

! DESARROLLO EXPERIMENTAL II (2018-1)
 ! CODIGO 7
 ! L.YEOMANS
 ! CONTINUACION: ADAPTANDO CODIGO 4 CON EL SIGUIENTE
 ! OBJETIVO PRINCIPAL:
 ! **INCORPORAR DECISIÓN PARA ALMACENAR CONFIGURACIONES**
 ! **DE EQUILIBRIO (ENSEMBLE) EN LOS ARREGLOS CX, CY (y CZ EN 3DIM)**

! MODELO DE POTENCIAL DE INTERACCION: DISCOS DUROS (HD)
 !
 ! WARNING! ESTE ES UN EJEMPLO, ADAPTE Y CONSTRUYA DE ACUERDO
 ! A SUS HERRAMIENTAS Y CONOCIMIENTOS DE PROGRAMACION EN FORTRAN
 !
 ! UNIDADES REDUCIDAS: SIGMA-BETA

```

PROGRAM CODIGO7
! ** DEFINICION DEL TIPO DE VARIABLES Y PARAMETROS **
  IMPLICIT REAL*8 (A-H,O-Z)
  COMMON /POSITION/ X(N),Y(N)
  COMMON /VALORES/ PHIT,DENS,BOXL
  INTEGER N
  PARAMETER ( N = 500 )
  PARAMETER ( PI = 3.1415927 )
  PARAMETER ( NN2 = 1000 )
  PARAMETER ( NENER = 10000 )
  DIMENSION CX(N,NN2),CY(N,NN2)

! ** LECTURA DE DATOS DE ENTRADA **
  WRITE(*,*)'DE EL NUMERO DE CICLOS '
  READ (*,*) NSTEP
  WRITE(*,*)'MONITOREO POR PANTALLA: CADA CUANTOS PASO?'
  READ (*,*) IPRINT
  WRITE(*,*)(' DAME FRECUENCIA CORREGIR DESPLAZAMIENTO MAX')
  READ (*,*) IRATIO
  WRITE(*,*)'CADA CUANTOS PASOS GUARDARA CONFIGURACIONES?'
  READ (*,*) ISAVE
  WRITE(*,*)'CONCENTRACION REDUCIDA'
  READ (*,*) DENS

! ** DESPLAZAMIENTO MAXIMO INICIAL
  DRMAX = 0.1

! ** CALCULOS PRELIMINARES **
  AA=1.0/2.0
  BOXL=((1.0*N)/DENS)**AA
  RCUT=BOXL/2.0D0
  KI2=0
  ACATMA=0.0
  ACM=0.0

! ** ESCRIBIR DATOS DE ENTRADA EN ALGUN FORMATO DE SALIDA **
  WRITE(*,*)(' NUMERO OF PARTICULAS ",I10 )' ) N
  WRITE(*,*)(' NUMERO TOTAL DE CONFIGS ",I10 )' ) NSTEP

```

```

WRITE(*,(' FRACCION EN AREA TOTAL ',F10.4 )) PHIT
WRITE(*,(' LONGITUD DE LA CELDA . ',F10.4 )) BOXL

! ** CONFIGURACION INICIAL ALEATORIA SIN TRASLAPES (CALCULO
! EN SUBROUTINA) **

CALL CONFIGINI(BOXL)

! ** CORRECCION DE LARGO ALCANCE **
! ** NO HAY CORRECCION EL POTENCIAL ES DE CORTO ALCANCE (HD) **
VLRC = 0.0

! ** CALCULO DE LA ENERGIA DE LA CONFIGURACION INICIAL**
CALL SUMUP(RCUT,V)

VI=V+VLRC

WRITE(*,(' ENERGIA DE LA CONFIGURACION INICIAL ',F10.4 )) VI

! ** DOS SEMILLAS PARA GENERAR LOS MOVIMIENTOS DE LAS PARTICULAS
WRITE(*,*)'DAME DOS SEMILLAS NUMEROS ENTEROS ARBITRARIAS'
READ(*,*)ISEED,JSEED

! ** ABRIR ARCHIVOS PARA INFORMACION DE SALIDA **
OPEN (12, FILE = 'cfin.dat', STATUS = 'unknown')
OPEN (13, FILE = 'terma.dat',STATUS = 'unknown')

! ** MOVER A LAS PARTICULAS **
! ** ITERACION SOBRE CICLOS Y MOLECULES **

DO 100 ISTEP = 1, NSTEP
DO 99 I = 1, N

RXIOLD=X(I)
RYIOLD=Y(I)

! **ENERGIA DE LA I-ESIMA PARTICULA EN LA CONFIGURACION VIEJA**
CALL ENERGY(RXIOLD,RYIOLD,I,RCUT,VOLD)

! *****
! ** MOVIMIENTO TENTATIVO **
! *****
RXINEW = RXIOLD + ( 2.0 * ZRAN ( ISEED ) - 1.0 ) * DRMAX
RYINEW = RYIOLD + ( 2.0 * ZRAN ( JSEED ) - 1.0 ) * DRMAX
! *****

! ** INCLUYENDO CONDICIONES PERIODICAS **
RXINEW = RXINEW -BOXL*ANINT(RXINEW/BOXL)
RYINEW = RYINEW -BOXL*ANINT(RYINEW/BOXL)

! **ENERGIA DE LA I-ESIMA PARTICULA EN LA CONFIGURACION NUEVA***
CALL ENERGY(RXINEW,RYINEW,I,RCUT,VNEW)

```

! ** CRITERIO DE ACEPTACION O RECHAZO DE CONFIGURACIONES (MONTECARLO) **

DELTV = VNEW - VOLD

DELTVB = DELTV

IF (DELTVB .LT. 75.0) THEN

IF (DELTV .LE. 0.0) THEN

V = V + DELTV

X(I) = RXINNEW

Y(I) = RYINNEW

ACATMA = ACATMA + 1.0

ELSEIF (EXP (- DELTVB) .GT. RAN (LDUMMY)) THEN

V = V + DELTV

X(I) = RXINNEW

Y(I) = RYINNEW

ACATMA = ACATMA + 1.0

ENDIF

ENDIF

ACM = ACM + 1.0

VN = (V + VLRC) / REAL (N)

! *****
i ** CONCLUYE ITERACION SOBRE PARTICULAS **
i *****

99 CONTINUE

WRITE(13,*)ISTEP, VN

!C ** MONITOREO DE ENERGIA DE LA CONFIGURACION POR PARTICULA EN PANTALLA **

IF(MOD(ISTEP,IPRINT) .EQ. 0)THEN

WRITE(*,*)ISTEP, VN

ENDIF

```

! ** VERIFICANDO SI AJUSTA EL DESPLAZAMIENTO DRMAX **
  IF ( MOD ( ISTEP, IRATIO ) .EQ. 0 ) THEN
    RATIO = ACATMA / REAL ( N * IRATIO )

    IF ( RATIO .GT. 0.5 ) THEN
      DRMAX = DRMAX * 1.05
    ELSE
      DRMAX = DRMAX * 0.95
    ENDIF

    ACATMA = 0.0
  ENDIF

! **VERIFICANDO SI DEBE ALMACENAR CONFIGURACIONES DE EQUILIBRIO **
  IF ( MOD ( ISTEP, ISAVE ) .EQ. 0. AND. ISTEP.GT.NENER ) THEN
    KI2=KI2+1

! ** CONSTRUIR LOS ARREGLOS PARA LAS CONFIGURACIONES SELECCIONADAS **
    DO 11 K=1,N,1
      CX(K,KI2)=X(K)
      CY(K,KI2)=Y(K)
11    CONTINUE
    ENDIF

100  CONTINUE

!  ** IMPRESIÓN DE CONFIGURACION FINAL **
    DO K=1,N,1
      WRITE(12,*)X(I),Y(I)
    ENDDO

    STOP
  END

!C  ** INCLUIR LAS SUBROUTINAS O FUNCIONES NECESARIAS **
!C  ** FIN DEL PROGRAMA **

```