Call_Structure_Program

skarner

August 22, 2017

Contents

1 Hard$_{\textbf{spheredynamics}}$(.True.,.True.,.True.)

is In `program.f90`

Line define:28

Line$_{\text{Call}}$: 16

1.1 Sys$_{\textbf{VariablesAlloc}}$(species)

is In File: variables$_{\text{modules}}$.f90

Module: System$_{\text{Variables}}$

Line Define: 27

Line Call: 55

1.2 D0M$_{\textbf{ini}}$ (species,sigma) !Diffusion coefficient of the system particle

is In File: variables$_{\text{modules}}$.f90

Module:System$_{\text{Variables}}$

Line Define: 39
Line Call: 57

1.3 rho_{ini}()

is In File: $\text{variables}_{\text{modules}}$.f90
Module: $\text{System}_{\text{Variables}}$
Line Define: 55
Line Call: 60

1.4 $\text{Static}_{\text{Variablesmemalloc}}$(kpoints,species)

is In File: $\text{variables}_{\text{modules}}$.f90
Module: $\text{Static}_{\text{Variables}}$
Line Define: 116
Line Call: 62

1.5 $\text{Calc}_{\text{statick}}$(dk,kpoints)

File Defined: $\text{variables}_{\text{modules}}$.f90 Module: $\text{Static}_{\text{Variables}}$ Line Define: 116
Line Call: 63

1.6 $\text{Calc}_{\text{Skhspymono}}$ #####Finally

is inFile: $\text{structure}_{\text{module}}$.f90
Module: $\text{Structure}_{\text{module}}$
Line Define:24
Line Call: 65
Code snippet

```
Subroutine Calc_Sk_hs_py_mono(vw_option)
    Implicit None
    Integer :: i1                                                     ! Define an enti
    Logical :: vw_option                                              ! Define an Boole
    Real * 8, Dimension(:), Allocatable :: k_vw                       ! Define a Vecto
    Real * 8 :: eta_vw,dum1,dum2,dum3,dum4,dumsin,dumcos              ! Define several
    !Start of parameter checks
    If (SDimen/= 3 .or. Species /=1 ) Then                            !Make desicion f
       print*, ('Error, cannot compute the system direct correlation function')
       Stop
    End If
```

```
!End of parameter checks

!### Here we must to watch the value for kpoints
!The value of kpoints is persistent, obviously xD : p
Allocate(k_vw(kpoints))                                           ! Allocate memory

!Start of VW Correction
If (vw_option .EQV. .TRUE.) Then                                  ! Make desicion 
   !Some data are calculated
   eta_vw=eta(1)*(1.d0-(eta(1)/16.d0))
   !##############################################################################
   !#########Analize space #######################
   !This instruction is interesting cause in this line whole array is traveled, th
   k_vw(:)=k(:)*((eta_vw/eta(1))**(1.d0/3.d0))
   !##############################################################################
Else
   eta_vw=eta(1)                                                  ! eta()
   k_vw(:)=k(:)
End If
!End of VW correction
dum1=-((1.0+2.0*eta_vw)**2)
dum2=6.0*eta_vw*((1.0+(eta_vw/2.0))**2)
dum3=-eta_vw*((1.0+(2.0*eta_vw))**2)/2.0
dum4=((1.0-eta_vw)**4)
dum1=dum1/dum4
dum2=dum2/dum4
dum3=dum3/dum4
Do i1=1, kpoints
   dumsin=sin(k_vw(i1))
   dumcos=cos(k_vw(i1))
   !##############################################################################
   !#########Analize space #######################
   !This sintax
   !                 ck(i1,1,1)= (dum1*(dumsin-k_vw(i1)*dumcos)/(k_vw(i1)**2.d0))+&
   !                 &%&(%&/"#$Other line.......
   ! Is used for concat severals lines of code as one

   !This is the same
   ck(i1,1,1)= (dum1*(dumsin-k_vw(i1)*dumcos)/(k_vw(i1)**2.d0))+&
&(dum2*(((2.d0*k_vw(i1))*dumsin)+((-(k_vw(i1)**2.d0)+2.d0)*dumcos)-2.d0)/&
  &(k_vw(i1)**3.d0))+(dum3*(((4.d0*(k_vw(i1)**3.d0)-24.0*k_vw(i1))*dumsin)&
  &+((-(k_vw(i1)**4.d0)+12.d0*(k_vw(i1)**2.d0)-24.d0)*dumcos)+24.d0)/(k_vw(i1)**5))
     !to This
     !ck(i1,1,1)= (dum1*(dumsin-k_vw(i1)*dumcos)/(k_vw(i1)**2.d0))+ &&(dum2*(((2.d0*
     Ck(i1,1,1)=  24.d0*eta_vw*ck(i1,1,1)/k_vw(i1)
     Sk(i1,1,1)=1.d0/(1.d0-ck(i1,1,1))
     Ski(i1,1,1)=1.d0-Ck(i1,1,1)
  End Do
End Subroutine Calc_Sk_hs_py_mono
```