (112/121)(P_j - C_j)$

Corrector: $C_{j+1} = (9Y_j - Y_{j-2} + 3h(F(x_{j+1},M_{j+1}) + 2Y'_j - Y'_{j-1}))/8$

Final Value: $Y_{j+1} = C_{j+1} + (9/121)(P_{j+1} - C_{j+1})$

Truncation error: $t_e = (9/121) \sum_{i=1}^{\text{no. of eqns.}} w_i \left| P_{j+1,i} - C_{j+1,i} \right|$

where P,M and C are also column vectors and w_i's are the error weights to take care of the order of magnitude of the elements of Y.

During the first pass, the modifier is ignored because the values of P_j and C_j are yet to be evaluated. In case the truncation error, t_e, exceeds a prescribed value, then the step size is halved, four new values at x_j are obtained using the same starting procedure.

The special Runge-Kutta algotithm as proposed by Ralston has the following form. Starting at x_j the results at x_{j+1} are obtained by computing the following quantities:

$$K_1 = hY'_j$$

$$K_2 = hF(x_j+.4h, Y_j+.4K_1)$$

$$K_3 = hF(x_j+.4557373h, Y_j+.2969776K_1+.1587596K_2)$$

$$K_4 = hF(x_j+h, Y_j+.218003K_1-3.050965K_2+3.832865K_3)$$

$$Y_{j+1} = Y_j+.1747603K_1-.5514807K_2+1.205536K_3+.1711848K_4$$

where K_1, K_2, K_3, and K_4 are also column vectors.

PROGRAM DETAILS:

The program listing included is capable of executing the above procedure in FOCAL-S 11/70 language for solving only two coupled equations. The listing includes some explanatory comments which may be deleted to conserve core storage area. All the variables in the program are global. So, the variables used in any auxiliary programs must be different from those already used. A list of variable names is given in Table 1

Illustrative example:

A trivial problem is solved using the above program for the purposes of illustration:

$$\frac{dy}{dx} = -y \quad \& \quad \frac{dM}{dx} = -M \quad \text{with} \quad y(0)=1 \text{ and } M(0)=1$$

The exact solutions are $y=M=\exp(-x)$

The user needs to type in the additional steps in parts 1, 3, 7, 8, 15 and 20 as shown in the example listing:

Auxiliary Listing

```
C-FOCAL S 11/70

01.70 A "XM"XM,"X0"X0,"Y0"Y0,"M0"M0,"RR"RR,"H"H,"ST"ST
01.80 A "WY"WY,"WM"WM,"TP"TP

03.20 T !!"XM=",XM," X0=",X0," Y0=",Y0," M0=",M0," RR=",RR,!
03.30 T " H=",H," ST=",ST," WY=",WY," WM=",WM," TP=",TP

07.15 T "      X          Y          M          TE       EXP(-X)",!
07.20 T %7.4,X0,"  ",Y0,"  ",M0,"  ",TE,"  ",FEXP(-X0),!

08.20 T %7.4,X[4],"  ",Y[4],"  ",M[4],"  ",TE,"  ",FEXP(-X[4]),!

15.20 S YX=-Y

20.20 S MX=-M
*
```

Sample Output Printout

```
*G
XM:.5 X0:0 Y0:1 M0:1 RR:.005 H:.01 ST:.1 WY:1 WM:1 TP:0

XM= 0.500000 X0= 0.000000 Y0= 1.000000 M0= 1.000000 RR= 0.005000
 H= 0.010000 ST= 0.100000 WY= 1.000000 WM= 1.000000 TP= 0.000000

     X          Y          M          TE       EXP(-X)
 0.000000   1.000000   1.000000   0.000000   1.000000
 0.100000   0.904836   0.904836   0.000000   0.904837
 0.200000   0.818728   0.818728   0.000000   0.818731
 0.300000   0.740814   0.740814   0.000000   0.740818
 0.400000   0.670315   0.670315   0.000000   0.670320
 0.500000   0.606525   0.606525   0.000000   0.606531
END

*G
XM:2 X0:0 Y0:1 M0:1 RR:.005 H:2 ST:.1 WY:1 WM:1 TP:1

XM= 2.000000 X0= 0.000000 Y0= 1.000000 M0= 1.000000 RR= 0.005000
 H= 2.000000 ST= 0.100000 WY= 1.000000 WM= 1.000000 TP= 1.000000

     X          Y          M          TE       EXP(-X)
 0.000000   1.000000   1.000000   0.000000   1.000000
H= 0.100000E+01
H= 0.500000E+00
 2.000000   0.135035   0.135035   0.000567   0.135335
END

*
```

This illustrates the automatic reduction in step size H, because of initially high truncation error.

REFERENCES:

1. System/360 Scientific Subroutine Package, Version III Published by IBM, 1968.
2. Wrege, D.E., 'FOCL.S, an expanded language for small computers, based on FOCAL, Decus Program Library No. FOCAL8-148.

TABLE 1

*T %,$	
Q1(00) 0.138560E+00	
P1(00) 0.138560E+00	
D@(00) 0.819100E-01	
C@(00) 0.819100E-01	
T@(00) 0.812352E-01	
S@(00) 0.812352E-01	
Q@(00) 0.847597E-01	
P@(00) 0.847597E-01	
MP(04)-0.821219E-01	
YP(04)-0.821219E-01	
M@(04) 0.821219E-01	
Y@(04) 0.821219E-01	NOTE: THE RATIO OF ST TO H MUST BE
X@(04) 0.250000E+01	AN EVEN NUMBER FOR PRINTOUT
MP(03)-0.135035E+00	TO OCCUR CORRECTLY.
YP(03)-0.135035E+00	
M@(03) 0.135035E+00	
Y@(03) 0.135035E+00	
X@(03) 0.200000E+01	
MP(02)-0.223395E+00	
YP(02)-0.223395E+00	
M@(02) 0.223395E+00	
Y@(02) 0.223395E+00	
MK(00) 0.223395E+00	
YK(00) 0.223395E+00	
L4(00)-0.110667E+00	
K4(00)-0.110667E+00	
L3(00)-0.145061E+00	
K3(00)-0.145061E+00	
L2(00)-0.147268E+00	
K2(00)-0.147268E+00	
L1(00)-0.184085E+00	
K1(00)-0.184085E+00	
XX(00) 0.000000E+00	
MM(00) 0.368171E+00	
YY(00) 0.368171E+00	
X@(02) 0.150000E+01	
I@(00) 0.500000E+01	
TE(00) 0.423931E-03	...Truncation Error
MP(01)-0.368171E+00	
YP(01)-0.368171E+00	
MX(00)-0.821219E-01	
YX(00)-0.821219E-01	
M@(00) 0.821219E-01	
Y@(00) 0.821219E-01	
X@(00) 0.250000E+01	
M@(01) 0.368171E+00	
Y@(01) 0.368171E+00	
X@(01) 0.100000E+01	
PN(00) 0.100000E+01	
N@(00) 0.500000E+01	
TP(00) 0.100000E+01	...If TP=1, then the output is printed out at every H provided H is halved. Otherwise it is ineffectual
WM(00) 0.100000E+01	
WY(00) 0.100000E+01	
ST(00) 0.100000E+00	...output printout interval size.
H@(00) 0.500000E+00	...step size
RR(00) 0.500000E-02	
MØ(00) 0.100000E+01	
YØ(00) 0.100000E+01	
XØ(00) 0.000000E+00	
XM(00) 0.200000E+01	
Z@(00) 0.371200E+04	
*	

```
C-FOCAL S 11/70

01.10 C HAMMING'S ALGORITHM TO SOLVE FOR Y=Y(X), M=M(X) GIVEN
01.15 C Y'=F(X,Y,M) AND M'=G(X,Y,M)
01.20 C INPUT THE FOLLOWING: MAXIMUM X=XM
01.30 C INITIAL X=X0, INITIAL Y=Y0, INITIAL M=M0,
01.40 C TRUNCATION ERROR LIMIT=RR, STEP SIZE=H, INTERVAL SIZE=ST,
01.50 C WEIGHTING FACTORS FOR THE CUMULATIVE ERROR, WY AND WM
01.55 C AND PRINTOUT SELECTOR=TP
01.60 C COMPLETE PARTS 3,7,8,15 AND 20

03.10 C INPUT DISPLAY PROGRAM

05.01 C HAMMING ALGORITHM:
05.05 T !!;S N=3;S PN=0;S X[1]=X0;S Y[1]=Y0;S M[1]=M0;S X=X[1][↑K
      S Y=Y[1];S M=M[1];D 15;D 20;S YP[1]=YX;S MP[1]=MX;S TE=0
05.08 D 7
05.11 F I=2,4;S X[I]=X[I-1]+H;S YY=Y[I-1];S MM=M[I-1];D 10[↑K
      S Y[I]=YK;S M[I]=MK;S Y=YK;S M=MK;D 15;D 20[↑K
      S YP[I]=YX;S MP[I]=MX
05.14 S P=Y[1]+4*H*(2*(YP[4]+YP[2])-YP[3])/3[↑K
      S Q=M[1]+4*H*(2*(MP[4]+MP[2])-MP[3])/3
05.17 S S=P;S T=Q
05.20 S X=X[4]+H;S Y=S;S M=T;D 15;D 20
05.23 S C=(9*Y[4]+3*H*(YX+2*YP[4]-YP[3])-Y[2])/8[↑K
      S D=(9*M[4]+3*H*(MX+2*MP[4]-MP[3])-M[2])/8
05.26 S TE=9*((WY*FABS(P-C))+(WM*FABS(Q-D)))/121
05.29 I (RR-TE) 6.05
05.32 S N=N+1;F I=2,4;S X[I-1]=X[I];S Y[I-1]=Y[I];S YP[I-1]=YP[I][↑K
      S M[I-1]=M[I];S MP[I-1]=MP[I]
05.35 S X[4]=X[4]+H;S Y[4]=C+9*(P-C)/121;S M[4]=D+9*(Q-D)/121[↑K
      S X=X[4];S Y=Y[4];S M=M[4];D 15;D 20;S YP[4]=YX;S MP[4]=MX
05.44 I (XM-X[4]) 5.99;I (PN-1) 5.47,5.5,5.47
05.47 I (.00001+FITR(N*H/ST)-N*H/ST) 5.62
05.50 D 8
05.62 S P1=P;S Q1=Q;D 5.14;S S=P-112*(P1-C)/121;S T=Q-112*(Q1-D)/121
05.65 G 5.2
05.99 T "END"!!;Q

06.05 S H=H/2;S N=3;S PN=TP;T "H=",%,H,!;G 5.11

07.10 C INITIAL VALUES DISPLAY PROGRAM: X0, Y0, M0

08.10 C OUTPUT DISPLAY PROGRAM: X[4], Y[4], M[4]

10.01 C RALSTON'S FOURTH-ORDER ALGORITHM:
10.05 S X=XX;S Y=YY;S M=MM;D 15;D 20;S K1=H*YX;S L1=H*MX
10.10 S X=XX+.4*H;S Y=YY+.4*K1;S M=MM+.4*L1;D 15;D 20[↑K
      S K2=H*YX;S L2=H*MX
10.15 S X=XX+.4557373*H;S Y=YY+.2969776*K1+.1587596*K2[↑K
      S M=MM+.2969776*L1+.1587596*L2;D 15;D 20;S K3=H*YX;S L3=H*MX
10.20 S X=XX+H[↑K
      S Y=YY+.2181004*K1-3.050965*K2+3.8328648*K3[↑K
      S M=MM+.2181004*L1-3.050965*L2+3.8328648*L3[↑K
      D 15;D 20;S K4=H*YX;S L4=H*MX
10.25 S YK=YY+.1747603*K1-.5514807*K2+1.205536*K3+.1711848*K4[↑K
      S MK=MM+.1747603*L1-.5514807*L2+1.205536*L3+.1711848*L4;R

15.10 C EVALUATION OF YX=F(X,Y,M)

20.10 C EVALUATION OF MX=G(X,Y,M)
*
```