

```

/*----- */
/* */
/* */
/*          3D VARIABLE PARTICLE CORE CENTRAL DENSITY          */
/* */
/*----- */
/* (c) Copyright Terence Witt 2007, All Rights Reserved */
/* NULLPHYSICS.COM */
/* HYPERDENSITY CALCULATIONS */

/* master includes */
#include "witt.h"
#include "resource.h"
#include "zproject.h"

BOOL CoreRecession(char *parfile, char *resultfile, BOOL x_symmetric, BOOL force_once, double force_d,
double force_rc, double force_rscan, double force_corrlimit, double force_centralssamplesize,
char *tottim);

/*----- */
/* */
/*          MULTINUCLEON CORE CENTRAL ENERGY DENSITY          */
/* */
/*----- */
/* calculate the average energy density in the central region of a hexagonally packed core lattice */
void CoreEnergyDensity(void)
{
    char    parfile[200];
    double  rcenter=0.0, rscan=0.0;
    long    maxcore=0, n;

    if(GetText(hWndMain, "Centroid Size"))
    {
        rcenter = atof(note_text);
        rscan = (2*rcenter) + 0.1;
    }

    if(GetText(hWndMain, "Maximum Core Number"))
    {
        maxcore = atol(note_text);
    }

    for(n=12; n<=maxcore; n++)
    {
        sprintf(parfile, "%snu%ld.dat", NUCPATH, n);
        if(FileExist(parfile))
        {
            CoreRecession(parfile, "", FALSE, TRUE, 0.0, 0.0, rscan, 0.0, rcenter, "");
        }
        if(KeyDown(VK_ESCAPE))
        {
            break;
        }
    }
}

```

```

/*-----*/
/*
/*      MULTINUCLEON CORE CENTRAL ENERGY DENSITY AVERAGE      */
/*
/*-----*/
/* average the energy densities of all of the core clusters */
void CoreDensityAverage(void)
{
    char    grafile[200], af[300];
    double  dense, avedense=0.0, totdense=0.0, volume;
    long    n;

    strcpy(grafile, CORE_DENS);

    for(n=41; n<=64; n++)
    {
        FileGetLine(grafile, n, af, sizeof(af));
        dense = atof(&af[80]);
        if(dense>0.0)
        {
            avedense += dense;
            totdense++;
        }
    }
    Gframe(MESSAGEFRAME);
    volume = 64.0*(4.0*PI/3.0)*Rp*Rp*Rp;
    sprintf(af, "Average Density: %6.4e", (avedense/totdense)*(mp)/volume);
    Gbold(1);
    Gjust(TA_CENTER);
    Gtextframesmall(MESSAGEFRAME, COLORBLUE, "", af, 21);
}

```