ソースファイル名：lbmtherm.for，実行ファイル名：lbmtherm.out

```
$ cp    ../common/lbmtherm_ori.for    .
$ cp    lbmtherm_ori.for    lbmtherm.for
$ vi    lbmtherm.for
$ gfortran    -o    lbmtherm.out      lbmtherm.for
$ ./ lbmtherm.out
```

```
      program lbm_therm

      real*8    cx(9), cy(9)
      real*8    f(9,120,120), f0(9,120,120), tmpf(9,120,120)
      real*8    g(9,120,120), g0(9,120,120), tmpg(9,120,120)
      real*8    u(120,120), v(120,120), rho(120,120), e(120,120)
      real*8    fx(120,120), fy(120,120), fi(9,120,120)
      real*8    u0, v0, ut, ub, ul, ur, vt, vb, vl, vr
      real*8    rho0, e0, et, eb, el, er, em, umax, umin
      real*8    u2, tmp, c, tauf, ntime, taug, kap, mu, pr, ra, h, beta
      real*8    tmp1, tmp2, tmp3
      integer i, j, k, m, ii, jj, nx, ny, nfnl1, nfnl2
      character a(120,120), b(120,120)

      nfnl1 = 100; nfnl2 = 10; ntime = 0.;
      nx = 36; ny = 36; h = ny - 1
      pr = 0.71;
      ra = 10000.;

      u0 =    0.; v0 = 0.; rho0 = 1.0;
      ut =    0.; ub = 0.; ul     =    0.; ur = 0.;
      vt =    0.; vb = 0.; vl     =    0.; vr = 0.;
      et =    0.; eb = 0.; el     =    0.; er = 1.;
      em = 0.5; e0 = 0.5*(el + er)

      beta = 0.1/(el - er)/h
      kap = sqrt(0.1*h**2/pr/ra);        taug = kap*1.5 + 0.5
      mu = pr*kap;                        tauf    = 3.*mu      + 0.5

      c = 1.
      cx(1) =    0.0; cy(1) =    0.0
      cx(2) =    □; cy(2) =    □
      cx(3) =    □; cy(3) =    □
      cx(4) =    □; cy(4) =    □
```

```
cx(5) =   -c ;   cy(5) =   c
cx(6) = -1.0; cy(6) =
cx(7) = -1.0; cy(7) =
cx(8) =   0.0; cy(8) =
cx(9) =   1.0; cy(9) =
```

```
c.. initial condition
      do i = 1,nx; do j = 1,ny
        u(i,j) = u0; v(i,j) = v0; rho(i,j) = rho0;
        e(i,j) = (er - el)*(i - 1)/(nx - 1) + el
       end do; end do

      do i = 1,nx
       u(i,1 ) = ub; v(i,1 ) = vb;   e(i,1 ) = eb
       u(i,ny) = ut; v(i,ny) = vt;   e(i,ny) = et
      end do
      do j = 1,ny
       u(1 ,j) = ul; v(1 ,j) = vl;   e(1 ,j) = el
       u(nx,j) = ur; v(nx,j) = vr;   e(nx,j) = er
      end do

     do i = 1,nx
      do j = 1,ny
       u2 = u(i,j)**2 + v(i,j)**2
       f0(1,i,j) = rho(i,j)*(1. - 3./2.*u2)*4./9.
       do k = 1,4
        m = k*2       ; tmp = cx(m)*u(i,j) + cy(m)*v(i,j)
        f0(m,i,j) =
        m = k*2 + 1; tmp = cx(m)*u(i,j) + cy(m)*v(i,j)
        f0(m,i,j) = rho(i,j)*(1. + 3.*tmp
     .                  + (9./2.*tmp**2 - 3./2.*u2))/36.
       end do
      end do
     end do

     do i = 1,nx
      do j = 1,ny
       u2 = u(i,j)**2 + v(i,j)**2
       g0(1,i,j) =-rho(i,j)*e(i,j)*u2*2./3.
       do k = 1,4
```

```
      m = k*2       ; tmp = cx(m)*u(i,j) + cy(m)*v(i,j)
      g0(m,i,j) = [redacted]
      m = k*2 + 1; tmp = cx(m)*u(i,j) + cy(m)*v(i,j)
      g0(m,i,j) = rho(i,j)*e(i,j)*(3. + 6.*tmp
     .                + 9./2.*tmp**2 - 3./2.*u2)/36.
    end do
   end do
  end do

  do k = 1,9; do i = 1,nx; do j = 1,ny
   f(k,i,j) = f0(k,i,j)
  end do; end do; end do

  do k = 1,9; do i = 1,nx; do j = 1,ny
   g(k,i,j) = g0(k,i,j)
  end do; end do; end do


100   do n1 = 1, nfnl1; do n2 = 1, nfnl2

   ntime = ntime + 1

   do i = 1,nx;do j = 1,ny
    fx(i,j) =    0.; fy(i,j) = beta*(e(i,j) - em)
   end do; end do

  do i = 1,nx;do j = 1,ny
   u(i,j) = 0; v(i,j) = 0; rho(i,j) = f(1,i,j); e(i,j) = g(1,i,j);
   do k = 2,9
     u(i,j) = [redacted]
     v(i,j) = [redacted]
   rho(i,j) = [redacted]
     e(i,j) = [redacted]
    end do
   end do; end do

  do i = 1,nx; do j = 1,ny
   if (rho(i,j) .ne. 0.) then
    u(i,j) = u(i,j)/rho(i,j)
    v(i,j) = v(i,j)/rho(i,j)
```

```
    e(i,j) = e(i,j)/rho(i,j)
   else
    u(i,j)   = 0.
    v(i,j)   = 0.
    e(i,j)   = 0.
   endif
end do; end do

do i = 1,nx; do j = 1,ny
    v(i,j) =    v(i,j) + beta*(e(i,j) - em)*0.5
end do; end do

do i = 1,nx
  do j = 1,ny
    u2 = u(i,j)**2 + v(i,j)**2
    f0(1,i,j) = rho(i,j)*(1. - 3./2.*u2)*4./9.
    do k = 1,4
      m = k*2       ; tmp = cx(m)*u(i,j) + cy(m)*v(i,j)
      f0(m,i,j) =
      m = k*2 + 1; tmp = cx(m)*u(i,j) + cy(m)*v(i,j)
      f0(m,i,j) = rho(i,j)*(1. + 3.*tmp
     .                  + (9./2.*tmp**2 - 3./2.*u2))/36.
    end do
  end do
end do
do i = 1,nx
  do j = 1,ny
    u2 = u(i,j)**2 + v(i,j)**2
    g0(1,i,j) =-rho(i,j)*e(i,j)*u2*2./3.
    do k = 1,4
      m = k*2       ; tmp = cx(m)*u(i,j) + cy(m)*v(i,j)
      g0(m,i,j) =
      m = k*2 + 1; tmp = cx(m)*u(i,j) + cy(m)*v(i,j)
      g0(m,i,j) = rho(i,j)*e(i,j)*(3. + 6.*tmp
     .                  + 9./2.*tmp**2 - 3./2.*u2)/36.
    end do
  end do
end do

do i = 1,nx
```

```
      do j = 1,ny
       tmp1 = u(i,j)*fx(i,j) + v(i,j)*fy(i,j)
       fi(1,i,j) = 4./9.*(1. - 0.5/tauf)*rho(i,j)*(-3.*tmp1)
       do k = 1,4
        m = k*2
        tmp2 =    cx(m)*fx(i,j) +         cy(m)*fy(i,j)
        tmp3 = u(i,j)*fx(i,j)*cx(m)*cx(m)
     .             + u(i,j)*fy(i,j)*cx(m)*cy(m)
     .             + v(i,j)*fx(i,j)*cy(m)*cx(m)
     .             + v(i,j)*fy(i,j)*cy(m)*cy(m)
        fi(m,i,j) = 1./9.*(1. - 0.5/tauf)*rho(i,j)*
     .             (3.*tmp2 + 9.*tmp3 - 3.*tmp1)
        m = k*2 + 1
        tmp2 =      cx(m)*fx(i,j) +         cy(m)*fy(i,j)
        tmp3 = u(i,j)*fx(i,j)*cx(m)*cx(m)
     .             + u(i,j)*fy(i,j)*cx(m)*cy(m)
     .             + v(i,j)*fx(i,j)*cy(m)*cx(m)
     .             + v(i,j)*fy(i,j)*cy(m)*cy(m)
        fi(m,i,j) = 1./36.*(1. - 0.5/tauf)*rho(i,j)*
     .             (3.*tmp2 + 9.*tmp3 - 3.*tmp1)
       end do
      end do
     end do

c..  collision
      do k = 1,9; do i = 1,nx; do j = 1,ny
       f(k,i,j) = f(k,i,j) - (f(k,i,j) - [        ])/tauf + fi(k,i,j)
       g(k,i,j) = g(k,i,j) - (g(k,i,j) - [        ])/taug
      end do; end do; end do

c..  streaming
      do k = 1,9; do i = 1,nx; do j = 1,ny
       tmpf(k,i,j) = f(k,i,j); tmpg(k,i,j) = g(k,i,j)
      end do; end do; end do

      do k = 1,9
       if(k .eq. 1) then
        do i = 1,nx; do j = 1,ny
          ii = i       ; jj = j
        f(k,ii,jj) = tmpf(k,i,j)
```

```
    g(k,ii,jj) = tmpg(k,i,j)
  end do; end do

else if(k .eq. 2) then
  do i = 1,nx - 1; do j = 1,ny
    ii = i + 1; jj = j
    f(k,ii,jj) = tmpf(k,i,j)
    g(k,ii,jj) = tmpg(k,i,j)
  end do; end do

else if(k .eq. 3) then
  do i = 1,nx - 1; do j = 1,ny - 1
    ii = [    ]; jj = [    ]
    f(k,ii,jj) = tmpf(k,i,j)
    g(k,ii,jj) = tmpg(k,i,j)
  end do; end do

else if(k .eq. 4) then
  do i = 1,nx; do j = 1,ny - 1
    ii = i       ; jj = [    ]
    f(k,ii,jj) = tmpf(k,i,j)
    g(k,ii,jj) = tmpg(k,i,j)
  end do; end do

else if(k .eq. 5) then
  do i = 2,nx; do j = 1,ny - 1
    ii = [    ]; jj = [    ]
    f(k,ii,jj) = tmpf(k,i,j)
    g(k,ii,jj) = tmpg(k,i,j)
  end do; end do

else if(k .eq. 6) then
  do i = 2,nx; do j = 1,ny
    ii = [     ]; jj = [    ]
    f(k,ii,jj) = tmpf(k,i,j)
    g(k,ii,jj) = tmpg(k,i,j)
  end do; end do

else if(k .eq. 7) then
  do i = 2,nx; do j = 2,ny
```

```
      ii = [ ]; jj = [ ]
      f(k,ii,jj) = tmpf(k,i,j)
      g(k,ii,jj) = tmpg(k,i,j)
     end do; end do

    else if(k .eq. 8) then
     do i = 1,nx; do j = 2,ny
      ii = [ ]; jj = [ ]
      f(k,ii,jj) = tmpf(k,i,j)
      g(k,ii,jj) = tmpg(k,i,j)
     end do; end do

    else if(k .eq. 9) then
     do i = 1,nx-1; do j = 2,ny
      ii = [ ]; jj = [ ]
      f(k,ii,jj) = tmpf(k,i,j)
      g(k,ii,jj) = tmpg(k,i,j)
     end do; end do
    end if
   end do

c..  boundary condition
     do j = 1,ny
      g(3,1,j ) =-g(7,1,j )
     . + 2.*(f(3,1,j ) + f(7,1,j ))
     .     + rho(1,j )*(1. +      (ul**2 + 3.*ul*vl + vl**2))*el/6.
     .     - rho(1,j )*(1. + 3.*(ul**2 + 3.*ul*vl + vl**2))/9.
      g(2,1,j ) =-g(6,1,j )
     .   +      (f(2,1,j ) + f(6,1,j ))
     .     + rho(1,j )*(1. +      (2.*ul**2          - vl**2))*el/3.
     .     - rho(1,j )*(2. + 3.*(2.*ul**2           - vl**2))/9.
      g(9,1,j ) =-g(5,1,j )              + 2.*(f(9,1,j ) + f(5,1,j ))
     .     + rho(1,j )*(1. +      (ul**2 - 3.*ul*vl + vl**2))*el/6.
     .     - rho(1,j )*(1. + 3.*(ul**2 - 3.*ul*vl + vl**2))/9.

      g(7,nx,j ) =-g(3,nx,j )
     . + 2.*(f(3,nx,j ) + f(7,nx,j ))
     .     + rho(nx,j )*(1. +      (ur**2 + 3.*ur*vr + vr**2))*er/6.
     .     - rho(nx,j )*(1. + 3.*(ur**2 + 3.*ur*vr + vr**2))/9.
      g(6,nx,j ) =-g(2,nx,j )
```

```
     .       +       (f(2,nx,j ) + f(6,nx,j ))
     .         + rho(nx,j )*(1. +       (2.*ur**2                - vr**2))*er/3.
     .         - rho(nx,j )*(2. + 3.*(2.*ur**2                  - vr**2))/9.
           g(5,nx,j ) =-g(9,nx,j )                              + 2.*(f(9,nx,j ) + f(5,nx,j ))
     .         + rho(nx,j )*(1. +       (ur**2 - 3.*ur*vr + vr**2))*er/6.
     .         - rho(nx,j )*(1. + 3.*(ur**2 - 3.*ur*vr + vr**2))/9.
           end do
           do i = 1,nx
            eb = 4.*e(i,      2)/3. - e(i,      3)/3.
            et = 4.*e(i,ny-1)/3. - e(i,ny-2)/3.
c            eb = e(i,      2)
c            et = e(i,ny-1)

           g(3,i,1 ) =-g(7,i,1 )
     .     + 2.*(f(3,i,1 ) + f(7,i,1 ))
     .         + rho(i,1 )*(1. +       (ub**2 + 3.*ub*vb + vb**2))*eb/6.
     .         - rho(i,1 )*(1. + 3.*(ub**2 + 3.*ub*vb + vb**2))/9.
           g(4,i,1 ) =-g(8,i,1 )     +       (f(4,i,1 ) + f(8,i,1 ))
     .         + rho(i,1 )*(1. -       (ub**2                   - 2.*vb**2))*eb/3.
     .         - rho(i,1 )*(2. - 3.*(ub**2                      - 2.*vb**2))/9.
           g(5,i,1 ) =-g(9,i,1 )                + 2.*(f(5,i,1 ) + f(9,i,1 ))
     .         + rho(i,1 )*(1. +       (ub**2 - 3.*ub*vb + vb**2))*eb/6.
     .         - rho(i,1 )*(1. + 3.*(ub**2 - 3.*ub*vb + vb**2))/9.

           g(7,i,ny) =-g(3,i,ny)                                + 2.*(f(3,i,ny) + f(7,i,ny))
     .         + rho(i,ny)*(1. +       (ut**2 + 3.*ut*vt + vt**2))*et/6.
     .         - rho(i,ny)*(1. + 3.*(ut**2 + 3.*ut*vt + vt**2))/9.
           g(8,i,ny) =-g(4,i,ny)                                +       (f(4,i,ny) + f(8,i,ny))
     .         + rho(i,ny)*(1. -       (ut**2                     - 2.*vt**2))*et/3.
     .         - rho(i,ny)*(2. - 3.*(ut**2                      - 2.*vt**2))/9.
           g(9,i,ny) =-g(5,i,ny)                                + 2.*(f(5,i,ny) + f(9,i,ny))
     .         + rho(i,ny)*(1. +       (ut**2 - 3.*ut*vt + vt**2))*et/6.
     .         - rho(i,ny)*(1. + 3.*(ut**2 - 3.*ut*vt + vt**2))/9.
           end do
c.. corner
           do n = 1,4
            if(n .eq. 1) then                    ! Bot - Lef
             i =    1; j = 1; k1 = 5; k2 = 9
             utmp = (ub+ul)*0.5; vtmp = (vb+vl)*0.5; etmp = (eb+el)*0.5;
            else if(n .eq. 2) then           ! Top - Rig
```

```
      i = nx; j = ny; k1 = 5; k2 = 9
      utmp = (ut+ur)*0.5; vtmp = (vt+vr)*0.5; etmp = (et+er)*0.5;
    else if(n .eq. 3) then          ! Top - Lef
      i =    1; j = ny; k1 = 3; k2 = 7
      utmp = (ut+ul)*0.5; vtmp = (vt+vl)*0.5; etmp = (et+el)*0.5;
    else if(n .eq. 4) then          ! Bot - Rig
      i = nx; j =    1; k1 = 3; k2 = 7
      utmp = (ub+ur)*0.5; vtmp = (vb+vr)*0.5; etmp = (eb+er)*0.5;
    endif
    u2 = utmp**2 + vtmp**2;
    tmp = cx(k1)*utmp + cy(k1)*vtmp
    g(k1,i,j) = rho(i,j)*etmp*(3. + 6.*tmp
   .                + 9./2.*tmp**2 - 3./2.*u2)/36.
    tmp = cx(k2)*utmp + cy(k2)*vtmp
    g(k2,i,j) = rho(i,j)*etmp*(3. + 6.*tmp
   .                + 9./2.*tmp**2 - 3./2.*u2)/36.

    enddo


    do j = 1,ny
     rho(1,j) =(f(1,1,j) + f(4,1,j) + f(8,1,j)
   .         + 2.*(f(5,1,j) + f(6,1,j) + f(7,1,j)))/(1. - ul)
     f(3,1,j) = f(7,1,j) + 0.5*(f(8,1,j) - f(4,1,j))
   .                + rho(1,j)*(ul/6. + vl/2.)
     f(2,1,j) = f(6,1,j) + rho(1,j)*ul*2./3.
     f(9,1,j) = f(5,1,j) - 0.5*(f(8,1,j) - f(4,1,j))
   .                + rho(1,j)*(ul/6. - vl/2.)

     rho(nx,j) =(f(1,nx,j) + f(4,nx,j) + f(8,nx,j)
   .         + 2.*(f(3,nx,j) + f(2,nx,j) + f(9,nx,j)))/(1. + ur)
     f(7,nx,j) = f(3,nx,j) + 0.5*(f(4,nx,j) - f(8,nx,j))
   .                - rho(nx,j)*( ur/6. + vr/2.)
     f(6,nx,j) = f(2,nx,j) - rho(nx,j)*ur*2./3.
     f(5,nx,j) = f(9,nx,j) - 0.5*(f(4,nx,j) - f(8,nx,j))
   .                - rho(nx,j)*( ur/6. - vr/2.)
    end do
    do i = 1,nx
     rho(i,1) =(f(1,i,1) + f(2,i,1) + f(6,i,1)
   .         + 2.*(f(7,i,1) + f(8,i,1) + f(9,i,1)))/(1. - vb)
```

```
      f(3,i,1) = f(7,i,1) - 0.5*(f(2,i,1) - f(6,i,1))
     .                    + rho(i,1)*( ub/2. + vb/6.)
      f(4,i,1) = f(8,i,1) - rho(i,1)*vb*2./3.
      f(5,i,1) = f(9,i,1) + 0.5*(f(2,i,1) - f(6,i,1))
     .                    + rho(i,1)*(-ub/2. + vb/6.)

      rho(i,ny) =(f(1,i,ny) + f(2,i,ny) + f(6,i,ny)
     .          + 2.*(f(3,i,ny) + f(4,i,ny) + f(5,i,ny)))/(1. + vt)
      f(7,i,ny) = f(3,i,ny) + 0.5*(f(2,i,ny) - f(6,i,ny))
     .                     - rho(i,ny)*( ut/2. + vt/6.)
      f(8,i,ny) = f(4,i,ny) - rho(i,ny)*vt*2./3.
      f(9,i,ny) = f(5,i,ny) - 0.5*(f(2,i,ny) - f(6,i,ny))
     .                     - rho(i,ny)*(-ut/2. + vt/6.)
    end do
c.. corner
    do n = 1,2
     if(n .eq. 1) then
      i = 1; j = 1;
      utmp = (ub + ul)*0.5; vtmp = (vb + vl)*0.5;
     else
      i = nx; j = ny;
      utmp = (ut + ur)*0.5; vtmp = (vt + vr)*0.5;
     endif
     u2 = utmp**2 + vtmp**2;
     do m = 1,2
      if(m .eq. 1) k = 5
      if(m .eq. 2) k = 9
      tmp = cx(k)*utmp + cy(k)*vtmp
      f(k,i,j) = rho(i,j)*(1. + 3.*tmp
     .                     + (9./2.*tmp**2 - 3./2.*u2))/36.
     enddo
    enddo
    do n = 1,2
     if(n .eq. 1) then
      i = 1; j = ny;
      utmp = (ut + ul)*0.5; vtmp = (vt + vl)*0.5;
     else
      i = nx; j = 1;
      utmp = (ub + ur)*0.5; vtmp = (vb + vr)*0.5;
     endif
```

```
      u2 = utmp**2 + vtmp**2;
      do m = 1,2
       if(m .eq. 1) k = 3
       if(m .eq. 2) k = 5
       tmp = cx(k)*utmp + cy(k)*vtmp
       f(k,i,j) = rho(i,j)*(1. + 3.*tmp
     .                    + (9./2.*tmp**2 - 3./2.*u2))/36.
      enddo
     enddo

    end do

c.. graphics

      do i = 1, nx; do j = 1, ny
        a(i,j)= '9'; b(i,j)= '9'
      end do; end do
      write(*,*) ntime

      do i = 1, nx; do j = 1, ny
       if(e(i,j) .le. er*0.9 + el*0.1) a(i,j)= '8'
       if(e(i,j) .le. er*0.8 + el*0.2) a(i,j)= '7'
       if(e(i,j) .le. er*0.7 + el*0.3) a(i,j)= '6'
       if(e(i,j) .le. er*0.6 + el*0.4) a(i,j)= '5'
       if(e(i,j) .le. er*0.5 + el*0.5) a(i,j)= '4'
       if(e(i,j) .le. er*0.4 + el*0.6) a(i,j)= '3'
       if(e(i,j) .le. er*0.3 + el*0.7) a(i,j)= '2'
       if(e(i,j) .le. er*0.2 + el*0.8) a(i,j)= '1'
       if(e(i,j) .le. er*0.1 + el*0.9) a(i,j)= '0'
      end do; end do

      umax = 0.; umin = 100.;
      do i = 1, nx; do j = 1, ny
       u2 = sqrt(u(i,j)**2 + v(i,j)**2)
       if(u2 .gt. umax) umax = u2
       if(u2 .lt. umin) umin = u2
      end do; end do
      do i = 1, nx; do j = 1, ny
       u2 = sqrt(u(i,j)**2 + v(i,j)**2)
       if(u2 .le. umax*0.9 + umin*0.1) b(i,j)= '8'
```

```
        if(u2 .le. umax*0.8 + umin*0.2) b(i,j)= '7'
        if(u2 .le. umax*0.7 + umin*0.3) b(i,j)= '6'
        if(u2 .le. umax*0.6 + umin*0.4) b(i,j)= '5'
        if(u2 .le. umax*0.5 + umin*0.5) b(i,j)= '4'
        if(u2 .le. umax*0.4 + umin*0.6) b(i,j)= '3'
        if(u2 .le. umax*0.3 + umin*0.7) b(i,j)= '2'
        if(u2 .le. umax*0.2 + umin*0.8) b(i,j)= '1'
        if(u2 .le. umax*0.1 + umin*0.9) b(i,j)= '0'
      end do; end do

      do j = ny,1,-1
       write(*,*) (a(i,j),i = 1,nx),'    ', (b(i,j),i = 1,nx)
      end do
      write(*,*) '*----------------------------*'
     end do
```

c.. Excel data writing

```
      open(unit = 10, file = 'velocityx', status= 'unknown')
       do j = 1, ny
        write(10,200)(u(i, j), i = 1, nx)
       end do
      close(10)

      open(unit = 20, file = 'velocityy', status= 'unknown')
       do j = 1, ny
        write(20,200)(v(i, j), i = 1, nx)
       end do
      close(20)

      open(unit = 30, file = '      ', status= 'unknown')
       do j = 1, ny
        write(30,200)(            )
       end do
      close(30)

200   format(1x, 36e15.6)
      end
```