MATHEMATICS NOTES

NOTE 7

OCTOBER, 1969

Accuracy Check on Subroutine BESSEL for Bessel Functions of the First and Fourth Kind for Various Orders and Arguments

A1C Richard T. Clark

Air Force Weapons Laboratory

Abstract

After various changes to the BESSEL subroutine are described, an accuracy check is made on the subroutine for Bessel Functions of the first and fourth kind.

INTRODUCTION

Various changes in subroutine BESSEL[1] were necessary because recent changes were made in the CDC 6600 compiler, in order to allow the argument to have a phase value of π, and in order to eliminate a few inaccuracies. BESSEL is used to compute tables of Bessel Functions of the first and fourth kind for phase values of 0, $\pi/4$, $\pi/2$, $3\pi/4$, π, $5\pi/4$, $3\pi/2$, and $7\pi/4$, for orders 0, and 1, and for arguments(x).1 to 100. These tables are compared to listings found directly in reference material or to listings computed from intermediate results. In any case the listings are considered to be tabulated and must be capable of being expressed in terms of functions which are in reference material so a comparison and accuracy check can be made. We then have a comparison of tables, one computed and the other tabulated. Formulas for relating the functions for different phase values to tabulated ber, bei, ker, kei, K(x), or I(x) had to be derived in many cases and the necessary relations and expressions were all found in Reference 2. Especially in the case of the eight Bessel Functions of the first kind it was feasible to use the same tables for various phase values since only a sign difference was indicated by the formula relation. A check is made for the derivative of the Bessel Function of the first kind for order zero by a comparison to the Bessel Function of the first order. The accuracy is extremely good and the variation of accuracy due to intermediate calculations is discussed.

CHANGES IN SUBROUTINE BESSEL

In the original version of Mathematics Note 1, BESSEL had arguments in its formal parameter list which also appeared in an EQUIVALENCE statement. Recent changes to the CDC 6600 system at AFWL do not permit this usage. Consequently, many of the parameters were switched to COMMON statements in order to provide for proper communication between BESSEL and its calling routine. Previously the calling sequence for BESSEL was

```
CALL BESSEL(N,Z,J,Y,H2,JPRIME,YPRIME,H2PRIME,IVALCHK,IPRINT)
```

The parameters J,Y,H2,JPRIME,YPRIME,H2PRIME,IVALCHK,IPRINT all appeared in an EQUIVALENCE statement which equivalenced them to other variables within the subroutine. The calling sequence was changed to

```
CALL BESSEL(N,Z)
```

and the arguments deleted from the parameter list were placed in the labeled COMMON statement

```
COMMON/ARGBESS/J,Y,H2,JPRIME,YPRIME,H2PRIME,IVALCHK,IPRINT
```

The same COMMON block must now appear in the calling routine to achieve proper communication between the routines. A similar change occurs in the calling of subroutine CBESS from BESSEL. Previously the calling sequence was

```
CALL CBESS(Z,JZ,J1,YZ,Y1,H2Z,H21,IVALCHK,IPRINT)
```

The parameters JZ,J1,YZ,H2Z,H21,IVALCHK, and IPRINT all appear in an EQUIVALENCE statement. In the original paper, Math Note 1, the variable Y1 was not included in the EQUIVALENCE statement, yet it is not used by CBESS. It was determined that it should have been equivalenced to the variable YONE. In the current version of CBESS the calling sequence is

```
CALL CBESS(Z,YONE,IVALCHK,IPRINT)
```

The arguments now missing from the original list are in the COMMON statement

```
COMMON/ARGCBES/JZ,J1,H2Z,H21
```

IVALCHK and IPRINT remain in the parameter list. To achieve this, the variables were deleted from the equivalence statement where they appeared as below

```
EQUIVALENCE ...   ,(IVALCHK,CHECK),(IPRINT,PRINT)
INTEGER PRINT,CHECK
```

The INTEGER statement above was also deleted, and where PRINT and CHECK appeared in the subroutine they were replaced by IPRINT and IVALCHK. To achieve proper communication between CBESS and BESSEL the COMMON statement which contains those parameters deleted from the argument list when BESSEL calls CBESS must be included in BESSEL. So, now one has two COMMON statements in subroutine BESSEL, one to communicate with the main program and the other for communication with CBESS.

As explained in Mathematics Note 1, the algorithm to compute the functions of order 0 and 1 is different for values of $|Z| \leq 6.0$ from that of $|Z| > 6.0$. It was found that the accuracy of the functions drops at this point by about five significant digits. The accuracy improves gradually, however, until at $|Z| \geq 10.0$ it is back to its original accuracy. In order to attempt to correct this loss of accuracy at the switchover point of the algorithms it was decided to try changing the switchover point to different values. The first algorithm was used for increasing arguments until it was noted that the numbers produced by this method matched those of the second algorithm in accuracy. This point was at Z=10.0, so now the first algorithm is used to this point and the second algorithm is used thereafter. The original card in CBESS was

```
IF(CABS(Z)-6.)1,1,4
```

and the new card is

```
IF(CABS(Z)-10.)1,1,4
```

Another change in CBESS was made after discovering that the variable JZERO goes to 0+i0 in certain instances, namely when Z=2+i0 and Z=-2+i0, causing an arithmetic error when JZERO is used as a divisor. Previous to this division the following check is made

```
IF(JZERO.EQ.(0.,0.))JZERO=CMPLX(1.E-15,0.)
```

This will change an exact zero to a very small number, introducing a minute error in the computations, but avoiding a computer dump in the process.

The BESSEL subroutine was not designed to handle a phase of π so a few changes were made in order for the subroutine to accept it.

$$H_\nu^{(2)}(xe^{\pi i}) = J_\nu(xe^{\pi i})-iY_\nu(xe^{\pi i}) \qquad \nu=0,1$$

by definition of the Hankel Function and it is known that

$$J_\nu(x) = (-1)^\nu J_\nu(xe^{\pi i})$$

and that

$$Y_\nu(x) + 2(-1)^\nu J_\nu(x)\, i = Y_\nu(xe^{\pi i})$$

so the Hankel with a phase of π can be calculated from the Hankel with a phase of 0.

The following 11 cards were added in CBESS in order to equip the subroutine for the phase value of π:

```
4    K=0
     IF(ABS(AIMAG(Z)).LT. 1.E-9.AND.REAL(Z).LT.0.) GO TO 5
     GO TO 6
5    Z=-Z
     K=1
              .
              .
              .
     IF(K.EQ.O) GO TO 12
     Z=-Z
     YZERO=YZERO+2.*JZERO*(0.,1.)
     H2ZERO=JZERO-(0.,1.)*YZERO
     JONE=-JONE
     YONE=-YONE+2.*JONE*(0.,1.)
     H2ONE=JONE-(0.,1.)*YONE
```

A card had to be added to FUNCTION ARG in order to eliminate round-off error. The manner in which the main program calls the BESSEL subroutine determines the accumulation of the round-off error. The new card is

```
IF(Y*Y.LT.1.E-12)Y=0.
```

It was also noticed that the value for π in that subroutine was incorrectly given. The correct value is used throughout this note.

The final change made in BESSEL was to remove from the function FUNCTION ZERO the following statements

```
EQUIVALENCE(IPRINT,PRINT)
INTEGER PRINT
```

The variable PRINT was replaced by IPRINT everytime it appeared in the function routine.

All of the above changes do not affect the operation of the routine as described in Mathematics Note 1 except in the cases noted. A TIDY[3] computer listing of the current version of BESSEL containing the major subprograms is included in the appendix at the end of this note. The changes which were made are underlined.

GENERAL

Bessel Functions of the first kind, $J_\nu(xe^{i\phi})$, and of the fourth kind, $H_\nu^{(2)}(xe^{i\phi})$, also called Hankel Functions were generated for phase values of 0, $\pi/4$, $\pi/2$, $3\pi/4$, π, $5\pi/4$, $3\pi/2$, $7\pi/4$, for orders 0 and 1, and for arguments (x) .1 to 100 in various steps of incrementation. The derivative of the Bessel Function of the first kind for zero order was also generated for arguments (x) .1 to 100. for accuracy verification. The computed tables, obtained by a direct call to the BESSEL subroutine were compared to tabulated results found directly in reference material or compared to tables computed from equations based on tabulated intermediate results. These tables of comparison are found in Math Memo 1[4]

The Bessel Functions for phase 0 and π could be compared directly to tables for accuracy check and the Bessel Functions for the other phase angles could be related to the ber, bei, and the modified Bessel Function, I(x), formulas and tables for comparison.

The Hankel Functions for phase 0 and π were resolved into the Bessel of the first kind and the Bessel of the second kind, Weber Function, and readily compared to tables. The functions for the other phase values were resolved into ker, kei, ber, bei, I(x), or K(x) values, and then as data points these values were used to calculate the listings for the different phase values as tabulated values. Note that the need to relate to the ber, bei,...functions and for the derivation of formulas is for the production of valid tables of comparison to the computed values.

Reference 2 provided the main source for formulas used in resolving values into ber, bei, and ker, kei relation, for formulas used in analytic continuation,for the various tables used for comparisons in determining accuracy, and for the modified Bessel Function relations needed in developing relations when certain phase angles were employed.

The arguments (x) used for the computed values incremented consistently from .1 to 10. in steps of .1 and from 10 to 100 in steps of 10. The tabulated values used to check the accuracy of the BESSEL subroutine were not as consistent; in various instances a table had to be computed from equations involving modulus and phase values and by using arguments which did not consistently follow the argument list used for the computed values.

The accuracy of these comparisons was very good and a valid check was able to be made for all phase angles. For arguments .1 to 10. of the $J_\nu(xe^{i\phi})$ and $H_\nu^{(2)}(xe^{i\phi})$ when direct tabulated comparisons were possible the accuracy was at least nine places, and for arguments from 10 to 100 the accuracy dropped to about eight places. When the tabulated values for arguments five to ten had to be calculated from the modulus and phase values, the accuracy decreased to about four or five places. For $H_\nu^{(2)}(xe^{\pi i/4})$ and $H_\nu^{(2)}(xe^{3\pi i/4})$ tabulated values the resulting calculations made from the ber, bei, and ker, kei values have a varying accuracy dependent upon the accuracy used in determining the resulting calculations.

The Bessel and Hankel Functions are determined by two different methods and therefore an accuracy check for the BESSEL subroutine can be made and a valid presentation of tables for the Bessel Functions of the first and fourth kind can be given.

EQUATIONS AND DERIVATIONS USED IN OBTAINING TABULATED RESULTS BY RELATING BESSEL AND HANKEL FUNCTIONS TO $J_\nu(x)$, $Y_\nu(x)$, ber, bei, ker, kei, $I_\nu(x)$, or $K_\nu(x)$ FUNCTIONS

Bessel of the first kind

Analytic continuation .... $J_\nu(xe^{m\pi i}) = e^{m\nu\pi i} J_\nu(x) \qquad \nu = 0, 1$

ber, bei relation ... $\text{ber}_\nu x + i\text{bei}_\nu x = J_\nu(xe^{3\pi i/4}) = e^{\nu\pi i} J_\nu(xe^{-\pi i/4})$

$$= e^{\nu\pi i/2} I_\nu(xe^{\pi i/4}) = e^{3\nu\pi i/2} I_\nu(xe^{-3\pi i/4})$$

Modified Bessel Relation ... $I_\nu(x) = e^{3\nu\pi i/2} J_\nu(xe^{-3\pi i/2})$

Modified Analytic Cont ..... $I_\nu(xe^{m\pi i}) = e^{m\nu\pi i} I_\nu(x)$

$0, \pi$

$$J_o(x) = J_o(xe^{\pi i})$$

$$J_1(x) = -J_1(xe^{\pi i})$$

$3\pi/4,\ 7\pi/4$

$$J_o(xe^{-\pi i/4}) = J_o(xe^{3\pi i/4}) = \text{ber}_o x + i\text{bei}_o x$$

$$J_1(xe^{-\pi i/4}) = -J_1(xe^{3\pi i/4}) = -\text{ber}_1 x - i\text{bei}_1 x$$

$\pi/4,\ 5\pi/4$

$$J_o(xe^{\pi i/4}) = J_o(xe^{5\pi i/4}) = \text{ber}_o x - i\text{bei}_o x$$

$$J_1(xe^{\pi i/4}) = -J_1(xe^{5\pi i/4}) = -\text{ber}_1 x + i\text{bei}_1 x$$

$\pi/2,\ 3\pi/2$

$$J_o(xe^{\pi i/2}) = J_o(xe^{3\pi i/2}) = I_o(x)$$

$$J_1(xe^{\pi i/2}) = -J_1(xe^{3\pi i/2}) = iI_1(x)$$

Bessel of the fourth kind

ker, kei, relation ... $\text{ker}_\nu x + i\ \text{kei}_\nu x = e^{-\nu\pi i/2} K_\nu(xe^{\pi i/4})$

$$= \pi i/2\ H_\nu^{(1)}(xe^{3\pi i/4}) = \pi i/2\ e^{-\nu\pi i} H_\nu^{(2)}(xe^{-\pi i/4})$$

Modified Hankel relation ... $K_\nu(x) = -\ \pi i/2\ e^{-\nu\pi i/2} H_\nu^{(2)}(xe^{-\pi i/2})$

Analytic Cont .... $K_\nu(xe^{m\pi i}) = e^{-m\nu\pi i} K_\nu(x) - \pi i \sin(m\nu\pi)\csc(\nu\pi)\ I_\nu(x)$

$\pi/4$

$$H_\nu^{(1)}(xe^{-\pi i/4}) = J_\nu(xe^{-\pi i/4}) + iY_\nu(xe^{-\pi i/4})$$

$$H_\nu^{(2)}(xe^{\pi i/4}) = \overline{J_\nu(xe^{-\pi i/4})} - i\,\overline{Y_\nu(xe^{-\pi i/4})}$$

$$H_\nu^{(2)}(xe^{\pi i/4}) = \overline{H_\nu^{(1)}(xe^{-\pi i/4})}$$

$$= 2\,\overline{J_\nu(xe^{-\pi i/4})} - \overline{J_\nu(xe^{-\pi i/4})} - i\,\overline{Y_\nu(xe^{-\pi i/4})}$$

$$= 2\,\overline{J_\nu(xe^{-\pi i/4})} - [\,\overline{J_\nu(xe^{-\pi i/4})} + i\,\overline{Y_\nu(xe^{-\pi i/4})}]$$

$$H_\nu^{(2)}(xe^{\pi i/4}) = \overline{2\,J_\nu(xe^{-\pi i/4})} - \overline{[J_\nu(xe^{-\pi i/4}) - i\,Y_\nu(xe^{-\pi i/4})]}$$

$$H_\nu^{(2)}(xe^{\pi i/4}) = \overline{2\,J_\nu(xe^{-\pi i/4})} - \overline{H_\nu^{(2)}(xe^{-\pi i/4})}$$

$$H_\nu^{(2)}(xe^{\pi i/4}) = 2\,e^{\nu\pi i}\,[\mathrm{ber}_\nu x - i\,\mathrm{bei}_\nu\,x] + 2i/\pi\,e^{-\nu\pi i}[\mathrm{ker}_\nu x - i\,\mathrm{kei}_\nu\,x]$$

$$= 2(-1)^\nu\,[(\mathrm{ber}_\nu\,x + 1/\pi\,\mathrm{kei}_\nu\,x) + i\,(-\mathrm{bei}_\nu x + 1/\pi\,\mathrm{ker}_\nu\,x)]$$

$$H_o^{(2)}(xe^{\pi i/4}) = 2\,[(\mathrm{ber}_o x + 1/\pi\,\mathrm{kei}_o x) + i(-\mathrm{bei}_o x + 1/\pi\,\mathrm{ker}_o x)]$$

$$H_1^{(2)}(xe^{\pi i/4}) = -2\,[(\mathrm{ber}_1 x + 1/\pi\,\mathrm{kei}_1 x) + i(-\mathrm{bei}_1 x + 1/\pi\,\mathrm{ker}_1 x)]$$

$\pi/2,\ 3\pi/2$

$$H_\nu^{(2)}(xe^{3\pi i/2}) = 2i/\pi\,e^{\pi\nu i/2}\,K_\nu(x)$$

$$H_o^{(2)}(xe^{3\pi i/2}) = 2i/\pi\,K_o(x)$$

$$H_1^{(2)}(xe^{3\pi i/2}) = 2i/\pi\,e^{\pi i/2}\,K_1(x)$$

$$J_\nu(xe^{3\pi i/2}) = (-1)^\nu\,I_\nu(x)$$

$$H_\nu^{(2)}(xe^{3\pi i/2}) = J_\nu(xe^{3\pi i/2}) - i\,Y_\nu(xe^{3\pi i/2})$$

$$H_\nu^{(1)}(xe^{3\pi i/2}) = J_\nu(xe^{3\pi i/2}) + i\,Y_\nu(xe^{3\pi i/2})$$

$$H_\nu^{(2)}(xe^{\pi i/2}) = \overline{H_\nu^{(1)}(xe^{3\pi i/2})}$$

$$= \overline{J_\nu(xe^{3\pi i/2})} - i\,\overline{Y_\nu(xe^{3\pi i/2})}$$

$$= 2\,\overline{J_\nu(xe^{3\pi i/2})} - \overline{J_\nu(xe^{3\pi i/2})} - i\,\overline{Y_\nu(xe^{3\pi i/2})}$$

$$= 2\,\overline{J_\nu(xe^{3\pi i/2})} - [\overline{J_\nu(xe^{3\pi i/2})} + i\,\overline{Y_\nu(xe^{3\pi i/2})}]$$

$$= 2\,\overline{J_\nu(xe^{3\pi i/2})} - \overline{[J_\nu(xe^{3\pi i/2}) - i\,Y_\nu(xe^{3\pi i/2})]}$$

$$= 2\,\overline{J_\nu(xe^{3\pi i/2})} - \overline{H_\nu^{(2)}(xe^{3\pi i/2})}$$

$$= 2(i)^\nu\,I_\nu(x) + 2i/\pi\,e^{-\nu\pi i/2}\,K_\nu(x)$$

$$H_o^{(2)}(xe^{\pi i/2}) = 2\,I_o(x) + 2i/\pi\,K_o(x)$$

$$H_1^{(2)}(xe^{\pi i/2}) = 2i\,I_1(x) + 2/\pi\,K_1(x)$$

$3\pi/4$

$$H_\nu^{(2)}(xe^{3\pi i/4}) = \overline{H_\nu^{(1)}(xe^{-3\pi i/4})}$$

$$= \overline{J_\nu(xe^{-3\pi i/4})} - i\ \overline{Y_\nu(xe^{-3\pi i/4})}$$

$$= 2\ \overline{J_\nu(xe^{-3\pi i/4})} - \overline{J_\nu(xe^{-3\pi i/4})} - i\ \overline{Y_\nu(xe^{-3\pi i/4})}$$

$$= 2\ \overline{J_\nu(xe^{-3\pi i/4})} - [\overline{J_\nu(xe^{-3\pi i/4})} + i\ \overline{Y_\nu(xe^{-3\pi i/4})}]$$

$$= 2\ \overline{J_\nu(xe^{-3\pi l/4})} - [\overline{J_\nu(xe^{-3\pi i/4}) - i\ Y_\nu(xe^{-3\pi l/4})}]$$

$$= 2\ \overline{J_\nu(xe^{-3\pi i/4})} - \overline{H_\nu^{(2)}(xe^{-3\pi i/4})}$$

$$= 2[\mathrm{ber}_\nu x + i\ \mathrm{bei}_\nu x] - 2/\pi\ \mathrm{kei}_\nu x + 2i/\pi\ \mathrm{ker}_\nu x$$

$$H_\nu^{(2)}(xe^{3\pi i/4}) = 2(\mathrm{ber}_\nu x - (\mathrm{kei}_\nu x)/\pi + (\mathrm{bei}_\nu x + (\mathrm{ker}_\nu x)/\pi)i)$$

$$H_o^{(2)}(xe^{3\pi i/4}) = 2(\mathrm{ber}_o x - (\mathrm{kei}_o x)/\pi + (\mathrm{bei}_o x + (\mathrm{ker}_o x)/\pi)i)$$

$$H_1^{(2)}(xe^{3\pi i/4}) = 2(\mathrm{ber}_1 x - (\mathrm{kei}_1 x)/\pi + (\mathrm{ber}_1 x + (\mathrm{ker}_1 x)/\pi)i)$$

π

$$H_\nu^{(2)}(xe^{\pi i}) = J_\nu(xe^{\pi i}) - i\ Y_\nu(xe^{\pi i})$$

$$= e^{\nu\pi i}\ J_\nu(x) - i\ [e^{-\nu\pi i}\ Y_\nu(x) + 2i\cos(\nu\pi)\ J_\nu(x)]$$

$$= J_\nu(x)(e^{\nu\pi i} + 2\ \cos(\nu\pi)) - ie^{-\nu\pi i}\ Y_\nu(x)$$

$$H_o^{(2)}(xe^{\pi i}) = 3\ J_o(x) - i\ Y_o(x)$$

$$H_1^{(2)}(xe^{\pi i}) = -3\ J_1(x) + i\ Y_1(x)$$

$5\pi/4$

$$H_\nu^{(1)}(xe^{3\pi i/4}) = 2/\pi\ \mathrm{kei}_\nu x - 2i/\pi\ \mathrm{ker}_\nu x$$

$$H_\nu^{(2)}(xe^{5\pi i/4}) = 2/\pi\ \mathrm{kei}_\nu x + 2i/\pi\ \mathrm{ker}_\nu x$$

$$H_o^{(2)}(xe^{5\pi i/4}) = 2/\pi\ \mathrm{kei}_o x + 2i/\pi\ \mathrm{ker}_o x$$

$$H_1^{(2)}(xe^{5\pi i/4}) = 2/\pi\ \mathrm{kei}_1 x + 2i/\pi\ \mathrm{ker}_1 x$$

$7\pi/4$

$$H_o^{(2)}(xe^{7\pi i/4}) = 2i/\pi\ [\mathrm{ker}_o x + i\ \mathrm{kei}_o x]$$

$$= -2/\pi\ \mathrm{kei}_o x + 2i/\pi\ \mathrm{ker}_o x$$

$$H_1^{(2)}(xe^{7\pi i/4}) = -2i/\pi\ [\mathrm{ker}_1 x + i\ \mathrm{kei}_1 x]$$

$$= 2/\pi\ \mathrm{kei}_1 x - 2i/\pi\ \mathrm{ker}_1 x$$

SUMMARY

Various changes were made to the BESSEL subroutine mainly due to new specifications on programs running on the 6600 computer at the AFWL and in order for the subroutine to generate functions for arguments with a phase of π. Bessel Functions of the first kind, simply called Bessel Functions, and Bessel Functions of the fourth kind, Hankel Functions, were generated for phase angles of 0, $\pi/4$, $\pi/2$, $3\pi/4$, π, $5\pi/4$, $3\pi/2$, $7\pi/4$, for orders 0 and 1, and for arguments (x) .1 to 100 both by a direct call to the BESSEL subroutine (the computed value), and by using formulas employing intermediate tabulated results to produce tables for comparison. These tables are found in Math Memo 1.

REFERENCES

1. Mathematics Note 1, BESSEL: A Subroutine for the Generation of Bessel Functions with Real or Complex Arguments, Richard C. Lindberg, October 1966.

2. Handbook of Mathematical Functions, M. Abramowitz and I. Stegun, A.M.S. #55, National Bureau of Standards, 1964.

3. TIDY, A Computer Code for Renumbering and Editing FORTRAN Source Programs, Harry M. Murphy, Air Force Weapons Laboratory Technical Report No. AFWL-TR-66-93. October 1966.

4. Mathematics Memo 1: Accuracy Check on Subroutine BESSEL for Bessel Functions of the First and Fourth Kind for Various Orders and Arguments, Richard T. Clark, October 1969.

ACKNOWLEDGEMENTS

The suggestions and assistance of Capt. Carl E. Baum and Joe P. Martinez are gratefully acknowledged.

```
      SUBROUTINE BESSEL (N,Z)
      COMMON /RATIO/ B(2000),FY(1000),FH(1000)
      COMPLEX CSQRT,CLOG,CCOS,CSIN,CEXP,CCON
      COMPLEX Z,JNZ,YNZ,HNZ,JNZPRM,YNZPRM,HNZPRM,JZ,J1,YZ,Y1,HZ,H1,B,FY,
     1FH,CONST,R1,WRONSK,JNZADD,YNZADD,HNZADD,HNONEZ,HNONEA
      COMMON /ARGBESS/ J,Y,H2,JPRIME,YPRIME,H2PRIME,IVALCHK,IPRINT
      COMMON /ARGCBES/ JZ,J1,YZ,HZ,H1
      COMPLEX J,Y,H2,JPRIME,YPRIME,H2PRIME
      REAL MAX
      INTEGER PRINT,CHECK,VALCHK
      EQUIVALENCE (JNZ,J), (YNZ,Y), (HNZ,H2), (JNZPRM,JPRIME), (YPRIME,Y
     1NZPRM), (H2PRIME,HNZPRM), (IVALCHK,VALCHK), (IPRINT,PRINT)
C
C                  -PI.LT.ARG(Z).LT.+PI FOR PROPER OPERATION
C                  WILL ACCEPT NEGATIVE N,ZERO,Z
C
      VALCHK=0
      MAX=SQRT(2.0)*(1.0E150)
      IF (N.LT.0) GO TO 25
      IRETURN=1
      IF (N.GT.1) GO TO 3
      CALL CBESS (Z,Y1,CHECK,PRINT)
      IF (CHECK.NE.0) VALCHK=1
      IF (N.EQ.1) GO TO 1
      JNZ=JZ
      YNZ=YZ
      HNZ=HZ
      JNZPRM=-J1
      YNZPRM=-Y1
      HNZPRM=-H1
      RETURN
    1 JNZ=J1
      YNZ=Y1
      HNZ=H1
      IF (Z.EQ.(0.,0.)) GO TO 2
      JNZPRM=JZ-(1./Z)*J1
      YNZPRM=YZ-(1./Z)*Y1
      HNZPRM=HZ-(1./Z)*H1
      RETURN
    2 JNZPRM=0.
      YNZPRM=0.
      HNZPRM=0.
      RETURN
    3 IF (Z.EQ.(0.,0.)) GO TO 24
      CALL CBESS (Z,Y1,CHECK,PRINT)
      IF (CHECK.NE.0) VALCHK=1
      IDIM=N*20
      IF (IDIM.LT.200) IDIM=200
      IF (IDIM.GT.2000) IDIM=2000
      CALL BKWRD (Z,B,IDIM)
```

```
      JNZ=JZ
      DO 4 I=1,N
      JNZ=JNZ*B(I)
4     CONTINUE
      JNZPRM=(N/Z)*JNZ+(-1.,0.)*JNZ*B(N+1)
      R1=Y1/Y2
      IDIM=N+2
      IF (IDIM.GT.1000) IDIM=1000
      CALL FORWD (Z,FY,IDIM,R1)
      YNZ=YZ
      DO 5 I=1,N
      YNZ=YNZ*(FY(I))
5     CONTINUE
      YNZPRM=(N/Z)*YNZ-YNZ*FY(N+1)
      JNZADD=JNZ*B(N+1)
      YNZADD=YNZ*FY(N+1)
      IF (AIMAG(Z).EQ.0.) GO TO 7
      R1=H1/H2
      IDIM=N+2
      IF (IDIM.GT.1000) IDIM=1000
      CALL FORWD (Z,FH,IDIM,R1)
      HNZ=HZ
      DO 6 I=1,N
      HNZ=HNZ*FH(I)
6     CONTINUE
      HNZPRM=(N/Z)*HNZ-HNZ*FH(N+1)
      GO TO 8
7     HNZ=JNZ+(0.,-1.)*YNZ
      HNZPRM=(N/Z)*HNZ-(JNZ*B(N+1)+(0.,-1.)*(YNZ*FY(N+1)))
      HNZADD=(JNZ*B(N+1))+(0.,-1.)*(YNZ*FY(N+1))
      GO TO 9
8     HNZADD=HNZ*FH(N+1)
9     DIFF=ABS(ZERO(Z,N,JNZ,JNZADD,JNZPRM,PRINT))
      IF (DIFF.GT.1.0E-8) GO TO 16
10    DIFF=ABS(ZERO(Z,N,YNZ,YNZADD,YNZPRM,PRINT))
      IF (DIFF.GT.1.0E-8) GO TO 20
11    DIFF=ABS(ZERO(Z,N,HNZ,HNZADD,HNZPRM,PRINT))
      IF (DIFF.GT.1.0E-8) GO TO 12
      GO TO (27,26), IRETURN
12    IF (PRINT.EQ.0) GO TO (27,26), IRETURN
      IF (DIFF.GT.1.0E-6) GO TO 13
      PRINT 28, N,Z
      GO TO (27,26), IRETURN
13    IF (DIFF.GT.1.0E-4) GO TO 14
      PRINT 29, N,Z
      GO TO (27,26), IRETURN
14    IF (DIFF.GT.1.0E-2) GO TO 15
      PRINT 30, N,Z
      GO TO (27,26), IRETURN
15    PRINT 31, N,Z
```

```
      GO TO (27,26), IRETURN
16    IF (IPRINT.EQ.0) GO TO 10
      IF (DIFF.GT.1.0E-6) GO TO 17
      PRINT 32, N,Z
      GO TO 10
17    IF (DIFF.GT.1.0E-4) GO TO 18
      PRINT 33, N,Z
      GO TO 10
18    IF (DIFF.GT.1.0E-2) GO TO 19
      PRINT 34, N,Z
      GO TO 10
19    PRINT 35, N,Z
      GO TO 10
20    IF (IPRINT.EQ.0) GO TO 11
      IF (DIFF.GT.1.0E-6) GO TO 21
      PRINT 36, N,Z
      GO TO 11
21    IF (DIFF.GT.1.0E-4) GO TO 22
      PRINT 37, N,Z
      GO TO 11
22    IF (DIFF.GT.1.0E-2) GO TO 23
      PRINT 38, N,Z
      GO TO 11
23    PRINT 39, N,Z
      GO TO 11
24    JNZ=(0.,0.)
      YNZ=(-1.E300,0.)
      HNZ=(1.E300,0.)
      JNZPRM=(0.,0.)
      YNZPRM=(1.E300,0.)
      HNZPRM=(-1.E300,0.)
      GO TO (27,26), IRETURN
25    IRETURN=2
      N=-N
      GO TO 3
26    IF ((CABS(JNZ).GT.MAX).OR.(CABS(YNZ).GT.MAX).OR.(CABS(HNZ).GT.MAX)
     1.OR.(CABS(JNZPRM).GT.MAX).OR.(CABS(YNZPRM).GT.MAX).OR.(CABS(HNZPRM
     2).GT.MAX)) VALCHK=1
      IF (((N-(N/2))-(N/2)).EQ.0) RETURN
      JNZ=-JNZ
      YNZ=-YNZ
      HNZ=-HNZ
      JNZPRM=(N/Z)*JNZ+( JNZ*B(N+1))
      YNZPRM=(N/Z)*YNZ+(YNZ*FY(N+1))
      HNZPRM=JNZPRM+(0.,-1.)*YNZPRM
      RETURN
27    IF ((CABS(JNZ).GT.MAX).OR.(CABS(YNZ).GT.MAX).OR.(CABS(HNZ).GT.MAX)
     1.OR.(CABS(JNZPRM).GT.MAX).OR.(CABS(YNZPRM).GT.MAX).OR.(CABS(HNZPRM
     2).GT.MAX)) VALCHK=1
      RETURN
```

```
 28   FORMAT (58H DIFF. EQN CHECK FOR HANKELS SHOWS AGREEMENT TO 10**-6.
     1   N=I3,4H, Z=2E16.7)
 29   FORMAT (58H DIFF. EQN CHECK FOR HANKELS SHOWS AGREEMENT TO 10**-4.
     1   N=I3,4H, Z=2E16.7)
 30   FORMAT (58H DIFF. EQN CHECK FOR HANKELS SHOWS AGREEMENT TO 10**-2.
     1   N=I3,4H, Z=2E16.7)
 31   FORMAT (66H0DIFF. EQN CHECK FOR HANKELS DOES NOT SHOW AGREEMENT TO
     1   10**-2. N=I3,4H, Z=2E16.7/)
 32   FORMAT (58H0DIFF. EQN CHECK FOR HANKELS SHOWS AGREEMENT TO 10**--8
     1.N=I3,4H, Z=2E16.7)
 33   FORMAT (58H0DIFF. EQN CHECK FOR HANKELS SHOWS AGREEMENT TO 10**-4.
     1   N=I3,4H, Z=IE16.7)
 34   FORMAT (58H0DIFF. EQN CHECK FOR HANKELS SHOWS AGREEMENT TO 10**-2.
     1   N=I3,4H, Z=2E16.7)
 35   FORMAT (66H0DIFF. EQN CHECK FOR HANKELS DOES NOT SHOW AGREEMENT TO
     1   10**-2. N=I3,4H, Z=2E16.7/)
 36   FORMAT (69H0DIFF. EQN CHECK FOR NEUMANN FUNCTIONS SHOWS AGREEMENT
     1 TO 10**-6. N=I3,4H, Z=2E16.7)
 37   FORMAT (69H0DIFF. EQN. CHECK FOR NEUMANN FUNCTIONS SHOWS AGREEMENT
     1 TO 10**-4. N=I3,4H, Z=2E16.7)
 38   FORMAT (69H0DIFF. EQN. CHECK FOR NEUMANN FUNCTIONS SHOWS AGREEMENT
     1 TO 10**-2. N=I3,4H, Z=2E16.7)
 39   FORMAT (77H0DIFF. EQN. CHECK FOR NEUMANN FUNCTIONS DOES NOT SHOW A
     1GREEMENT TO 10**-2. N=I3,4H, Z=2E16.7/)
      END
```

```
      FUNCTION ZERO (Z,N,A,AADD,APRIME,IPRINT)
C
      COMPLEX Z,A,AADD,APRIME,FACT1,FACT2,RATIO
      FACT1=Z*(AADD+APRIME)
      FACT2=N*A
      RATIO=FACT1/FACT2
      ZERO=1.000000000001-CABS(RATIO)
      IF (ABS(ZERO).LT.1.0E-08) RETURN
      RATIO=REAL(FACT1)/REAL(FACT2)+(0.,1.)*(AIMAG(FACT1)/AIMAG(FACT2))
      ZERO=1.000000000001-CABS(RATIO)
      IF (ABS(ZERO).LT.1.0E-08) RETURN
      IF (IPRINT.EQ.0) GO TO 1
      PRINT 3, FACT1,FACT2
1     FACT1R=REAL(FACT1)
      FACT1I=AIMAG(FACT1)
      FACT2R=REAL(FACT2)
      FACT2I=AIMAG(FACT2)
      L1RE=FACT1R.AND.03777000000000000
      L1IE=FACT1I.AND.03777000000000000
      L2RE=FACT2R.AND.03777000000000000
      L2IE=FACT2I.AND.03777000000000000
      L1RM=FACT1R.AND.07777777777777777
      L1IM=FACT1I.AND.07777777777777777
      L2RM=FACT2R.AND.07777777777777777
      L2IM=FACT2I.AND.07777777777777777
      IF ((L1RE.NE.L2RE).OR.(L1IE.NE.L2IE))  GO TO 2
      ZERO=CABS((L1RM-L2RM)+(0.0,1.0)*(L1IM-L2IM))*1.0E-9
      IF (ABS(ZERO).LT.1.0E-08) RETURN
2     IF (IPRINT.EQ.0) RETURN
      PRINT 4, FACT1R,L1RE,L1RM,FACT2R,L2RE,L2RM,FACT1I,L1IE,L1IM,FACT2I
     1,L2IE,L2IM
      RETURN
C
3     FORMAT (4E20.10)
4     FORMAT (8H FACT1R=3(020,3X)/8H FACT2R=3(020,3X)//8H FACT1I=3(020,3
     1X)/8H FACT2I=3(020,3X))
      END
```

```
      SUBROUTINE BKWRD (Z,RATIO,IDIM)
      DIMENSION RATIO(IDIM)
      COMPLEX RATIO,KONST,DENOM,R1,Z
      KONST(I)=2.*I/Z
      DO 1 J=1,IDIM
      RATIO(J)=( .,0.)
1     CONTINUE
      I=IDIM-1
      RATIO(IDIM)=Z/(2.*IDIM)
2     DENOM=KONST(I)-RATIO(I+1)
      IF (DENOM) 3,4,3
3     RATIO(I)=1./DENOM
      GO TO 5
4     RATIO(I)=1.0E300
5     I=I-1
      IF (I) 6,6,2
6     RETURN
      END
```

```
      SUBROUTINE FRWRD (Z,RATIO,IDIM,R1)
      DIMENSION RATIO(IDIM)
      COMPLEX RATIO,R1,Z,KONST
      KONST(I)=2.*I/Z
      IMAX=IDIM-1
      RATIO(1)=R1
      DO 1 I=1,IMAX
      RATIO(I+1)=(RATIO(I)*KONST(I)-(1.,0.))/RATIO(I)
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CBESS (Z,YONE,IVALCHK,IPRINT)
      COMPLEX  JZ,J1,YZ,Y1,H2Z,H21
      COMPLEX Z,JZERO,JONE,YZERO,YONE,ZSQ,FACT,ZFACT,JZADD,J1ADD,EZ,P,Q,
     1COSP,SINP,H2ZERO,H2ONE,CONST,WRONSK
      COMMON /ARGCBES/ JZ,J1,YZ,H2Z,H21
      COMPLEX CSQRT,CLOG,CCOS,CSIN,CEXP
      REAL MAX
      EQUIVALENCE (JZERO,JZ), (JONE,J1), (YZERO,YZ), (H2ZERO,H2Z), (H2ON
     1E,H21)
      MAX=SQRT(2.0)*(1.0E150)
      IVALCHK=0
      IF (CABS(Z)-10.) 1,1,4
1     JZERO=(1.,0.)
      JONE=(1.,0.)
      YZERO=(0.,0.)
      YONE=(1.,0.)
      FK=1.
      FKFACT=1.
      ZSQ=-Z*Z*.25
      ZFACT=(1.,0.)
      SKINV=1.
2     ZFACT=ZFACT*ZSQ
      FACT=ZFACT/FKFACT
      JZADD=FACT/FKFACT
      FK=FK+1.
      FKFACT=FKFACT*FK
      J1ADD=FACT/FKFACT
      JZERO=JZERO+JZADD
      JONE=JONE+J1ADD
      YZERO=YZERO+JZADD*SKINV
      YONE=YONE+J1ADD*(SKINV+SKINV+1./FK)
      SKINV=SKINV+1./FK
      IF (JZERO.EQ.(0.,0.)) JZERO=CMPLX(1.E-15,0.)
      IF (CABS(JZADD/JZERO).GT.1.0E-25) GO TO 2
      IF (CABS(J1ADD/JONE).GT.1.0E-25) GO TO 2
      JONE=JONE*Z*.5
      IF (Z.EQ.(0.,0.)) GO TO 3
      YZERO=((.5772156649+CLOG(Z*.5))*JZERO-YZERO)/1.570796326795
      YONE=((.5772156649+CLOG(Z*.5))*JONE-(1.-ZSQ*YONE)/Z)/1.57079632579
     15
      H2ZERO=JZERO+(0.,-1.)*YZERO
      H2ONE=JONE+(0.,-1.)*YONE
      GO TO 12
3     YZERO=(-1.E300,0.)
      YONE=(-1.E300,0.)
      H2ZERO=(1.E300,0.)
      H2ONE=(1.E300,0.)
      IVALCHK=1
      RETURN
4     K=0
```

```
      IF (ABS(AIMAG(Z)).LT.1.E-9.AND.REAL(Z).LT.0.) GO TO 5
      GO TO 6
5     Z=-Z
      K=1
6     FACT=3.141592653589793*Z
      FACT=CSQRT(FACT)
      COSP=CCOS(Z)/FACT
      SINP=CSIN(Z)/FACT
      ZFACT=(1.,1.)*CEXP((0.,-1.)*Z)/FACT
      U=0.
      EZ=8.0*Z
7     FN=1.
      FK=1.
      P=1.
      Q=(U-1.)/EZ
      FACT=Q
8     FN=FN+2.
      FK=FK+1.
      FACT=-FACT*(U-FN*FN)/EZ/FK
      P=P+FACT
      FN=FN+2
      FK=FK+1
      FACT=FACT*(U-FN*FN)/EZ/FK
      Q=Q+FACT
      IF (CABS(FACT/Q).LT.1.0E-8) GO TO 9
      IF (FK.LT.21.0) GO TO 8
9     IF (U) 10,10,11
10    JZERO=(P+Q)*COSP+(P-Q)*SINP
      YZERO=(P+Q)*SINP-(P-Q)*COSP
      H2ZERO=ZFACT*(P+(0.,-1.)*Q)
      U=4.
      GO TO 7
11    JONE=(P+Q)*SINP-(P-Q)*COSP
      YONE=-(P+Q)*COSP-(P-Q)*SINP
      H2ONE=ZFACT*(Q+(0.,1.)*P)
      IF (K.EQ.0) GO TO 12
      Z=-Z
      YZERO=YZERO+2.*JZERO*(0.,1.)
      H2ZERO=JZERO-(0.,1.)*YZERO
      JONE=-JONE
      YONE=-YONE+2.*JONE*(0.,1.)
      H2ONE=JONE-(0.,1.)*YONE
12    CONST=(2.,0.)/(3.141592653589793*Z)
      IF ((CABS(JZERO).GT.MAX).OR.(CABS(JONE).GT.MAX).OR.(CABS(YZERO).GT
     1.MAX).OR.(CABS(YONE).GT.MAX)) IVALCHK=1
      IF (IPRINT.EQ.0) RETURN
      P=JONE*YZERO
      Q=JZERO*YONE+CONST
      WRONSK=P/Q
      DIFF=1.000000000001-CABS(WRONSK)
```

```
      IF (ABS(DIFF).LT.1. E-8) RETURN
      IF (ABS(DIFF).GT.1.0E-6) GO TO 13
      PRINT 16, Z
      RETURN
 13   IF (ABS(DIFF).GT.1.0E-4) GO TO 14
      PRINT 17, Z
      RETURN
 14   IF (ABS(DIFF).GT.1.0E-2) GO TO 15
      PRINT 18, Z
      RETURN
 15   PRINT 19, Z
      RETURN
C
 16   FORMAT (66H0WRONSKIAN CHECK FOR BESSELS, N=0,1, SHOWS AGREEMENT TO
     1 10**-6. Z=2E16.7)
 17   FORMAT (66H0WRONSKIAN CHECK FOR BESSELS, N=0,1, SHOWS AGREEMENT TO
     1 10**-4. Z=2E16.7)
 18   FORMAT (66H0WRONSKIAN CHECK FOR BESSELS, N=0,1,SHOWS AGREEMENT TO
     1 10**-2. Z=2E16.7)
 19   FORMAT (74H0WRONSKIAN CHECK FOR BESSELS, N=0,1, DOES NOT SHOW AGRE
     1EMENT TO 10**-2. 7=2E16.7/)
      END
```

```
      FUNCTION ARG (Z)
      COMPLEX Z
      X=REAL(Z)
      Y=AIMAG(Z)
      IF (Y*Y.LT.1.E-12) Y=0.
      IF (Y) 4,1,4
1     IF (X) 2,3,3
2     ARG=3.141592653589793
      RETURN
3     ARG=0.0
      RETURN
4     ARG=2.0*ATAN(Y/(SQRT(X*X+Y*Y)+X))
      RETURN
      END
```