```
program main

implicit none
double precision length,c,Phi
integer i_prtcl,j_prtcl,n_prtcl
integer i_iter,n_iter
double precision energy

double precision, dimension(:), allocatable :: k_vec_0
double precision, dimension(:), allocatable :: k_vec_new
double precision, dimension(:), allocatable :: k_vec_old

call init_var(length,n_prtcl,c,n_iter,Phi)

allocate( k_vec_0(n_prtcl) )
allocate( k_vec_old(n_prtcl) )
allocate( k_vec_new(n_prtcl) )

call init_k_vec(n_prtcl,length,k_vec_0,k_vec_old)

do i_iter = 1,n_iter

 call calc_k_vec_new(n_prtcl,k_vec_new,k_vec_old,k_vec_0,c,length,Phi)

enddo

energy = 0.d0
do i_prtcl = 1,n_prtcl
 energy = energy + k_vec_new(i_prtcl)*k_vec_new(i_prtcl)/2.d0
enddo

 write(6,*)"energy",energy

do i_prtcl = 1,n_prtcl-1
 write(10,*)i_prtcl,k_vec_new(i_prtcl)
 write(11,*)(k_vec_new(i_prtcl) + k_vec_new(i_prtcl+1))*0.5d0,&
     &1./(k_vec_new(i_prtcl+1) - k_vec_new(i_prtcl))
 write(12,*)(k_vec_0(i_prtcl) + k_vec_0(i_prtcl+1))*0.5d0,&
     &1./(k_vec_0(i_prtcl+1) - k_vec_0(i_prtcl))
enddo
write(10,*)n_prtcl,k_vec_new(n_prtcl)

deallocate( k_vec_0 )
deallocate( k_vec_old )
deallocate( k_vec_new )

end



subroutine init_var(length,n_prtcl,c,n_iter,Phi)

implicit none
double precision length,c,Phi
integer n_prtcl,n_iter

write(6,*)"length"
read(5,*)length
```

```
write(6,*)"n_prtcl"
read(5,*)n_prtcl
write(6,*)"c"
read(5,*)c
write(6,*)"Phi"
read(5,*)Phi
write(6,*)"n_iter"
read(5,*)n_iter

end


subroutine init_k_vec(n_prtcl,length,k_vec_0,k_vec_old)

implicit none

integer i_prtcl,n_prtcl
double precision length
double precision k_vec_0(n_prtcl)
double precision k_vec_old(n_prtcl)

do i_prtcl = 1,n_prtcl
 k_vec_0(i_prtcl) = 2*4*datan(1.d0) * (-dble(n_prtcl-1)/2 + i_prtcl - 1)/length
 k_vec_old(i_prtcl) = k_vec_0(i_prtcl)
enddo

end



subroutine calc_k_vec_new(n_prtcl,k_vec_new,k_vec_old,k_vec_0,c,length,Phi)

implicit none
integer i_prtcl,j_prtcl,n_prtcl
double precision c,length,Phi
double precision k_vec_0(n_prtcl)
double precision k_vec_old(n_prtcl)
double precision k_vec_new(n_prtcl)
double complex csum,cprod,eye
double precision delta_k

eye = (0.d0,1.d0)

do i_prtcl = 1,n_prtcl
 csum = (0.d0,0.d0)
 cprod = 1.d0
 do j_prtcl = 1,n_prtcl
  if (j_prtcl.ne.i_prtcl) then
  delta_k = k_vec_old(i_prtcl)-k_vec_old(j_prtcl)
  csum = csum - eye*log((eye*c + delta_k)/(eye*c-delta_k))
  cprod = cprod*(eye*c + delta_k)/(eye*c-delta_k)
  endif
 enddo
! k_vec_new(i_prtcl) = k_vec_0(i_prtcl) - eye*log(cprod)/length
 k_vec_new(i_prtcl) = k_vec_0(i_prtcl) + csum/length + Phi
enddo

do i_prtcl = 1,n_prtcl
```

```
 k_vec_old(i_prtcl) = .1*k_vec_new(i_prtcl) + .9*k_vec_old(i_prtcl)
enddo

end
```