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

AIC 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 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, π/4, π/2, 3π/4, π, 5π/4, 3π/2, and 7π/4, for orders 0, and 1, and for arguments (x) for 0.1 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 $|Z| > 10.0$ 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+0i 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 \( \pi \) so a few changes were made in order for the subroutine to accept it.

\[
H^{(2)}_\nu(xe^{\pi i}) = J_\nu(xe^{\pi i}) - iY_\nu(xe^{\pi i}) \quad \nu = 0, 1
\]

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 \( \pi \) 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 \( \pi \):

4  K=0
   IF(ABS(ALMAG(Z)).LT.1.E-9.AND.REAL(Z).LT.0.) GO TO 5
   GO TO 6
5   Z=-Z
   K=1
   
   IF(K.EQ.0) GO TO 12
   
   Z=-Z
   YZERO=YZERO+2.*JZERO*(0.,1.)
   HZZERO=JZERO-(0.,1.)*YZERO
   JONE=-JONE
   YONE=-YONE+2.*JONE*(0.,1.)
   HZONE=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

\[
\text{IF}(Y*Y,LT,1.E-12)Y=0.
\]

It was also noticed that the value for \( \pi \) 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

\[
\text{EQUIVALENCE(IPRINT,PRINT)}
\]
\[
\text{INTEGER PRINT}
\]

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 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, \( H^{(2)}_\nu(xe^{i\phi}) \), also called Hankel Functions were generated for phase values of 0, \( \pi/4 \), \( \pi/2 \), \( 3\pi/4 \), \( \pi \), \( 5\pi/4 \), \( 3\pi/2 \), \( 7\pi/4 \), for orders 0 and 1, and for arguments \( (x) \cdot 1 \) 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 \( (x) \cdot 1 \) 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 14.

The Bessel Functions for phase 0 and \( \pi \) 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 \( \pi \) 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 \( H^{(2)}_\nu(xe^{i\phi}) \) 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 \( H^{(2)}_\nu(xe^{\pi i/4}) \) and \( H^{(2)}_\nu(xe^{3\pi i/4}) \) 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 .... $J_\nu(x e^{\nu \pi i}) = e^{\nu \pi i} J_\nu(x) \quad \nu = 0, 1$
ber, bei relation ... $ber_\nu x + bei_\nu x = J_\nu(xe^{3\pi i/4}) = e^{\nu \pi i} J_\nu(xe^{-\pi i/4})$
$= e^{\nu \pi i/2} I_\nu(xe^{\pi i/4}) = e^{3\nu \pi i/2} I_\nu(xe^{-3\pi i/4})$

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

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

$0, \pi$

$J_0(x) = J_0(x e^{\pi i})$

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

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

$J_0(x e^{-\pi i/4}) = J_0(x e^{3\pi i/4}) = ber_0 x + bei_0 x$

$J_1(x e^{-\pi i/4}) = -J_1(x e^{3\pi i/4}) = -ber_1 x - bei_1 x$

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

$J_0(x e^{\pi i/4}) = J_0(x e^{5\pi i/4}) = ber_0 x - bei_0 x$

$J_1(x e^{\pi i/4}) = -J_1(x e^{5\pi i/4}) = -ber_1 x + bei_1 x$

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

$J_0(x e^{\pi i/2}) = J_0(x e^{3\pi i/2}) = I_0(x)$

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

Bessel of the fourth kind

ker, kei, relation ... $ker_\nu x + i kei_\nu x = e^{-\nu \pi i/2} K_\nu(xe^{\pi i/4})$
$= \pi i/2 H_{1/2}^{(1)}(x e^{3\pi i/4}) = \pi i/2 e^{-\nu \pi i} H_{1/2}^{(2)}(x e^{-\pi i/4})$

Modified Hankel relation ... $K_\nu(x) = -\pi i/2 e^{-\nu \pi i/2} H_{1/2}^{(2)}(xe^{-\pi i/2})$

Analytic Cont .... $K_\nu(x e^{\nu \pi i}) = e^{-\nu \pi i} K_\nu(x) - \pi i \sin(\nu \pi) \csc(\nu \pi) I_\nu(x)$
\[
\begin{align*}
H_0^{(1)}(xe^{-\pi i/4}) &= J_0(xe^{-\pi i/4}) + iY_0(xe^{-\pi i/4}) \\
H_0^{(2)}(xe^{\pi i/4}) &= J_0(xe^{-\pi i/4}) - iY_0(xe^{-\pi i/4}) \\
H_1^{(2)}(xe^{\pi i/4}) &= H_0^{(1)}(xe^{\pi i/4}) \\
&= 2J_0(xe^{-\pi i/4}) - [J_0(xe^{-\pi i/4}) + iY_0(xe^{-\pi i/4})] \\
H_0^{(2)}(xe^{\pi i/4}) &= 2J_0(xe^{-\pi i/4}) - 2iY_0(xe^{-\pi i/4}) \\
H_1^{(2)}(xe^{\pi i/4}) &= 2H_0^{(2)}(xe^{-\pi i/4}) \\
&= 2(-1)^{\nu} [(ber_{\nu} x + i\operatorname{bei}_{\nu} x) + i(-\operatorname{bei}_{\nu} x + 1/\pi \operatorname{ker}_{\nu} x)] \\
&= 2(1/\pi) e^{-\nu i}[\operatorname{ker}_{\nu} x - i \operatorname{kei}_{\nu} x] \\
&= 2(-1)^{\nu} [(ber_{\nu} x + 1/\pi \operatorname{kei}_{\nu} x) + i(-\operatorname{bei}_{\nu} x + 1/\pi \operatorname{ker}_{\nu} x)] \\
&= 2 [\operatorname{ber}_{\nu} x + 1/\pi \operatorname{kei}_{\nu} x] + i \operatorname{bei}_{\nu} x + 1/\pi \operatorname{ker}_{\nu} x] \\
&\quad + i \operatorname{ber}_{\nu} x + 1/\pi \operatorname{kei}_{\nu} x) + i(-\operatorname{ber}_{\nu} x + 1/\pi \operatorname{kei}_{\nu} x)] \\
&= 2 [\operatorname{ber}_{\nu} x + 1/\pi \operatorname{kei}_{\nu} x] + i(-\operatorname{ber}_{\nu} x + 1/\pi \operatorname{kei}_{\nu} x)] \\
&= 2 [\operatorname{ber}_{\nu} x + 1/\pi \operatorname{kei}_{\nu} x] + i(-\operatorname{ber}_{\nu} x + 1/\pi \operatorname{kei}_{\nu} x)] \\
&= 2(1/\pi) e^{-\nu i}[\operatorname{ker}_{\nu} x - i \operatorname{kei}_{\nu} x] \\
&= 2(-1)^{\nu} [(ber_{\nu} x + 1/\pi \operatorname{kei}_{\nu} x) + i(-\operatorname{bei}_{\nu} x + 1/\pi \operatorname{ker}_{\nu} x)] \\
&= 2 [\operatorname{ber}_{\nu} x + 1/\pi \operatorname{kei}_{\nu} x] + i(-\operatorname{ber}_{\nu} x + 1/\pi \operatorname{kei}_{\nu} x)] \\
&= 2 [\operatorname{ber}_{\nu} x + 1/\pi \operatorname{kei}_{\nu} x] + i(-\operatorname{ber}_{\nu} x + 1/\pi \operatorname{kei}_{\nu} x)] \\
&= 2 [\operatorname{ber}_{\nu} x + 1/\pi \operatorname{kei}_{\nu} x] + i(-\operatorname{ber}_{\nu} x + 1/\pi \operatorname{kei}_{\nu} x)] \\
&= 2(1/\pi) e^{-\nu i}[\operatorname{ker}_{\nu} x - i \operatorname{kei}_{\nu} x] \\
&= 2(-1)^{\nu} [(ber_{\nu} x + 1/\pi \operatorname{kei}_{\nu} x) + i(-\operatorname{bei}_{\nu} x + 1/\pi \operatorname{ker}_{\nu} x)] \\
&= 2 [\operatorname{ber}_{\nu} x + 1/\pi \operatorname{kei}_{\nu} x] + i(-\operatorname{ber}_{\nu} x + 1/\pi \operatorname{kei}_{\nu} x)] \\
&= 2 [\operatorname{ber}_{\nu} x + 1/\pi \operatorname{kei}_{\nu} x] + i(-\operatorname{ber}_{\nu} x + 1/\pi \operatorname{kei}_{\nu} x)] \\
&= 2(1/\pi) e^{-\nu i}[\operatorname{ker}_{\nu} x - i \operatorname{kei}_{\nu} x] \\
&= 2(-1)^{\nu} [(ber_{\nu} x + 1/\pi \operatorname{kei}_{\nu} x) + i(-\operatorname{bei}_{\nu} x + 1/\pi \operatorname{ker}_{\nu} x)] \\
&= 2 [\operatorname{ber}_{\nu} x + 1/\pi \operatorname{kei}_{\nu} x] + i(-\operatorname{ber}_{\nu} x + 1/\pi \operatorname{kei}_{\nu} x)]}
\begin{align*}
H_{\pi/4}^{(2)}(xe^{3\pi i/4}) &= H_{\pi/4}^{(1)}(xe^{-3\pi i/4}) \\
&= J_{\pi/4}(xe^{-3\pi i/4}) - i Y_{\pi/4}(xe^{-3\pi i/4}) \\
&= 2 J_{\pi/4}(xe^{-3\pi i/4}) - J_{\pi/4}(xe^{-3\pi i/4}) - i Y_{\pi/4}(xe^{-3\pi i/4}) \\
&= 2 J_{\pi/4}(xe^{-3\pi i/4}) - [J_{\pi/4}(xe^{-3\pi i/4}) + i Y_{\pi/4}(xe^{-3\pi i/4})] \\
&= 2 J_{\pi/4}(xe^{-3\pi i/4}) - [J_{\pi/4}(xe^{-3\pi i/4}) - i Y_{\pi/4}(xe^{-3\pi i/4})] \\
&= 2 J_{\pi/4}(xe^{-3\pi i/4}) - H_{\pi/4}^{(2)}(xe^{-3\pi i/4}) \\
&= 2[ber_{\pi/4}x + i bei_{\pi/4}x] - 2/\pi kei_{\pi/4}x + 2i/\pi ker_{\pi/4}x
\end{align*}

\begin{align*}
H_{\pi/4}^{(2)}(xe^{3\pi i/4}) &= 2(ber_{\pi/4}x - (kei_{\pi/4}x)/\pi + (bei_{\pi/4}x + (ker_{\pi/4}x)/\pi)i) \\
H_{\pi/4}^{(2)}(xe^{3\pi i/4}) &= 2(ber_{\pi/4}x - (kei_{\pi/4}x)/\pi + (bei_{\pi/4}x + (ker_{\pi/4}x)/\pi)i) \\
H_{\pi/4}^{(2)}(xe^{3\pi i/4}) &= 2(ber_{\pi/4}x - (kei_{\pi/4}x)/\pi + (bei_{\pi/4}x + (ker_{\pi/4}x)/\pi)i)
\end{align*}

\begin{align*}
H_{\pi}^{(2)}(xe^{\pi i}) &= J_{\pi}(xe^{\pi i}) - i Y_{\pi}(xe^{\pi i}) \\
&= e^{\pi i} J_{\pi}(x) - i [e^{-\pi i} Y_{\pi}(x) + 2\cos(\pi) J_{\pi}(x)] \\
&= J_{\pi}(x)(e^{\pi i} + 2 \cos(\pi)) - ie^{-\pi i} Y_{\pi}(x)
\end{align*}

\begin{align*}
H_{\pi}^{(2)}(xe^{\pi i}) &= 3 J_{0}(x) - i Y_{0}(x) \\
H_{\pi}^{(2)}(xe^{\pi i}) &= -3 J_{1}(x) + i Y_{1}(x)
\end{align*}

\begin{align*}
H_{\pi/4}^{(1)}(xe^{3\pi i/4}) &= 2/\pi kei_{\pi/4}x - 2i/\pi ker_{\pi/4}x \\
H_{\pi/4}^{(2)}(xe^{5\pi i/4}) &= 2/\pi kei_{\pi/4}x + 2i/\pi ker_{\pi/4}x \\
H_{\pi/4}^{(2)}(xe^{5\pi i/4}) &= 2/\pi kei_{\pi/4}x + 2i/\pi ker_{\pi/4}x \\
H_{\pi}^{(2)}(xe^{5\pi i/4}) &= 2/\pi kei_{\pi}x + 2i/\pi ker_{\pi}x
\end{align*}

\begin{align*}
H_{\pi/4}^{(2)}(xe^{7\pi i/4}) &= 2i/\pi [ker_{\pi}x + i kei_{\pi}x] \\
&= -2/\pi kei_{\pi}x + 2i/\pi ker_{\pi}x \\
H_{\pi/4}^{(2)}(xe^{7\pi i/4}) &= -2i/\pi [ker_{\pi}x + i kei_{\pi}x] \\
&= 2/\pi kei_{\pi}x - 2i/\pi ker_{\pi}x
\end{align*}
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 $\pi$. 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, $\pi/4$, $\pi/2$, $3\pi/4$, $\pi$, $5\pi/4$, $3\pi/2$, $7\pi/4$, 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.


ACKNOWLEDGEMENTS

The suggestions and assistance of Capt. Carl E. Baum and Joe P. Martinez are gratefully acknowledged.
SUBROUTINE HESSEL (N,Z)
COMMON /HAT10/ B(2000),FY(1000),FH(1000)
COMPLEX CSCTR,CLOG,CCOS,CSIN,CEXP,CCON
COMPLEX Z,JNZ,YNZ,HNZ,JNZPRM,YNZPRM,HNZPRM,J1,Y1,H1,B,FY,
1FH,CUNST,M1,WRNSK,JNZAUD,YNZAUD,HNZAUD,HNONEZ,HNONEA
COMMON /AMGBESS/ J,Y,H2,JPRIME,YPRIME,J2PRIME,JIVALCHK,JPRINT
COMMON /AMGBRES/ J7,J1,YZ,H7,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),
1JNZPRM, (H2PRIME*H2PRIME), (IVALCHK*,VALCHK), (JPRINT*,PRINT)
-PI*LT(ARG(Z))/LT*PI FOR PROPER OPERATION
WILL ACCEPT NEGATIVE N,Z EQ 0,Z

Valchk=0
MAX=SUM(1.0*E150)
IF (N .LT. 0) GO TO 25
IRETURN=1
IF (N .GT. 1) GO TO 3
CALL CRESS (Z,Y,CHECK,PRINT)
IF (CHECK. NE. 0) VALCHK=1
IF (N .EQ. 1) GO TO 1
JNZ=J
YNZ=Y
HNZ=H
JNZPRM=-J1
YNZPRM=-Y1
HNZPRM=-H1
RETURN
JNZ=J
YNZ=Y
HNZ=H
IF (Z .GE. 1.0*E0) GO TO 2
JNZPRM=-J1
YNZPRM=Y1
HNZPRM=H1
RETURN
JNZPRM=*
YNZPRM=*
HNZPRM=*
RETURN
IF (Z .LE. 0.0*E0) GO TO 24
CALL CRESS (Z,Y,1,CHECK,PRINT)
IF (CHECK=. NE. 0) VALCHK=1
IDIM=NO2
IF (IDIM.LT.200) IDIM=200
IF (IDIM .LT. 2000) IDIM=2000
CALL BWHDU (Z,IM,INTM)
JNZ=JZ
DO 4 I=1,N
JNZ=JZ+M(N)
4 CONTINUE
JNZ=JZ+M(N)
M1=Y/Z
I1=M+1
IF (I1.GT.GL1000) IM=1000
CALL FMHU(Z;FY;IM;R1)
Y=Z+Y
DO 5 J=1,N
YNZ=Y+Y(J)
5 CONTINUE
YNZ=YNZ+Y(N)
JNZ=JZ+M(N)
YNZ=YNZ+Y(N+1)
IF (A*IMAG(Z)*JU*0.8) GO TO 7
M1=M1+1
I1=I1+1
IF (I1.GT.GL1000) IM=1000
CALL FMHU(Z;FY;IM;R1)
YNZ=YNZ+Y(N)
DO 6 J=1,N
YNZ=YNZ+Y(N)
6 CONTINUE
YNZ=YNZ+Y(N)
GO TO 7
7 MNZ=MNZ+(I1+1)*YMNZ
MNZ=MNZ-(MNZK+M(N)+M(N+1))*YMNZ*FY(N+1))
MNZ=MNZ-(JNZ*Y(J+1)+M(N+1))*YMNZ*FY(N+1))
GO TO 10
8 MNL=MNL+Z*FH(N+1)
9 DIFF=ABS(Z+H0(Z+H;H;J;ZM;PK;NT))
IF (DIFF.GT.1.E-8) GO TO 16
10 DIFF=ABS(Z+H0(Z+H;H;ZMN;ZP;PK;NT))
IF (DIFF.GT.1.E-8) GO TO 20
11 DIFF=ABS(Z+H0(Z+H;H;ZMN;ZP;PK;NT))
IF (DIFF.GT.1.E-8) GO TO 12
GO TO (27*Z6), IRETURN
12 IF (PK=INT(I.E-6)) GO TO (27*Z6), IRETURN
IF (DIFF.GT.1.E-6) GO TO 13
PRINT 26,N
GO TO (27*Z6), IRETURN
13 IF (DIFF.GT.1.E-4) GO TO 14
PRINT 29,N
GO TO (27*Z6), IRETURN
14 IF (DIFF.GT.1.E-2) GO TO 15
PRINT 30,N
GO TO (27*Z6), IRETURN
15 PRINT 31,N
100
GO TO (L1*)+6). IHEH-^RIN 101
15  IF (PRT18.EQ.0) GO TO 10 102
15  IF (TIIFF.*Gt.1.*E=6) GO TO 17 103
PRT18 32*+Z 104
GO TO 16 105
17  IF (DIFF.*GT.1.*E=4) GO TO 14 106
PRINT 33*+Z 107
GO TO 18 108
14  IF (DIFF.*GT.1.*E=2) GO TO 19 109
PRINT 34*+Z 110
GO TO 16 111
19  PRINT 35*+Z 112
GO TO 18 113
20  IF (PRT18.EQ.0) GO TO 19 114
20  IF (DIFF.*GT.1.*E=6) GO TO 21 115
PRII 36*+Z 116
GO TO 11 117
21  IF (DIFF.*GT.1.*E=4) GO TO 22 118
PRINT 37*+Z 119
GO TO 17 120
22  IF (DIFF.*GT.1.*E=2) GO TO 23 121
PRRT 38*+Z 122
GO TO 11 123
23  PRINT 39*+Z 124
GO TO 11 125
24  JNZ=(O.0)* 126
YNDZ=-1300+0,0 127
HNZ=1300+0 128
JNZP2=O.0* 129
YNZP2=(1.300+0) 130
HNZP2=(-1.300+0) 131
GO TO (2+26). IHPRETURN 132
1  IRETURN=2 133
N=O 134
GO TO 2 135
24  IF (((CAHS(1NZ)*GT,*MAX) OR (CAHS(1NZ)*GT,*MAX)) OR (CAHS(1NS)*GT,MAX) 136
1 OR (CAHS(1NZP2)*GT,*MAX) OR (CAHS(1NZP2)*GT,*MAX)) OR (CAHS(1NSP2) 137
2)*GT,*MAX) IVALCHK=1 138
IF (((1-(N/2))-(N/2)).EQ.0) RETURN 139
JNZ=JNZ 140
YNDZ=YNDZ 141
HNZ=HNZ 142
JNZP2=(N/2)*JNZ* (1NZP2(N/2)) 143
YNZP2=(N/2)*YNZ* (1NZP2(N/2)) 144
HNZP2=JNZP2*(0.0+1)*YNZP2 145
K=K146
27  IF (((CAHS(1NZ)*GT,*MAX) OR (CAHS(1NZ)*GT,*MAX)) OR (CAHS(1NS)*GT,*MAX) 147
1 OR (CAHS(1NZP2)*GT,*MAX) OR (CAHS(1NZP2)*GT,*MAX)) OR (CAHS(1NSP2) 148
2)*GT,*MAX)) IVALCHK=1 149
RETURN 150
FORMAT (5DF, 4FH, 4FH, 3D, 3D) 151
1 N=13,4 H, Z=2E16,7) 152
2 FORMAT (5DF, 4FH, 4FH, 3D, 3D) 153
1 N=13,4 H, Z=2E16,7) 154
3 FORMAT (5DF, 4FH, 4FH, 3D, 3D) 155
1 N=13,4 H, Z=2E16,7) 156
4 FORMAT (5DF, 4FH, 4FH, 3D, 3D) 157
1 N=13,4 H, Z=2E16,7) 158
5 FORMAT (5DF, 4FH, 4FH, 3D, 3D) 159
1 N=13,4 H, Z=2E16,7) 160
6 FORMAT (5DF, 4FH, 4FH, 3D, 3D) 161
1 N=13,4 H, Z=2E16,7) 162
7 FORMAT (5DF, 4FH, 4FH, 3D, 3D) 163
1 N=13,4 H, Z=2E16,7) 164
8 FORMAT (5DF, 4FH, 4FH, 3D, 3D) 165
1 N=13,4 H, Z=2E16,7) 166
9 FORMAT (5DF, 4FH, 4FH, 3D, 3D) 167
1 N=13,4 H, Z=2E16,7) 168
10 FORMAT (5DF, 4FH, 4FH, 3D, 3D) 169
1 N=13,4 H, Z=2E16,7) 170
11 FORMAT (5DF, 4FH, 4FH, 3D, 3D) 171
1 N=13,4 H, Z=2E16,7) 172
12 FORMAT (5DF, 4FH, 4FH, 3D, 3D) 173
1 N=13,4 H, Z=2E16,7) 174
13 END

END
FUNCTION ZERU (Z, N, A, AADD, APHIME, IPRINT)

COMPLEX Z, A, AADD, APHIME, FACT1, FACT2, HA

FACT1 = A(LOAD + APHIME)

FACT2 = A

RATIO = FACT1/FACT2

ZERU = 0.0000000001 - CABS(RATIO)

IF (ABS(ZERU) .LT. 1.0E-08) RETURN

RATIO = REAL(FACT1)/(REAL(FACT2) + (0.0, 1.0) * (AIMAG(FACT1)/AIMAG(FACT2)))

ZERU = 0.0000000001 - CABS(RATIO)

IF (ABS(ZERU) .LT. 1.0E-08) RETURN

IF (IPRINT .EQ. 0) GO TO 1

PRINT 3, FACT1, FACT2

FACT1 = REAL(FACT1)

FACT2 = AIMAG(FACT1)

FACT = REAL(FACT2)

L1HE = FACT1 + AND + 0.0000000000000000

L1LE = FACT1 + AND + 0.0000000000000000

L2HE = FACT2 + AND + 0.0000000000000000

L2LE = FACT2 + AND + 0.0000000000000000

L1HM = FACT1 + AND + 0.0000000000000000

L1LM = FACT1 + AND + 0.0000000000000000

L2HM = FACT2 + AND + 0.0000000000000000

L2LM = FACT2 + AND + 0.0000000000000000

IF ((L1LE .NE. L2LE) .OR. (L1IF .NE. L2IF)) 0 TO 2

ZERU = CABS((L1RN - L2RN) + (0.0, 1.0) * (L1IM - L2IM)) * 1.0E-9

IF (ABS(ZERU) .LT. 1.0E-08) RETURN

IF (IPRINT .EQ. 0) RETURN

PRINT 4, FACT1, L1RE, L1IM, FACT2, L2RE, L2IM, FACT2

1: L2LE = L2IM

RETURN

FORMAT (4L20, 10)

FORMAT (A, FACT1 = 3(020, 3X)) /HH FACT2 = 3(020, 3X) /HH FACT3 = 3(020, 3X)

END
SUBROUTINE SKWU (Z,RATIO,IDIM)
DIMENSION RATIO(IDIM)
COMPLEX RATIO,KUNST,DENOM
KUNST(1)=2.0*L/Z
DO 1 J=1,IDIM
RATIO(J)= ( 1,0 )
1 CONTINUE
I=IDIM-1
RATIO(IDIM)=Z/(2.0*L/Z)
DENOM=KUNST(I)+RATIO(I+1)
IF (DENOM .LE. 3.43) THEN
RATIO(I)=1.0/DENOM
GO TO 3
END
RATIO(I)=1.0E300
I=I-1
IF (I .LE. 0) RETURN
END
SUBROUTINE FHWD (Z, RATIO(I:I), IM, NR)
DIMENSION RATIO(I:IM)
COMPLEX RATIO, H1, Z, KONST
KONST(1) = 2.*T/7
IMAX = IM-1
RATIO(1) = H1
DO 1 I=1, IMAX
RATIO(I+1) = (RATIO(I) * KONST(I) - (1., 0.)) / RATIO(I)
1 CONTINUE
RETURN
END
SUBROUTINE CHESS (Z,YONE,IV,LCHK,IPRINT+)
COMPLEX JZ,J1,YZ,Y1,HZ,H2
COMPLEX Z,YZERO,JONE,YZERO,YONE,ZSU,FACT,FACT,JZADD,J1ADD,EZ,P,G
COMMON /ZEROES/ JZ,J1,YZ,HZ,H2
COMMON /CHKLIST/ CSH,T,CLG,CCOS,CSIN,CEXP
REAL MAX
EQUIVALENCE (JZERO,JZ), (JONE,J1), (YZERO,YZ), (HZERO,H2), (HZONE)

1*H<1
MAX=SINT(Z,0)*(1.0F150)
IVALCHK=11
IF (CABS(Z) <= 10.) 1,1+4
JZERO=(1.,0.)
JONE=(1.,0.)
YZERO=(0.,0.)
YONE=(1.,0.)
FK=1.
FKFACT=1.
ZSU=Z/E25
ZFACT=(1.,0.)
SKINV=1.
ZFACT=ZFACT/ZSU
FACT=ZFACT/FKFACT
JZADD=FACT/JZADD
FK=FK+.1
FKFACT=FKFACT*FK
J1AUD=FACT/J1AUD
JZADD=JZADD/JZADD
JONE=JONE+J1AUD
YZERO=YZERO+JZADD*SKINV
YONE=YONE+J1AUD*(SKINV+SKINV+1./FK)
SKINV=SKINV+1./FK
IF (JZERO.COMPLEX(1,E15),0.) JZERO=CMPLX(1,E-15,0.)
IF (CABS(JZADD/JONE).GT.1.E-25) GO TO 2
IF (CABS(J1AUD/JONE).GT.1.E-25) GO TO 2
JONE=JONE*E5
IF (Z.EQ.(0,,0)) GO TO 3
YZERO=(+577156569+CLG(Z,5))*JZERO+YZERO/1.5709632679
YONE=(+5772156649+CLG(Z,5))*JONE-(1.-ZSU*YONE)/Z/1.5709632579
CBE 1
CBE 2
CBE 3
CBE 4
CBE 5
CBE 6
CBE 7
CBE 8
CBE 9
CBE 10
CBE 11
CBE 12
CBE 13
CBE 14
CBE 15
CBE 16
CBE 17
CBE 18
CBE 19
CBE 20
CBE 21
CBE 22
CBE 23
CBE 24
CBE 25
CBE 26
CBE 27
CBE 28
CBE 29
CBE 30
CBE 31
CBE 32
CBE 33
CBE 34
CBE 35
CBE 36
CBE 37
CBE 38
CBE 39
CBE 40
CBE 41
CBE 42
CBE 43
CBE 44
CBE 45
CBE 46
CBE 47
CBE 48
CBE 49
CBE 50
15
H2ZERO=JZERO*(0.,-1.)*YZERO
HZONE=JONE*(0.,-1.)*YONE
GO TO 12
3
YZERO=(-1.+E300,0.)
YONE=(-1.+E300,0.)
H2ZERO=(-1.+E300,0.)
HZONE=(-1.+E300,0.)
IVALCKH=11
RETURN
K=0
IF (AABS(AIMAG(Z))*LT.1.E-9.AND.REAL(Z).LT.0.) GO TO 5  

GO TO 10  

Z = Z  

K = 1  

FACT = 3.141592653589793*Z  

FACT = COS^1(FAC)  

SIN = COS^1(Z)*FAC  

FACT = (1+1.)*CEAP((0.1,0.)*Z)/FACT  

U = U  

V = (0.1,0)  

F = 1  

P = 1  

W = (U-1,0)/V  

FACT = U  

FACT = FACT*(U-FACT)/EZ/FK  

P = P+FACT  

FACT = FACT*(U-FACT)/EZ/FK  

Q = Q+FACT  

IF (CABS(SQRT(U)).LT.1.E-9.AND.R) GO TO 9  

IF (FK.*LI.21.0) GO TO 10  

IF (U) 10 10 11  

GO TO 7  

J = (P+U)*COS^1(P-Q)*SIN^1  

Y = (P+U)*SIN^1(P-Q)*COS^1  

HZ = HZ + FACT*(P*(0.1,0.)*Q)  

U = U  

GO TO 12  

Z = Z  

YZ = YZ + Z*JZERO*(0.1,0)  

HZ = HZ + Z*(UZERO*(0.1,0)  

J = J + JUNE  

JUNE + JUN + ZUN + (0.1,0)  

HZ = HZ + JUNE*(0.1,0.)*ONE  

IF ((CABS(JZERO)).GT.1.)*OK, (CABS(JUNE)).GT.1.)*OK, (CABS(YZERO)).GT.1.)*OK, (CABS(YOUN)).GT.1.)*OK) VALCHK = 1  

IF (IPN+1,0) 4FTURN  

P = JUNE + JZCH  

Q = JZERO + YOUN*CONST  

WHQ = P/Q  

DIFF + 1.000000000001 + CABS(WHQ)  

CBE 51  

CBE 52  

CBE 53  

CBE 54  

CBE 55  

CBE 56  

CBE 57  

CBE 58  

CBE 59  

CRE 60  

CBE 61  

CBE 62  

CBE 63  

CBE 64  

CBE 65  

CBE 66  

CBE 67  

CBE 68  

CBE 69  

CBE 70  

CBE 71  

CBE 72  

CBE 73  

CBE 74  

CBE 75  

CBE 76  

CBE 77  

CBE 78  

CBE 79  

CBE 80  

CBE 81  

CBE 82  

CBE 83  

CBE 84  

CBE 85  

CBE 86  

CBE 87  

CBE 88  

CBE 89  

CBE 90  

CBE 91  

CBE 92  

CBE 93  

CBE 94  

CBE 95  

CBE 96  

CBE 97  

CBE 98  

CBE 99  

CHE 100
IF (ABS(DIFF) .LT. 1. E-4) RETURN
IF (ABS(DIFF) .GT. 1.0E-6) GO TO 13
PRINT 10, 2
RETURN
13 IF (ABS(DIFF) .LT. 1.0E-4) GO TO 14
PRINT 17, 7
RETURN
14 IF (ABS(DIFF) .GT. 1.0E-2) GO TO 15
PRINT 18, 4
RETURN
15 PRINT 19, 4
RETURN
C
16 FORMAT (66F14.7)!
       "HUNSNIAN CHECK FOR BESSELS, N=0,1, SHOWS AGREEMENT TO"
1 10**-6, (22E16,7)
17 FORMAT (66F14.7)!
       "HUNSNIAN CHECK FOR BESSELS, N=0,1, SHOWS AGREEMENT TO"
1 10**-4, (22E16,7)
18 FORMAT (66F14.7)!
       "HUNSNIAN CHECK FOR BESSELS, N=0,1, SHOWS AGREEMENT TO"
1 10**-2, (22E16,7)
19 FORMAT (74F14.7)!
       "HUNSNIAN CHECK FOR BESSELS, N=0,1, DOES NOT SHOW AGREEMENT TO"
1 10**-2, (22E16,7)
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
A=3.141592653589793
RETURN
3
ARG=0.0
RETURN
4
ARG=2.0*ATAN(Y/(SQR(X*X+Y*Y)+Y))
RETURN
END