الجمهورية العربية المتحدة

معهد التخطيط القومي

Memo No. 594

Fortran Subroutine For Simpson's Method of Numerical Integration

By
Khairy A. Sedrac
Supervised by
Dr. S. Hamid.

القاهرة
٣ شارع محمد مظهر - بالزمالك

Part I

The Case of One Integrand

The following Fortran subrogram is to carry on numerically the $\int_a^b f(x)\,dx$ using Simpson's one-third rule of integration. If the function f(x) is computed at the n+1 points;

$a, a + \frac{b-a}{n}, a + 2\frac{b-a}{n}, a + 3\frac{b-a}{n}, \ldots\ldots, b$ i.e. the points

$a, a + h, a + 2h, a + 3h, \ldots\ldots, b$

Where $h = \frac{b-a}{n}$

Then Simpson* one-third rule of integration states that:

$$\int_a^b f(x)\,dx = \frac{h}{3}\Big[f(a)+4f(a+h) + 2f(a+2h) + \ldots + 4f(a+(n-1)h) + f(a+nh)\Big] + E = S_n + E$$

Where E the truncation error is given by $E = \frac{-n}{180} h^5 f^{iv}(\xi)$

Where $f^{iv}(\xi)$ is the fourth derivative of f(x) at ; $a \leq \xi \leq b$

Our Fortran subroutine starts with given intial interval n and compute S_n and then repeat the computation with doubling the interval to get S_{2n}. From S_{2n} & S_n we get for the trunction error

$$E_n = \frac{16\,S_{2n} - S_n}{15} = S_{2n} + \frac{S_{2n} - S_n}{15}$$

The truncation error is compared with a pregiven value of tolerance (which may be absolute or relative). If it exceeds that tolerance, the program is repeated with doubling the intervals to get S_{4n}, E_{2n} and so on

The program gives as output the final number of intervals used and the value of integration (after allowing for correction term).

---

* See for example. ZDENEK KOPAL, Numerical Analysis, Page 405

Description of the Subroutine:

Name of the subroutine: SIMPS (A,B,N,EPS, S)

Input of the Subroutine:

A : the lower limit of the integration

B : the upper limit of the integration

N : number of intial intervals

EPS : absolute tolerance or relative with respect to the value of Integration.

Output of the Subroutine:

N : number of the final intervals of the integration

S : the value of the integration.

Subprogram Called by the Subroutine:

The subroutine calls the function subprogram F (u) where F (u) is the mathematical expression for the integrand and u the argument of the integration.

Description of the Function Subprogram F (u) :

The Fortran program for the function subprogram take the following form.

```
Function   F(u)
    .
    .
    .
    .
  F =
RETURN
  END
```

The statements following the first statement may be control, Common as well as arithematic statements to carry on the necessary steps to compute the function given by the integrand.

In the subprogram one should allow for such cases:

1. The absolute value of the integrand is less than the underflow limit then the integrand is replaced by zero
2. The integrand takesan indefinite value for a given value or values of the argument then it is replaced its mathematical limit.

Description of the main:

In the main program we carry on:

1. Read and / or compute the limits of the integration
2. Read and / or compute the parameters of the integration

The parameters of the integration may appear in the integrand. For generality to avoid in cluding it in the subroutine we put it as common variable between the Main program and the Function subprogram by a common statements.

3. To call the subroutine
4. To print and / or punch the results.

Part II

The case of many Integrands

In practical application of the above subprogram we had been met with the situation where the problem under consideration is to carry different integration. The same situation appears also when we are solving numerical integration of different problems. It is obvious that such situation can be met by reproducing the program several time each time corresponding to a separate integrand. It may be more convenient, however to include all these different integrands in one program through introduction of an integer parameter in the definition of the Function subrprograms which define, the integrands. That means that the subprogram functions F(u) will be replaced by the subprogram function F(L,u) where L an integer parameter defining the order of integrand. For example if we have three integrations to be carried, then

L = 1,2,3 and so on

In the computation of the integrand in the subprogram we have to use the Fortran statement "Computed Go To" to choose the proper function corresponding to the required integrand.

Description of the subroutine:

Name of the subroutine: SIMPS (A,B,N,L,EPS,S)

where A,B,N,EPS,S are as in the case of one integrand and L an input integer parameter defining the order of the integrand considered.

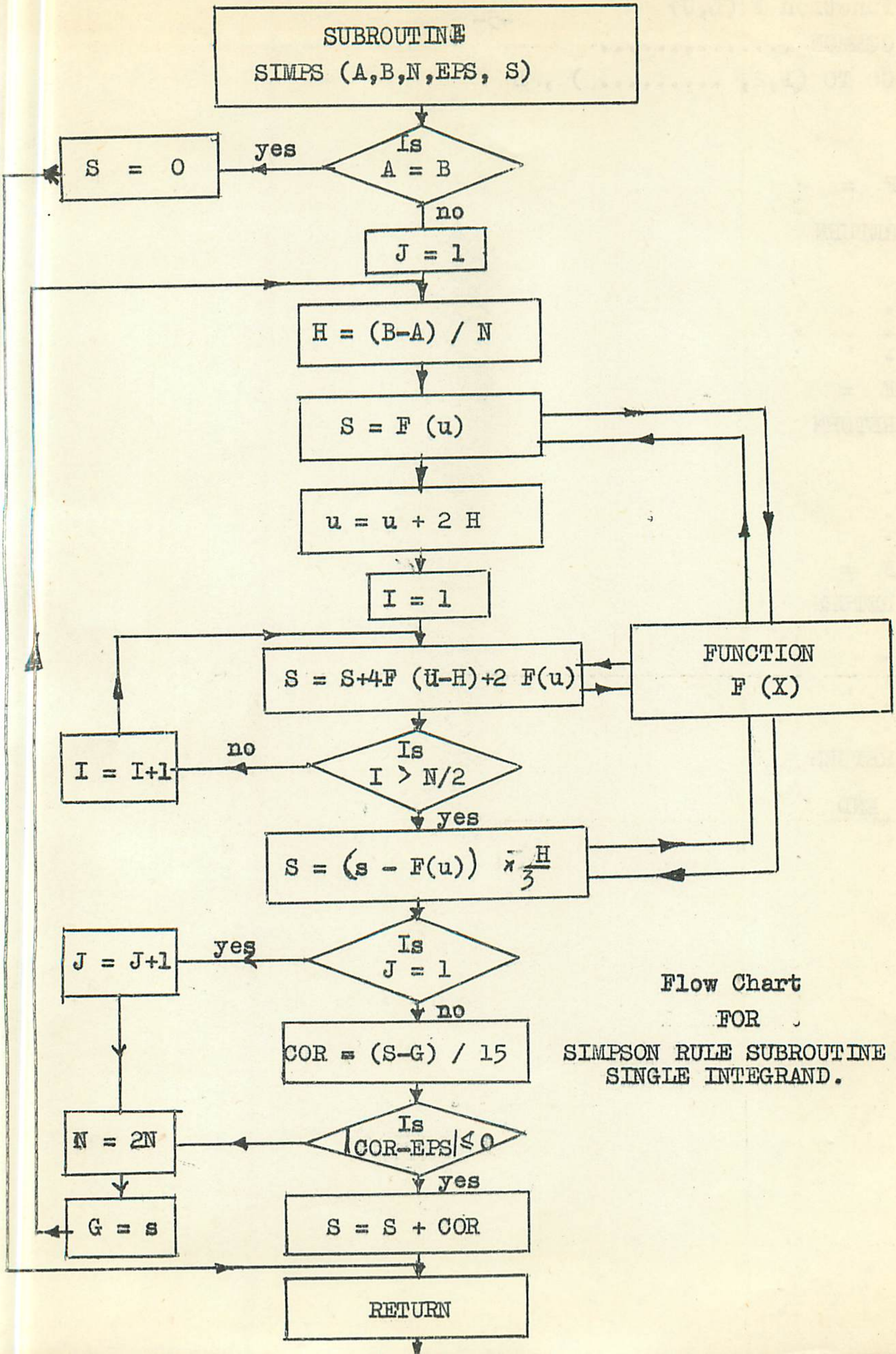
Subrogram Called by the subroutine

It calls the function subprogram F (L,U) where U the argument of the integration andLthe integer parameter defined above.

Description of the Function Subprogram

The Function subprogram takes the form:

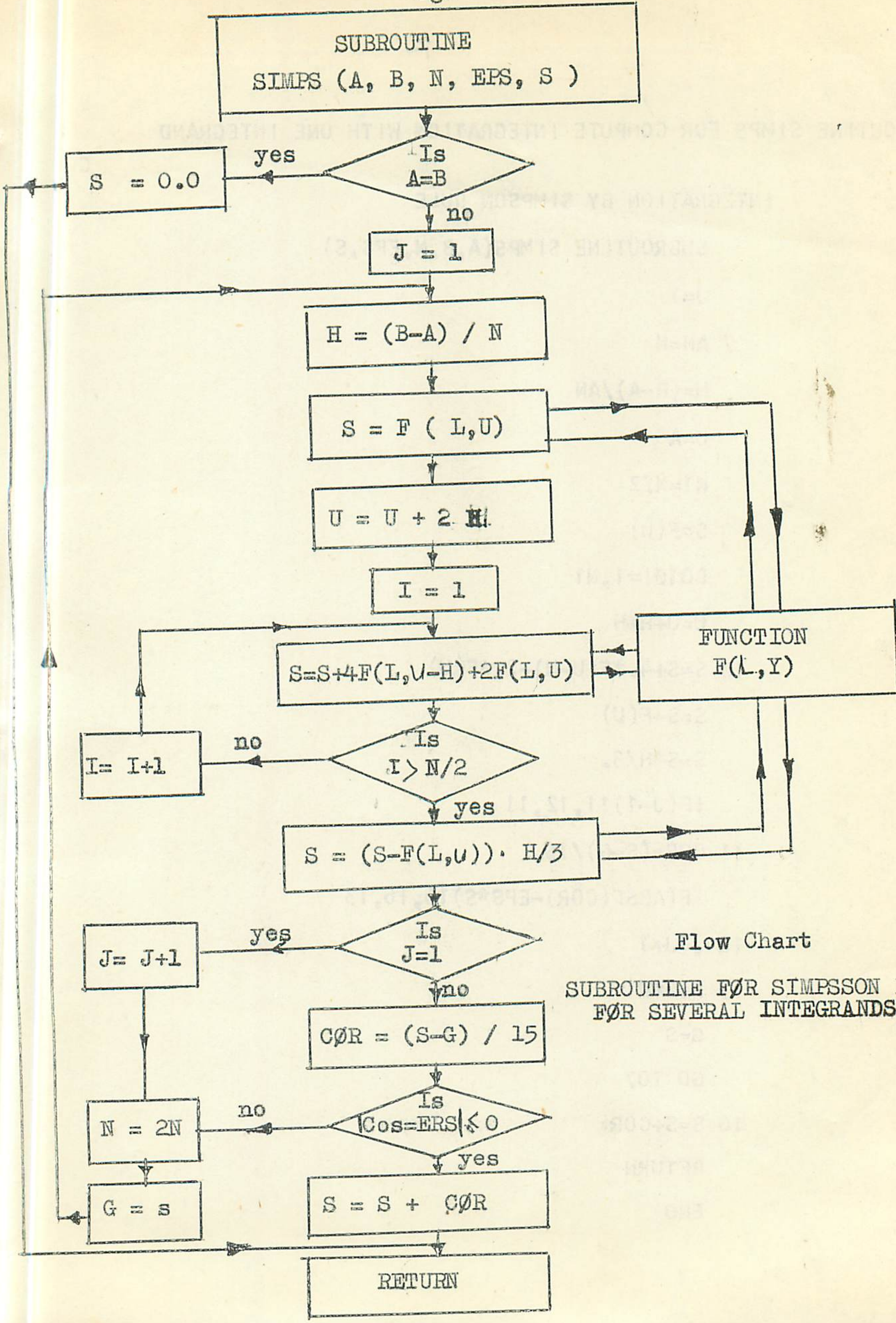
```
      Function F (L,U)
      COMMON ..............
      GO TO (1,2, ......... ) , L

1
      F =
      RETURN

2     .
      .
      .
      F =
      RETURN

3     .
      .
      .
      F =
      RETURN
      .
      .
      .
      .
      RETURN
       END
```



Flow Chart
FOR
SIMPSON RULE SUBROUTINE FOR
SINGLE INTEGRAND.

SUBROUTINE SIMPS FOR COMPUTE INTEGRATION WITH ONE INTEGRAND

```
C      INTEGRATION BY SIMPSON RULE
       SUBROUTINE SIMPS(A,B,N,EPS,S)
       J=1
     7 AN=N
       H=(B-A)/AN
       U=A
       N1=N/2
       S=F(U)
       DO10I=1,N1
       U=U+H+H
    10 S=S+4.*F(U-H)+2.*F(U)
       S=S-F(U)
       S=S*H/3.
       IF(J-1)11,12,11
    11 COR=(S-G)/15.
       IF(ABSF(COR)-EPS*S)16,16,13
    12 J=J+1
    13 N=2*N
       G=S
       GO TO7
    16 S=S+COR
       RETURN
       END
```



Flow Chart

SUBROUTINE FØR SIMPSSON RULE
FØR SEVERAL INTEGRANDS.

```
C     SUBROUTINE SIMPS FOR COMPUTE INTEGRATIONS WITH MANY INTEGRANDS

C     INTEGRATION BY SIMPSON RULE
      SUBROUTINE SIMPS(A,B,N,L,EPS,S)
      J=1
    7 AN=N
      H=(B-A)/AN
      U=A
      N1=N/2
      S=F(L,U)
      DO10I=1,N1
      U=U+H+H
   10 S=S+4.*F(L,U-H)+2.*F(L,U)
      S=S-F(L,U)
      S=S*H/3.
      IF(J-1)11,12,11
   11 COR=(S-G)/15.
      IF(ABSF(COR)-EPS*S)16,16,13
   12 J=J+1
   13 N=2*N
      G=S
      GO TO7
   16 S=S+COR
      RETURN
      END
```

Example 1: Case of Single Integrand

$$\text{To Compute INTG} = \int_0^{\pi} \text{Sin}(5x)\,dx$$

```
C     MAIN PROGRAM
      ------------
C     INTEGRATION OF F(X) FROM A1 TO A2 BY SIMPSON RULE
    1 READ2,A,B,EPS,M
    2 FORMAT(2X,2F10.4,E13.6,I3)
      CALL SIMPS(A,B,M,EPS,S)
      PRINT5,A1,A2,EPS,M,S
    5 FORMAT(15X10HTHE RESULT/2X,3HA1=F10.4,3X,3HA2=F10.4,3X,
     X4HEPS=E13.6,3X,2HM=I5/5X,5HINTG=E13.6)
      GO TO1
      END
      ---------------------

C     FUNCTION SUBPROGRAM
      -------------------
      FUNCTION F(X)
      F=SINF(5.*X)
      RETURN
      END
      ---------------------
```

THE RESULTS

```
A1=  0.000  A2=  3.1416  EPS= 1.000000E-05  M=96
INTG=3.999999E-01

EXACT SOLUTION   INTEG=0.4
```

Example 2: Case of Several integrands:*

To compute:

$$S(1) = \int_0^\infty \left[e^{\left(\frac{1}{y^3} - \frac{1}{y^9}\right)} - 1 \right] dy$$

$$S(2) = \int_0^\infty \left[e^{\left(\frac{1}{y^3} - \frac{1}{y^9}\right)} - 1 \right] dy$$

$$S(3) = \int_0^1 e^{2\nu\left[\left(\frac{\sigma}{y}\right)^3 - \left(\frac{\sigma}{y}\right)^9\right]} dy$$

$$S(4) = \int_0^1 e^{2\nu\left[\left(\frac{\sigma}{y}\right)^3 - \left(\frac{\sigma}{y}\right)^9\right]} y\,dy$$

$$S(5) = \int_0^1 e^{2\nu\left[(\sigma y)^3 - (\sigma y)^9\right]} \left(y - \frac{1}{15} y^7\right) dy$$

where σ , ν has many values.

we can write first and second integration in the form:

$$S(1) = S_1(1) + S_2(1)$$
$$S(2) = S_1(2) + S_2(2)$$

where

$$S_1(1) = \int_0^1 \left[e^{\nu\left(\frac{1}{y^3} - \frac{1}{y^9}\right)} - 1 \right] dy$$

$$S_2(2) = \int_0^1 \left[e^{\nu\left(\frac{1}{y^3} - \frac{1}{y^9}\right)} - 1 \right] \frac{dy}{y^2}$$

We shall use in the programs the symbol GN for ν , DN for 2ν and SGM for σ .

*The above example has been referred to us by Dr. M. EL-Bakry Applied Mathematic Departement - Faculty of science - Cairo University.

MAIN PROGRAM

```
C       PROGRAM 1 FOR  DR. BAKRY PROBLEM@COMPUTE THE INTGRATIONS
        DIMENSION S(7)
        COMMON GN,SGM,DN
      1 READ5,EPS,N,MØ,A
      5 FORMAT(E12.6,2I3,F4.1)
        READ6,SGM
      6 FORMAT(F6.4)
        PRINT7,SGM
        PUNCH7,SGM
        PRINT8
      7 FORMAT(1ØX,4HSGM=F8.4)
      8 FORMAT(4X,2HNU,9X,4HS(1),9X,4HS(2),9X,4HS(3),9X,4HS(4),9X,4HS
(5))
        DO2Ø J=1,N
        READ 9,TH,GN,DN
      9 FORMAT(3F8.4)
        SQRT=(2.*GN/2ØØ.)**(1./9.)
        DO 1Ø I=1,7
        GO TO(11,12,13,13,14,14,14),I
     11 B=SQRT/(2.)**(1./9.)
        GO TO 18
     12 B=SQRT
        GO TO 18
     13 B=SGM*SQRT
        GO TO 18
     14 B=Ø.Ø
     18 M=MØ
        CALL SIMPS(B,A,M,5,EPS,T)
     1Ø S(I)=T
        S(1)=S(1)+S(6)
        S(2)=S(2)+S(7)
        IF(SENSE SWITCH1)15,16
     15 PRINT 17,GN,(S(I),I=1,5)
     16 PUNCH 22,TH,GN,(S(I),I=1,5)
     17 FORMAT(F8.4,5(2X,E12.6))
     22 FORMAT(2F7.3,2E11.5,3E12.6)
     2Ø CONTINUE
        GO TO 1
        END
```

```
C     FUNCTION SUBPROGRAM
C     PROGRAM 3 FOR  DR. BAKRY PROBLEM
C     SUBPROGRAM FOR CALCULATION OF INTGRAND FUNCTIONSOF INTEG-
RATIONS
      FUNCTION F(I,Y)
      COMMON GN,SGM,DN
      GO TO (1,2,3,4,5,6,7),I
    1 RY=1./Y
      F=EXPF(GN*(RY**3-RY**9))
      RETURN
    2 RY=1./Y
      F=EXPF(DN*(RY**3-RY**9))
      RETURN
    3 RSY=SGM/Y
      F=EXPF(DN*(RSY**3-RSY**9))
      RETURN
    4 RSY=SGM/Y
      F=EXPF(DN*(RSY**3-RSY**9))*Y
      RETURN
    5 SY=SGM*Y
      F=(Y-1./15.*Y**7)*EXPF(DN*(SY**3-SY**9))
      RETURN
    6 IF(Y)9,8,9
    8 F=-1.
      RETURN
    9 F=(EXPF(GN*(Y**3-Y**9))-1.)/Y**2-1.
      RETURN
    7 IF(Y)11,1Ø,11
   1Ø F=-1.
      RETURN
   11 F=(EXPF(DN*(Y**3-Y**9))-1.)/(Y*Y)-1.
      RETURN
      END
```

THE RESULTS

SGM= .958Ø

NU	S(1)	S(2)	S(3)	S(4)	S(5)
8.410	.1ØØ99E+Ø2	.17461E+Ø3	.4Ø6362E-ØØ	.4Ø1362E-ØØ	.925459E+Ø2
5.607	.34428E+Ø1	.26228E+Ø2	.185891E-ØØ	.1828Ø5E-ØØ	.133458E+Ø2
4.205	.1689ØE+Ø1	.1ØØ99E+Ø2	.133445E-ØØ	.13Ø742E-ØØ	.528895E+Ø1
3.364	.92Ø78E-ØØ	.54561E+Ø1	.113Ø86E-ØØ	.11Ø434E-ØØ	.31Ø354E+Ø1
2.803	.49629E-ØØ	.34428E+Ø1	.1Ø3422E-ØØ	.1ØØ7Ø1E-ØØ	.22Ø217E+Ø1
2.403	.22965E-ØØ	.23571E+Ø1	.9843Ø6E-Ø1	.955868E-Ø1	.145813E+Ø1
2.103	.47249E-Ø1	.1689ØE+Ø1	.958321E-Ø1	.928365E-Ø1	.173567E+Ø1
1.869	.85325E-Ø1	.1241ØE+Ø1	.945941E-Ø1	.9143Ø8E-Ø1	.1277Ø8E+Ø1

SGM= .729Ø

NU	S(1)	S(2)	S(3)	S(4)	S(5)
9.551	.14973E+Ø2	.385Ø4E+Ø3	.231265E+Ø3	.2Ø5858E+Ø3	.366Ø95E+Ø2
10.750	.22448E+Ø2	.8929ØE+Ø3	.55613ØE+Ø3	.495112E+Ø3	.717886E+Ø2
12.280	.37482E+Ø2	.26699E+Ø4	.172337E+Ø4	.153423E+Ø4	.173578E+Ø3
14.330	.74688E+Ø2	.11713E+Ø5	.782Ø75E+Ø4	.696Ø23E+Ø4	.576284E+Ø3
17.190	.19835E+Ø3	.95738E+Ø5	.659Ø98E+Ø5	.586149E+Ø5	.32Ø967E+Ø4
21.490	.8929ØE+Ø3	.22999E+Ø7	.16249ØE+Ø7	.144311E+Ø7	.441143E+Ø5
28.660	.12Ø13E+Ø5	.51591E+Ø9	.351557E+Ø9	.311498E+Ø9	.382496E+Ø7
42.980	.23956E+Ø7	.26325E+14	.176816E+14	.156134E+14	.35625ØE+11