```
        subroutine newvar
        Implicit real*8(a-h,o-z)
        double precision:: alpha,r0,pas
        integer:: cont,cont2,walkers, m, mi
        common/dades1/alpha,r0,pas
        common/dades2/m, mi, sec,encerts
        common/results/energia
        double precision, allocatable :: llistae(:),llistar1(:),llistarm1(:),llistarm2(:)
        double precision, allocatable :: llistaer1(:),llistaerm1(:)

        allocate ( llistae(m),llistar1(m),llistarm1(m),llistarm2(m) )
        allocate ( llistaer1(m),llistaerm1(m) )

        x=(r0/2.d0)*ran2(idum)
        p0= prob(x)

        ! termalitzation

    cont=0
    do i=1,mi

        pastemx=pas*ran2(idum)
!       xb=x+pastemx
        xb=pastemx
        p1=prob(xb)
        w=ran2(idum);

        IF(p1/p0>sec*w) THEN
            x=xb
            p0=p1
            cont=cont+1
        ENDIF

    enddo

!     write(*,*) "initial step ",pas
      pas=pas*cont/(encerts*mi);
!     write(*,*) " fitted step ",pas

!   i2=i

    cont=0
    cont2=0

!   do i=i2,m
    do i=1,m
        pastemx=pas*ran2(idum)
!       xb=x+pastemx
        xb=pastemx
        p1=prob(xb)
        w=ran2(idum);

        IF(p1/p0>sec*w) THEN
            cont=cont+1
            aux=ene(xb)
            llistae(cont)=aux
            llistar1(cont)=xb
            llistaer1(cont)=aux*xb

            llistarm1(cont)=1.d0/xb
            llistarm2(cont)=1.d0/xb**2
            llistaerm1(cont)=aux/xb
            x=xb
            p0=p1
        ELSE
            cont2=cont2+1
        ENDIF

    enddo
            alpha1=alpha
            energia=sum(llistae(1:cont))/cont
            r1=sum(llistar1(1:cont))/cont
```

```
            rm1=sum(llistarm1(1:cont))/cont
            rm2=sum(llistarm2(1:cont))/cont
            er1=sum(llistaer1(1:cont))/cont
            dere1=-er1+energia*r1
            dere2=alpha**2+rm2+dere1**2-2.d0*alpha*rm1+2.d0*alpha*dere1-2*dere1*rm1
            alpha=alpha1-dere1/dere2
            write(*,*) "cont  ",cont
!           write(*,*) "in newton ", alpha1,"  energia ",energia," dere1 ",dere1, "  dere2
",dere2,"  alpha ",alpha
            write(40,*) "in newton ", alpha1,"  energia ",energia," dere1 ",dere1, "  dere2
            ",dere2,"  alpha ",alpha
    deallocate (llistae, llistar1,llistarm1,llistarm2,llistaer1,llistaerm1)
return
end
```