```
C*********************************************************
        PROGRAM RIEMANN
C       this is the PHASE TWO algorithm
C       two - track version
C       for holomorphic cusp-forms (March 1994)
C       CRAY VERSION -- SINGLE PRECISION
C       for arbitrary Hecke groups:  G(NGRP)
C*********************************************************
C       Prelim. version by:   Ze Wang (wangz@msi.umn.edu)
C       Advised by:           Prof. Dennis A. Hejhal
C       05/1994, D. Hejhal, Z. Wang, University of Minnesota
C*********************************************************
C
        IMPLICIT REAL(A-H,P-Y)
        IMPLICIT COMPLEX(Z)
        PARAMETER(NPAX=30)
C       here: NPAX= the exact M0-value lifted from file CDAT
        PARAMETER(NA=20,NB=100)
C       these are the desired start and finish N-values
C       make sure these are admissible (cf. IFT, Y11, Y22 below)!!!
        PARAMETER(NC=NB-NA+1)
        PARAMETER(NC2=2*NC)
        PARAMETER(IFT=800)
        PARAMETER(IFT2=2*IFT)
        PARAMETER(IFT4=4*IFT)
C       make certain that IFT > largest effective M-value for
C       level Y11 and Y22 (not merely for Y-min of polygon)
        DIMENSION ZA(IFT4), ZP(IFT4)
C       ZP will correspond to fundamental polygon pullbacks
        DIMENSION R(2,2), RINV(2,2), REC(IFT4,5)
        DIMENSION ZU(NC2,NPAX), RZU(NC2,NPAX), U00(NC2)
        DIMENSION Q11(NC2), Q22(NC2)
        DIMENSION CX(2*NPAX), CDZ(NC2), CDF(NC)
        DIMENSION W9(NPAX), QU(NPAX), QUU(NPAX)
        COMMON /DENNIS/ PI,TCXX,TSXX,PIXX
C
        PI=(4.0E0)*ATAN(1.0E0)
C
C       INPUT DATA:
C       in addition to PARAMETERS at top
        NGRP=6
        KW=8
        TCXX=COS(PI/FLOAT(NGRP))
        TSXX=SIN(PI/FLOAT(NGRP))
        PIXX=PI/TCXX
        Y11=.020E0
        Y22=.022E0
C
        DO 4 I=1,IFT2
          J=I-IFT
          XQW=(FLOAT(J)-.5E0)*(TCXX)/FLOAT(IFT)
          ZA(I)=CMPLX(XQW,Y11)
 4      CONTINUE
        DO 6 I=1,IFT2
          J=I-IFT
          XQW=(FLOAT(J)-.5E0)*(TCXX)/FLOAT(IFT)
          ZA(I+IFT2)=CMPLX(XQW,Y22)
 6      CONTINUE
C
C       END-OF-DATA
C
C       READ THE CDAT-FILE
C
```

```
        OPEN(10,FILE='CDAT',STATUS='OLD')
        DO 5100 J=1,NPAX
          READ(10,100) R33,R44
          CX(J)= R33
          CX(J+NPAX)=R44
 5100   CONTINUE
        CLOSE(10)
C
 100    FORMAT(12X,F20.15,2X,F20.15)
C
        DO 5103 I=1,IFT4
          Z33=ZA(I)
          XA=REAL(Z33)
          YA=AIMAG(Z33)
          CALL IMAGE(NGRP,Z33,ZSTAR,R,RINV)
          ZP(I)=ZSTAR
          REC(I,1)=R(1,1)
          REC(I,2)=R(1,2)
          REC(I,3)=R(2,1)
          REC(I,4)=R(2,2)
          REC(I,5)=ATAN2(R(2,1)*YA, R(2,1)*XA + R(2,2))
 5103   CONTINUE
C
        CALL FUNC(ZA,ZP,REC,KW,ZU,U00)
C
        DO 5107 N=1,NC2
        DO 5108 L=1,NPAX
          RZU(N,L)=REAL(ZU(N,L))
 5108   CONTINUE
 5107   CONTINUE
C
        DO 5121 N=1,NC
          DO 5122 L=1,NPAX
          W9(L)=RZU(N,L)*CX(L)
          TEMP=ABS(RZU(N,L))
          QU(L)=TEMP
          QUU(L)=TEMP**2
 5122     CONTINUE
          CDZ(N)=SSUM(NPAX,W9,1)/U00(N)
          Q11(N)=SSUM(NPAX,QU,1)/U00(N)
          Q22(N)=SQRT(SSUM(NPAX,QUU,1))/U00(N)
 5121   CONTINUE
C
        DO 5131 N=1,NC
          DO 5132 L=1,NPAX
          W9(L)=RZU(N+NC,L)*CX(NPAX+L)
          TEMP=ABS(RZU(N+NC,L))
          QU(L)=TEMP
          QUU(L)=TEMP**2
 5132     CONTINUE
          CDZ(N+NC)=SSUM(NPAX,W9,1)/U00(N+NC)
          Q11(N+NC)=SSUM(NPAX,QU,1)/U00(N+NC)
          Q22(N+NC)=SQRT(SSUM(NPAX,QUU,1))/U00(N+NC)
 5131   CONTINUE
C
        DO 5141 N=1,NC
          CDF(N)=ABS(CDZ(N)-CDZ(N+NC))
 5141   CONTINUE
C
C       now do the final print-out
C
        PRINT*,'   '
        PRINT*,'BASIC PARAMETERS:'
```

```
        PRINT 90, NGRP, KW
        PRINT 91, IFT
        PRINT 92, AIMAG(ZA(1)), AIMAG(ZA(1+IFT2))
        PRINT 95, NPAX
 90     FORMAT('NGRP=',I3,1X,'WEIGHT K=',I3)
 91     FORMAT('FINITE FOURIER Q=',I6)
 92     FORMAT('Y-VALUES:',F9.6,3X,F9.6)
 95     FORMAT('the M0-value is:',I4)
C
        PRINT*, '  '
        PRINT 893
 893    FORMAT(' order: coef, diff, control Q11a, control Q11b')
        PRINT*,'RENORMALIZED REAL PARTS:'
        PRINT 96, (N+NA-1, CDZ(N), CDZ(N+NC), CDF(N), Q11(N),
     &  Q11(N+NC), N=1,NC)
 96     FORMAT('C(',I5,')=',F17.12,2X,F17.12,2X,E9.2,1X,E9.2,1X,E9.2)
C
        PRINT*, '  '
        PRINT 993
 993    FORMAT('THE ORIGINAL COEFFICIENTS WERE:')
        PRINT 994, (I,CX(I),CX(NPAX+I),I=1,NPAX)
 994    FORMAT('C(',I5,')=',F17.12,2X,F17.12)
        END
C
C
        SUBROUTINE FUNC(ZA,ZP,REC,KW,ZU,U00)
C       CRAY VERSION -- SINGLE PRECISION
C       wherein we calculate the matrix ZU (sub n, sub ell)
        IMPLICIT REAL(A-B,D-H,P-Y)
        IMPLICIT COMPLEX(C,Z)
        INTEGER IFT
        PARAMETER(NPAX=30)
        PARAMETER(NA=20,NB=100)
        PARAMETER(NC=NB-NA+1)
        PARAMETER(NC2=2*NC)
C
        PARAMETER(IFT=800)
        PARAMETER(IFT2=2*IFT)
        PARAMETER(IFT4=4*IFT)
        DIMENSION ZA(IFT4), ZP(IFT4), ZU(NC2,NPAX), REC(IFT4,5)
        DIMENSION ZREV(IFT4,NPAX), CW(IFT2), U00(NC2)
        COMMON/ DENNIS/ PI,TCXX,TSXX,PIXX
C
C
        AIFT=1.0E0/FLOAT(IFT2)
        Y11 = AIMAG(ZA(1))
        Y22=AIMAG(ZA(1+IFT2))
C
        DO 51 N=NA,NB
          TMP=FLOAT(N)*Y11
          CALL CALCR(KW,TMP,TR)
          NT=N-NA+1
          U00(NT)=SQRT(Y11)*TR
 51     CONTINUE
        DO 52 N=NA,NB
          TMP=FLOAT(N)*Y22
          CALL CALCR(KW,TMP,TR)
          NT=N-NA+1
          U00(NC+NT)=SQRT(Y22)*TR
 52     CONTINUE
C
        DO 200 J=1,IFT4
        YPJ=AIMAG(ZP(J))
```

```
        B98=SQRT(YPJ)
        B99= -FLOAT(KW)*REC(J,5)
          DO 205 L=1,NPAX
          Z=FLOAT(L)*ZP(J)
          CALL CALC(KW,Z,ZT)
          ZREV(J,L)=B98*ZT*CEXP(CMPLX(0.0E0,B99))
 205      CONTINUE
 200    CONTINUE
C
        DO 300 N=NA,NB
        DO 310 L=1,NPAX
          DO 320 J=1,IFT2
          TMP= -PIXX*FLOAT(N)*REAL(ZA(J))
          ZTMP=CMPLX(0.0E0, TMP)
          CW(J)=ZREV(J,L)*CEXP(ZTMP)
 320    CONTINUE
        Z99=CSUM(IFT2,CW,1)*AIFT
        NT=N-NA+1
        ZU(NT,L)=Z99
 310    CONTINUE
 300    CONTINUE
C
        DO 400 N=NA,NB
        DO 410 L=1,NPAX
          DO 420 J=1,IFT2
          TMP= -PIXX*FLOAT(N)*REAL(ZA(J+IFT2))
          ZTMP=CMPLX(0.0E0, TMP)
          CW(J)=ZREV(J+IFT2,L)*CEXP(ZTMP)
 420    CONTINUE
        Z99=CSUM(IFT2,CW,1)*AIFT
        NT=N-NA+1
        ZU(NC+NT,L)=Z99
 410    CONTINUE
 400    CONTINUE
C
        RETURN
        END
C
C
        SUBROUTINE CALC(KW,Z,ZT)
CCC     ZT=Im(Z)^((KW-1)/2) * EXP(i*2PI*Z/2*TCXX)
        IMPLICIT REAL(A-B,D-H,P-Y)
        IMPLICIT COMPLEX(C,Z)
        COMMON/ DENNIS/ PI,TCXX,TSXX,PIXX
        CTMP= CMPLX(0.0E0, 1.0E0)*PIXX*Z
        B91=(.5E0)*FLOAT(KW-1)
        B92=AIMAG(Z)**B91
        ZT=B92*CEXP(CTMP)
        RETURN
        END
C
C
        SUBROUTINE CALCR(KW,Y,YR)
CCC     YR=Y^((KW-1)/2) * EXP(-2PI*Y/2*TCXX)
        IMPLICIT REAL(A-B,D-H,P-Y)
        IMPLICIT COMPLEX(C,Z)
        COMMON/ DENNIS/ PI,TCXX,TSXX,PIXX
        TMP= -Y*PIXX
        B91=(.5E0)*FLOAT(KW-1)
        B92=Y**B91
        YR=B92*EXP(TMP)
        RETURN
        END
```

```
C
C
      SUBROUTINE IMAGE(N,Z,ZSTAR,R,RINV)
C
C     MAP Z TO ZSTAR
C     N --> GROUP, Z --> INPUT COMPLEX
C     ZSTAR --> OUTPUT COMPLEX, R --> OUTPUT 2 BY 2 MATRIX
C     RINV --> INVERSE OF R
C
      IMPLICIT REAL(A-B,D-H,P-Y)
      IMPLICIT COMPLEX(C,Z)
      REAL  LH
      DIMENSION P(2,2),Q(2,2),R(2,2),RINV(2,2)
      COMMON/ DENNIS/ PI,TCXX,TSXX,PIXX
      NGRP=N
      X=REAL(Z)
      Y=AIMAG(Z)
      LH=(2.0E0)*TCXX
      XD=-ANINT(X/LH)*LH
      X=X+XD
      Q(1,1)=1.0E0
      Q(1,2)=XD
      Q(2,1)=0.0E0
      Q(2,2)=1.0E0

 100  SS=(X**2)+(Y**2)
      IF(SS.LT.(1.0E0)) THEN
         XNEW=-X/SS
         YNEW=Y/SS
         P(1,1)=0.0E0
         P(1,2)=-1.0E0
         P(2,1)=1.0E0
         P(2,2)=0.0E0
         CALL MMULT(2,P,Q,R)
         CALL COPYM(2,R,Q)
C
         XD=-ANINT(XNEW/LH)*LH
         X=XNEW+XD
         Y=YNEW
         P(1,1)=1.0E0
         P(1,2)=XD
         P(2,1)=0.0E0
         P(2,2)=1.0E0
         CALL MMULT(2,P,Q,R)
         CALL COPYM(2,R,Q)
C
         GOTO 100
      ENDIF
C
      ZSTAR=CMPLX(X,Y)
      CALL COPYM(2,Q,R)
      RINV(1,1)=R(2,2)
      RINV(1,2)=-R(1,2)
      RINV(2,1)=-R(2,1)
      RINV(2,2)=R(1,1)
C
      RETURN
      END
C
C
      SUBROUTINE MMULT(N,P,Q,R)
C
C     MULTIPLY TWO SQUARE MATRICES    P X Q --> R
```

```
C     N --> DIMENSION, P & Q --> INPUT MATRICES, R --> OUTPUT MATRIX
C
      IMPLICIT REAL(A-B,D-H,P-Y)
      IMPLICIT COMPLEX(C,Z)
      DIMENSION P(N,N),Q(N,N),R(N,N)
      DO I=1,N
         DO J=1,N
            R(I,J)=0.0E0
            DO K=1,N
               R(I,J)=R(I,J)+P(I,K)*Q(K,J)
            ENDDO
         ENDDO
      ENDDO
      RETURN
      END
C
C
      SUBROUTINE COPYM(N,A,B)
C
C     COPY THE SQUARE MATRIX    A --> B
C     N --> DIMENSION, A --> INPUT MATRIX, B --> OUTPUT MATRIX
C
      IMPLICIT REAL(A-B,D-H,P-Y)
      IMPLICIT COMPLEX(C,Z)
      DIMENSION A(N,N),B(N,N)
      DO I=1,N
         DO J=1,N
            B(I,J)=A(I,J)
         ENDDO
      ENDDO

      RETURN
      END
C
C
```