

```

c      +-----+
c      + helplib :                                     +
c      +-----+

c      ----- subroutine rms : compute rms -----
      subroutine hrms(ndim,a,b,rms)
      dimension a(ndim),b(ndim)
      rms = 0.
      do n=1,ndim
        rms = rms + (a(n)-b(n))**2
      enddo
      rms = rms/ndim
      rms = sqrt(rms)
      end

c      ----- subroutine hab2c : maxtrix inner product -----
      subroutine hab2c(idim,jdim,kdim,a,b,c)
      dimension a(idim,jdim),b(jdim,kdim),c(idim,kdim)
      do i = 1,idim
        do k = 1,kdim
          c(i,k) = 0.
        do j =1,jdim
          c(i,k) = c(i,k) + a(i,j)*b(j,k)
        enddo
        enddo
        enddo
        enddo

c      ----- subroutine hcpa2b : copy a to b -----
      subroutine hcpa2b(ndim,a,b)
      dimension a(ndim),b(ndim)
      do n = 1,ndim
        b(n) = a(n)
      enddo
      end

c      ----- subroutine hseta2c : set a to constant -----
      subroutine hseta2c(ndim,a,c)
      dimension a(ndim)
      do n = 1,ndim
        a(n) = c
      enddo
      end

c      ----- subroutine statis : min and max of a field -----
      subroutine statis(nam,idim,jdim,fld)
      character(*) nam
      dimension fld(idim,jdim)
      fmin=fld(1,1)
      fmax=fld(1,1)
      do j=1,jdim
        do i=1,idim
          fmin=amin1(fmin,fld(i,j))
          fmax=amax1(fmax,fld(i,j))
        enddo
      enddo
      write(6,'(a,2f12.2)')nam,fmin,fmax
      return
      end

c      ----- subroutine linds : matrix inversion -----
      subroutine linds(n,co,nduml,coi,ndum2,w)
      integer n,nduml,ndum2
      real co(n,n),coi(n,n),w(n,n)

      integer i,j,indx(n)
      real d

      do j=1,n
        do i=1,n
          w(i,j) = co(i,j)
          coi(i,j) = 0.
        enddo
        coi(j,j) = 1.
      enddo
      call ludcmp(w,n,n,indx,d)
      do j=1,n
        call lubksb(w,n,n,indx,coi(1,j))
      enddo
      return
      end

c      ----- subroutine used by linds -----
      subroutine ludcmp(a,n,np,indx,d)
      parameter (tiny=1.0e-20)
      dimension a(np,np),indx(n),vv(n)
      d=1.
      do i=1,n
        aamax=0.
        do j=1,n
          if (abs(a(i,j)).gt.aamax) aamax=abs(a(i,j))
        enddo
        if (aamax.eq.0.) pause 'singular matrix.'
        vv(i)=1./aamax
      enddo
      do j=1,n
        if (j.gt.1) then
          do i=1,j-1
            sum=a(i,j)
            if (i.gt.1)then
              do k=1,i-1
                sum=sum-a(i,k)*a(k,j)
              enddo
            a(i,j)=sum
          endif
        enddo
      enddo

```

```

      endif
      aamax=0.
      do i=j,n
        sum=a(i,j)
        if (j.gt.1)then
          do k=1,j-1
            sum=sum-a(i,k)*a(k,j)
          enddo
          a(i,j)=sum
        endif
        dum=vv(i)*abs(sum)
        if (dum.ge.aamax) then
          imax=i
          aamax=dum
        endif
      enddo
      if (j.ne.imax)then
        do k=1,n
          dum=a(imax,k)
          a(imax,k)=a(j,k)
          a(j,k)=dum
        enddo
        d=-d
        vv(imax)=vv(j)
      endif
      indx(j)=imax
      if(j.ne.n)then
        if(a(j,j).eq.0.)a(j,j)=tiny
        dum=1./a(j,j)
        do i=j+1,n
          a(i,j)=a(i,j)*dum
        enddo
      endif
    enddo
    if(a(n,n).eq.0.)a(n,n)=tiny
    return
    end

c      ----- subroutine used by linds -----
      subroutine lubksb(a,n,np,indx,b)
      dimension a(np,np),indx(n),b(n)
      ii=0
      do i=1,n
        ll=indx(i)
        sum=b(ll)
        b(ll)=b(i)
        if (ii.ne.0)then
          do j=ii,i-1
            sum=sum-a(i,j)*b(j)
          enddo
        else if (sum.ne.0.) then
          ii=i
        endif
        b(i)=sum
      enddo
      do i=n,1,-1
        sum=b(i)
        if(i.lt.n)then
          do j=i+1,n
            sum=sum-a(i,j)*b(j)
          enddo
        endif
        b(i)=sum/a(i,i)
      enddo
      return
      end

```