```
      subroutine GPOST2L(part,q,qm,nop,idimp,nxv,nyv)
c for 2d code, this subroutine calculates particle charge density
c using first-order linear interpolation, periodic boundaries
c scalar version using guard cells
c 17 flops/particle, 6 loads, 4 stores
c input: all, output: q
c charge density is approximated by values at the nearest grid points
c q(n,m)=qm*(1.-dx)*(1.-dy)
c q(n+1,m)=qm*dx*(1.-dy)
c q(n,m+1)=qm*(1.-dx)*dy
c q(n+1,m+1)=qm*dx*dy
c where n,m = leftmost grid points and dx = x-n, dy = y-m
c part(1,n) = position x of particle n
c part(2,n) = position y of particle n
c q(j,k) = charge density at grid point j,k
c qm = charge on particle, in units of e
c nop = number of particles
c idimp = size of phase space = 4
c nxv = first dimension of charge array, must be >= nx+1
c nxy = second dimension of charge array, must be >= ny+1
      dimension part(idimp,nop), q(nxv,nyv)
c find interpolation weights
      do 10 j = 1, nop
      nn = part(1,j)
      mm = part(2,j)
      dxp = qm*(part(1,j) - float(nn))
      dyp = part(2,j) - float(mm)
      nn = nn + 1
      mm = mm + 1
      amx = qm - dxp
      mp = mm + 1
      amy = 1. - dyp
      np = nn + 1
c deposit charge
      q(np,mp) = q(np,mp) + dxp*dyp
      q(nn,mp) = q(nn,mp) + amx*dyp
      q(np,mm) = q(np,mm) + dxp*amy
      q(nn,mm) = q(nn,mm) + amx*amy
   10 continue
      return
      end
```