

부록 :

형광등에 의한 실내조도 예측 프로그램

```

#include <iostream.h>
#include <fstream.h>
#include <conio.h>
#include <math.h>
#include<time.h>
#include<stdlib.h>

#define dsin(x) (sin((x)*0.01745))
#define dcos(x) (cos((x)*0.01745))
#define dasin(x) (asin((x))/0.01745)
#define dacos(x) (acos((x))/0.01745)

class C_plain_rectangle;
class C_plain_cylinder;
class C_plain_triangle;
class C_lightinit;

static const double PI=3.1415926535;

double maxdata(double data1, double data2)
{
    if (data1 >= data2) return data1;
    else return data2;
}

double mindata(double data1, double data2)
{
    if (data1 <= data2) return data1;
    else return data2;
}

void F_vcalc(double& f_vix, double& f_viy, double& f_viz, double& f_c, double f_xipos[4], double f_yipos[4],
double f_zipos[4])
{
    f_vix=((f_yipos[1]-f_yipos[0])*(f_zipos[3]-f_zipos[0]))-((f_yipos[3]-f_yipos[0])*(f_zipos[1]-
f_zipos[0]));
    f_viy=((f_zipos[1]-f_zipos[0])*(f_xipos[3]-f_xipos[0]))-((f_zipos[3]-f_zipos[0])*(f_xipos[1]-
f_xipos[0]));
    f_viz=((f_xipos[1]-f_xipos[0])*(f_yipos[3]-f_yipos[0]))-((f_xipos[3]-f_xipos[0])*(f_yipos[1]-
f_yipos[0]));
    f_c=(f_vix*f_xipos[0]+f_viy*f_yipos[0]+f_viz*f_zipos[0])*(-1.);
}

class C_plain_rectangle
{
private:
    int i_xnum, i_ynum;
    double f_vx, f_vy, f_vz, f_c, f_xpos[4], f_ypos[4], f_zpos[4], f_trans, f_reflec, f_spec,
    f_value;
public:
    C_plain_rectangle(){}
    ~C_plain_rectangle(){}
    void F_initcalc()

```

```

{
    F_vcalc(f_vx, f_vy, f_vz, f_c, f_xpos, f_ypos, f_zpos);
}
void F_getdata(double f_xpos_r[4], double f_ypos_r[4], double f_zpos_r[4], int i_xnum_r, int
i_ynum_r, double f_trans_r, double f_reflec_r, double f_spec_r)
{
    int i_ir;
    for(i_ir=0;i_ir<=3;i_ir++)
    {
        f_xpos[i_ir]=f_xpos_r[i_ir];
        f_ypos[i_ir]=f_ypos_r[i_ir];
        f_zpos[i_ir]=f_zpos_r[i_ir];
    }
    i_xnum=i_xnum_r;
    i_ynum=i_ynum_r;
    f_trans=f_trans_r;
    f_reflec=f_reflec_r;
    f_spec=f_spec_r;
    f_value=0.;
}
void F_getslope_vector(double& f_d_p_vx, double& f_d_p_vy, double& f_d_p_vz)
{
    f_d_p_vx=f_vx;
    f_d_p_vy=f_vy;
    f_d_p_vz=f_vz;
}
double F_calc_photon(double f_d_p_vx, double f_d_p_vy, double f_d_p_vz, double f_d_p_px, double
f_d_p_py, double f_d_p_pz)
{
    double f_temp, f_xt, f_yt, f_zt, f_r, f_first_xvt, f_first_yvt, f_first_zvt, f_xvt, f_yvt,
f_zvt;
    int i_temp;
    f_temp=f_vx*f_d_p_vx+f_vy*f_d_p_vy+f_vz*f_d_p_vz;
    if(f_temp>=0.) return (-1.);
// if((f_vx*f_d_p_px+f_vy*f_d_p_py+f_vz*f_d_p_pz+f_c)==0.) return(-1.);
    f_r=(f_vx*f_d_p_px+f_vy*f_d_p_py+f_vz*f_d_p_pz+f_c)/(f_vx*f_vx+f_vy*f_vy+f_vz*f_vz)*(-1.);
    f_xt=f_d_p_px+f_vx*f_r;
    f_yt=f_d_p_py+f_vy*f_r;
    f_zt=f_d_p_pz+f_vz*f_r;
    f_temp=(f_xt-f_d_p_px)*f_d_p_vx+(f_yt-f_d_p_py)*f_d_p_vy+(f_zt-f_d_p_pz)*f_d_p_vz;
    if(f_temp<=0.) return (-1.);
    if(f_xt==f_d_p_px&&f_yt==f_d_p_py&&f_zt==f_d_p_pz) return (-1.);
    f_r=(f_vx*f_d_p_px+f_vy*f_d_p_py+f_vz*f_d_p_pz+f_c)/(f_vx*f_d_p_vx+f_vy*f_d_p_vy+f_vz*f_d_p_vz)
*(-1.);
    f_xt=f_d_p_px+f_d_p_vx*f_r;
    f_yt=f_d_p_py+f_d_p_vy*f_r;
    f_zt=f_d_p_pz+f_d_p_vz*f_r;
    for(i_temp=0;i_temp<=4;i_temp++)
    {
        f_r=(f_vx*f_xt+f_vy*f_yt+f_vz*f_zt+f_c)/(f_vx*f_d_p_vx+f_vy*f_d_p_vy+f_vz*f_d_p_vz)*(-1.);
        f_xt=f_xt+f_d_p_vx*f_r;
        f_yt=f_yt+f_d_p_vy*f_r;
        f_zt=f_zt+f_d_p_vz*f_r;
    }
    f_first_xvt=(f_ypos[0]-f_ypos[3])*(f_zt-f_zpos[3])-(f_zpos[0]-f_zpos[3])*(f_yt-f_ypos[3]);
    f_first_yvt=(f_zpos[0]-f_zpos[3])*(f_xt-f_xpos[3])-(f_xpos[0]-f_xpos[3])*(f_zt-f_zpos[3]);
}

```

```

f_first_zvt=(f_xpos[0]-f_xpos[3])*(f_yt-f_ypos[3])-(f_ypos[0]-f_ypos[3])*(f_xt-f_xpos[3]);
for(i_temp=0;i_temp<=2;i_temp++)
{
    f_xvt=(f_ypos[i_temp+1]-f_ypos[i_temp])*(f_zt-f_zpos[i_temp])-(f_zpos[i_temp+1]-
    f_zpos[i_temp])*(f_yt-f_ypos[i_temp]);
    f_yvt=(f_zpos[i_temp+1]-f_zpos[i_temp])*(f_xt-f_xpos[i_temp])-(f_xpos[i_temp+1]-
    f_xpos[i_temp])*(f_zt-f_zpos[i_temp]);
    f_zvt=(f_xpos[i_temp+1]-f_xpos[i_temp])*(f_yt-f_ypos[i_temp])-(f_ypos[i_temp+1]-
    f_ypos[i_temp])*(f_xt-f_xpos[i_temp]);
    f_temp=f_first_xvt*f_xvt+f_first_yvt*f_yvt+f_first_zvt*f_zvt;
    if(f_temp<=0.) return (-1.);
}
return(sqrt((f_d_p_px-f_xt)*(f_d_p_px-f_xt)+(f_d_p_py-f_yt)*(f_d_p_py-f_yt)+(f_d_p_pz-
f_zt)*(f_d_p_pz-f_zt)));
}
voidF_add_value(doublef_ph_weight_t,double
f_relative_lumin_t){f_value=f_value+f_ph_weight_t*f_relative_lumin_t;}
void F_gen_diffu_pho(double& f_d_p_vx, double& f_d_p_vy, double& f_d_p_vz, double& f_d_p_px,
double& f_d_p_py, double& f_d_p_pz)
{
    double f_theta_t, f_pusai_t, f_random, f_xx_t, f_xy_t, f_xz_t, f_yx_t, f_yy_t, f_yz_t;
    double f_xt, f_yt, f_zt, f_r, f_xpolar_axis[3], f_ypolar_axis[3], f_zpolar_axis[3];
    void randomize();
    f_random=((double)rand())/32767.;
    f_xx_t=(f_xpos[1]-f_xpos[0])*f_random;
    f_xy_t=(f_ypos[1]-f_ypos[0])*f_random;
    f_xz_t=(f_zpos[1]-f_zpos[0])*f_random;
    f_random=((double)rand())/32767.;
    f_yx_t=(f_xpos[3]-f_xpos[0])*f_random;
    f_yy_t=(f_ypos[3]-f_ypos[0])*f_random;
    f_yz_t=(f_zpos[3]-f_zpos[0])*f_random;
    f_d_p_px=f_xx_t+f_yx_t+f_xpos[0];
    f_d_p_py=f_xy_t+f_yy_t+f_ypos[0];
    f_d_p_pz=f_xz_t+f_yz_t+f_zpos[0];
    f_xpolar_axis[0]=f_xpos[0]-f_d_p_px;
    f_xpolar_axis[1]=f_ypos[0]-f_d_p_py;
    f_xpolar_axis[2]=f_zpos[0]-f_d_p_pz;
    f_r=sqrt(f_xpolar_axis[0]*f_xpolar_axis[0]+f_xpolar_axis[1]*f_xpolar_axis[1]+f_xpolar_axis[
    2]*f_xpolar_axis[2]);
    f_xpolar_axis[0]=f_xpolar_axis[0]/f_r;
    f_xpolar_axis[1]=f_xpolar_axis[1]/f_r;
    f_xpolar_axis[2]=f_xpolar_axis[2]/f_r;
    f_r=sqrt(f_vx*f_vx+f_vy*f_vy+f_vz*f_vz);
    f_zpolar_axis[0]=f_vx/f_r;
    f_zpolar_axis[1]=f_vy/f_r;
    f_zpolar_axis[2]=f_vz/f_r;
    f_ypolar_axis[0]=f_zpolar_axis[1]*f_xpolar_axis[2]-f_zpolar_axis[2]*f_xpolar_axis[1];
    f_ypolar_axis[1]=f_zpolar_axis[2]*f_xpolar_axis[0]-f_zpolar_axis[0]*f_xpolar_axis[2];
    f_ypolar_axis[2]=f_zpolar_axis[0]*f_xpolar_axis[1]-f_zpolar_axis[1]*f_xpolar_axis[0];
    f_xt=f_xpolar_axis[0]*f_d_p_vx+f_xpolar_axis[1]*f_d_p_vy+f_xpolar_axis[2]*f_d_p_vz;
    f_yt=f_ypolar_axis[0]*f_d_p_vx+f_ypolar_axis[1]*f_d_p_vy+f_ypolar_axis[2]*f_d_p_vz;
    f_zt=f_zpolar_axis[0]*f_d_p_vx+f_zpolar_axis[1]*f_d_p_vy+f_zpolar_axis[2]*f_d_p_vz;
    f_pusai_t=((double)rand()/32767.)*PI*2.;
    f_theta_t=asin(sqrt((double)rand()/32767.));
    f_xt=sin(f_theta_t)*cos(f_pusai_t);
    f_yt=sin(f_theta_t)*sin(f_pusai_t);
}

```

```

f_zt=cos(f_theta_t);
f_r=1./(f_xpolar_axis[0]*(f_ypolar_axis[1]*f_zpolar_axis[2]-
f_ypolar_axis[2]*f_zpolar_axis[1])-f_xpolar_axis[1]*(f_ypolar_axis[0]*f_zpolar_axis[2]-
f_zpolar_axis[0]*f_ypolar_axis[2])+f_xpolar_axis[2]*(f_ypolar_axis[0]*f_zpolar_axis[1]-
f_zpolar_axis[0]*f_ypolar_axis[1]));
f_d_p_vx=((f_ypolar_axis[1]*f_zpolar_axis[2]-f_ypolar_axis[2]*f_zpolar_axis[1])*f_xt-
(f_xpolar_axis[1]*f_zpolar_axis[2]-
f_xpolar_axis[2]*f_zpolar_axis[1])*f_yt+(f_xpolar_axis[1]*f_ypolar_axis[2]-
f_xpolar_axis[2]*f_ypolar_axis[1])*f_zt)*f_r;
f_d_p_vy=((f_ypolar_axis[2]*f_zpolar_axis[0]-
f_ypolar_axis[0]*f_zpolar_axis[2])*f_xt+(f_xpolar_axis[0]*f_zpolar_axis[2]-
f_xpolar_axis[2]*f_zpolar_axis[0])*f_yt-(f_xpolar_axis[0]*f_ypolar_axis[2]-
f_xpolar_axis[2]*f_ypolar_axis[0])*f_zt)*f_r;
f_d_p_vz=((f_ypolar_axis[0]*f_zpolar_axis[1]-f_ypolar_axis[1]*f_zpolar_axis[0])*f_xt-
(f_xpolar_axis[0]*f_zpolar_axis[1]-
f_xpolar_axis[1]*f_zpolar_axis[0])*f_yt+(f_xpolar_axis[0]*f_ypolar_axis[1]-
f_xpolar_axis[1]*f_ypolar_axis[0])*f_zt)*f_r;
}
void F_exact_grid(int& i_tnum_t, int& i_tnum_t)
{
    i_tnum_t=i_xnum;
    i_tnum_t=i_ynum;
}
void F_calc_xyspace(double f_d_p_vx, double f_d_p_vy, double f_d_p_vz, double f_d_p_px,
double f_d_p_py, double f_d_p_pz, int& i_tnum_t, int& i_tnum_t)
{
    double f_xt, f_yt, f_zt, f_r, f_hy, f_hx;
    f_r=(f_vx*f_d_p_px+f_vy*f_d_p_py+f_vz*f_d_p_pz+f_c)/(f_vx*f_d_p_vx+f_vy*f_d_p_vy+f_vz*f
_d_p_vz)*(-1.);
    f_xt=f_d_p_px+f_d_p_vx*f_r;
    f_yt=f_d_p_py+f_d_p_vy*f_r;
    f_zt=f_d_p_pz+f_d_p_vz*f_r;
    f_r=((f_xpos[1]-f_xpos[0])*(f_xt-f_xpos[0])+(f_ypos[1]-f_ypos[0])*(f_yt-
f_ypos[0])+(f_zpos[1]-f_zpos[0])*(f_zt-f_zpos[0]))/((f_xpos[1]-f_xpos[0])*(f_xpos[1]-
f_xpos[0])+(f_ypos[1]-f_ypos[0])*(f_ypos[1]-f_ypos[0])+(f_zpos[1]-
f_zpos[0])*(f_zpos[1]-f_zpos[0]));
    f_hy=sqrt((f_xpos[0]+(f_xpos[1]-f_xpos[0])*f_r-f_xt)*(f_xpos[0]+(f_xpos[1]-
f_xpos[0])*f_r-f_xt)+(f_ypos[0]+(f_ypos[1]-f_ypos[0])*f_r-f_yt)*(f_ypos[0]+(f_ypos[1]-
f_ypos[0])*f_r-f_yt)+(f_zpos[0]+(f_zpos[1]-f_zpos[0])*f_r-f_zt)*(f_zpos[0]+(f_zpos[1]-
f_zpos[0])*f_r-f_zt));
    f_r=((f_xpos[3]-f_xpos[0])*(f_xt-f_xpos[0])+(f_ypos[3]-f_ypos[0])*(f_yt-
f_ypos[0])+(f_zpos[3]-f_zpos[0])*(f_zt-f_zpos[0]))/((f_xpos[3]-f_xpos[0])*(f_xpos[3]-
f_xpos[0])+(f_ypos[3]-f_ypos[0])*(f_ypos[3]-f_ypos[0])+(f_zpos[3]-
f_zpos[0])*(f_zpos[3]-f_zpos[0]));
    f_hx=sqrt((f_xpos[0]+(f_xpos[3]-f_xpos[0])*f_r-f_xt)*(f_xpos[0]+(f_xpos[3]-
f_xpos[0])*f_r-f_xt)+(f_ypos[0]+(f_ypos[3]-f_ypos[0])*f_r-f_yt)*(f_ypos[0]+(f_ypos[3]-
f_ypos[0])*f_r-f_yt)+(f_zpos[0]+(f_zpos[3]-f_zpos[0])*f_r-f_zt)*(f_zpos[0]+(f_zpos[3]-
f_zpos[0])*f_r-f_zt));
    i_tnum_t=(int)((double)i_xnum*f_hx)/sqrt((f_xpos[1]-f_xpos[0])*(f_xpos[1]-
f_xpos[0])+(f_ypos[1]-f_ypos[0])*(f_ypos[1]-f_ypos[0])+(f_zpos[1]-
f_zpos[0])*(f_zpos[1]-f_zpos[0]));
    i_tnum_t=(int)((double)i_ynum*f_hy)/sqrt((f_xpos[3]-f_xpos[0])*(f_xpos[3]-
f_xpos[0])+(f_ypos[3]-f_ypos[0])*(f_ypos[3]-f_ypos[0])+(f_zpos[3]-
f_zpos[0])*(f_zpos[3]-f_zpos[0]));
    if(i_tnum_t==i_xnum) i_tnum_t=i_tnum_t-1;
    if(i_tnum_t==i_ynum) i_tnum_t=i_tnum_t-1;
}

```

```

}
void F_calc_origin_xyspace(double f_xt, double f_yt, double f_zt, int& i_tnum_t, int&
i_tynum_t)
{
    double f_r, f_hy, f_hx;
    f_r=((f_xpos[1]-f_xpos[0])*(f_xt-f_xpos[0])+(f_ypos[1]-f_ypos[0])*(f_yt-
f_ypos[0])+(f_zpos[1]-f_zpos[0])*(f_zt-f_zpos[0]))/((f_xpos[1]-f_xpos[0])*(f_xpos[1]-
f_xpos[0])+(f_ypos[1]-f_ypos[0])*(f_ypos[1]-f_ypos[0])+(f_zpos[1]-
f_zpos[0])*(f_zpos[1]-f_zpos[0]));
    f_hy=sqrt((f_xpos[0]+(f_xpos[1]-f_xpos[0])*f_r-f_xt)*(f_xpos[0]+(f_xpos[1]-
f_xpos[0])*f_r-f_xt)+(f_ypos[0]+(f_ypos[1]-f_ypos[0])*f_r-f_yt)*(f_ypos[0]+(f_ypos[1]-
f_ypos[0])*f_r-f_yt)+(f_zpos[0]+(f_zpos[1]-f_zpos[0])*f_r-f_zt)*(f_zpos[0]+(f_zpos[1]-
f_zpos[0])*f_r-f_zt));
    f_r=((f_xpos[3]-f_xpos[0])*(f_xt-f_xpos[0])+(f_ypos[3]-f_ypos[0])*(f_yt-
f_ypos[0])+(f_zpos[3]-f_zpos[0])*(f_zt-f_zpos[0]))/((f_xpos[3]-f_xpos[0])*(f_xpos[3]-
f_xpos[0])+(f_ypos[3]-f_ypos[0])*(f_ypos[3]-f_ypos[0])+(f_zpos[3]-
f_zpos[0])*(f_zpos[3]-f_zpos[0]));
    f_hx=sqrt((f_xpos[0]+(f_xpos[3]-f_xpos[0])*f_r-f_xt)*(f_xpos[0]+(f_xpos[3]-
f_xpos[0])*f_r-f_xt)+(f_ypos[0]+(f_ypos[3]-f_ypos[0])*f_r-f_yt)*(f_ypos[0]+(f_ypos[3]-
f_ypos[0])*f_r-f_yt)+(f_zpos[0]+(f_zpos[3]-f_zpos[0])*f_r-f_zt)*(f_zpos[0]+(f_zpos[3]-
f_zpos[0])*f_r-f_zt));
    i_tnum_t=(int)((double)i_xnum*f_hx)/sqrt((f_xpos[1]-f_xpos[0])*(f_xpos[1]-
f_xpos[0])+(f_ypos[1]-f_ypos[0])*(f_ypos[1]-f_ypos[0])+(f_zpos[1]-
f_zpos[0])*(f_zpos[1]-f_zpos[0]));
    i_tynum_t=(int)((double)i_ynum*f_hy)/sqrt((f_xpos[3]-f_xpos[0])*(f_xpos[3]-
f_xpos[0])+(f_ypos[3]-f_ypos[0])*(f_ypos[3]-f_ypos[0])+(f_zpos[3]-
f_zpos[0])*(f_zpos[3]-f_zpos[0]));
    if(i_tnum_t==i_xnum) i_tnum_t=i_tnum_t-1;
    if(i_tynum_t==i_ynum) i_tynum_t=i_tynum_t-1;
}
int F_trans_polaraxis(double& f_d_p_vx, double& f_d_p_vy, double& f_d_p_vz, double&
f_d_p_px, double& f_d_p_py, double& f_d_p_pz)
{
    double f_t, f_xt, f_yt, f_zt, f_r, f_xpolar_axis[3], f_ypolar_axis[3], f_zpolar_axis[3],
f_theta, f_pusai;
    int i_temp;
    void randomize();
    if(((double)rand()/32767.)>(f_reflec)) {return (-1);}
    for(i_temp=0;i_temp<5;i_temp++)
    {
        f_t=(f_vx*f_d_p_px+f_vy*f_d_p_py+f_vz*f_d_p_pz+f_c)/(f_vx*f_d_p_vx+f_vy*f_d_p_vy+f_
vz*f_d_p_vz)*(-1.);
        f_d_p_px=f_d_p_px+f_d_p_vx*f_t;
        f_d_p_py=f_d_p_py+f_d_p_vy*f_t;
        f_d_p_pz=f_d_p_pz+f_d_p_vz*f_t;
    }
    f_xpolar_axis[0]=f_xpos[0]-f_d_p_px;
    f_xpolar_axis[1]=f_ypos[0]-f_d_p_py;
    f_xpolar_axis[2]=f_zpos[0]-f_d_p_pz;
    f_r=sqrt(f_xpolar_axis[0]*f_xpolar_axis[0]+f_xpolar_axis[1]*f_xpolar_axis[1]+f_xpolar_a
xis[2]*f_xpolar_axis[2]);
    f_xpolar_axis[0]=f_xpolar_axis[0]/f_r;
    f_xpolar_axis[1]=f_xpolar_axis[1]/f_r;
    f_xpolar_axis[2]=f_xpolar_axis[2]/f_r;
    f_r=sqrt(f_vx*f_vx+f_vy*f_vy+f_vz*f_vz);
    f_zpolar_axis[0]=f_vx/f_r;

```

```

f_zpolar_axis[1]=f_vy/f_r;
f_zpolar_axis[2]=f_vz/f_r;
f_ypolar_axis[0]=f_zpolar_axis[1]*f_xpolar_axis[2]-f_zpolar_axis[2]*f_xpolar_axis[1];
f_ypolar_axis[1]=f_zpolar_axis[2]*f_xpolar_axis[0]-f_zpolar_axis[0]*f_xpolar_axis[2];
f_ypolar_axis[2]=f_zpolar_axis[0]*f_xpolar_axis[1]-f_zpolar_axis[1]*f_xpolar_axis[0];
f_xt=f_xpolar_axis[0]*f_d_p_vx+f_xpolar_axis[1]*f_d_p_vy+f_xpolar_axis[2]*f_d_p_vz;
f_yt=f_ypolar_axis[0]*f_d_p_vx+f_ypolar_axis[1]*f_d_p_vy+f_ypolar_axis[2]*f_d_p_vz;
f_zt=f_zpolar_axis[0]*f_d_p_vx+f_zpolar_axis[1]*f_d_p_vy+f_zpolar_axis[2]*f_d_p_vz;
if(f_zt==0.)f_theta=PI/2.;
else f_theta=atan(sqrt(f_xt*f_xt+f_yt*f_yt)/f_zt);
if(f_zt<0.0)f_theta=PI+f_theta;
f_pusai=atan(f_yt/f_xt);
if(f_xt<0.&&f_yt>0.)f_pusai=PI+f_pusai;
if(f_xt<0.&&f_yt<0.)f_pusai=f_pusai-PI;
f_r=sqrt(f_xt*f_xt+f_yt*f_yt+f_zt*f_zt);
if(((f_trans)==1.0)||(((double)rand()/32767.)<(1.018*f_trans*(cos(PI-f_theta)+sin(PI-
f_theta)*sin(PI-f_theta)*sin(PI-f_theta)*cos(PI-f_theta))))))
{
    if(((double)rand()/32767.)>f_spec)
    {
        f_pusai=((double)rand()/32767.)*PI*2.;
        f_theta=asin(sqrt((double)rand()/32767.))+PI/2.;
    }
    else
    {
    }
}
else
{
    if(((double)rand()/32767.)<f_spec)
    {
        f_theta=PI-f_theta;
    }
    else
    {
        f_pusai=((double)rand()/32767.)*PI*2.;
        f_theta=asin(sqrt((double)rand()/32767.));
    }
}
f_xt=(sin(f_theta)*cos(f_pusai))*f_r;
f_yt=(sin(f_theta)*sin(f_pusai))*f_r;
f_zt=(cos(f_theta))*f_r;
f_r=1./(f_xpolar_axis[0]*(f_ypolar_axis[1]*f_zpolar_axis[2]-
f_ypolar_axis[2]*f_zpolar_axis[1])-f_xpolar_axis[1]*(f_ypolar_axis[0]*f_zpolar_axis[2]-
f_zpolar_axis[0]*f_ypolar_axis[2])+f_xpolar_axis[2]*(f_ypolar_axis[0]*f_zpolar_axis[1]-
f_zpolar_axis[0]*f_ypolar_axis[1]));
f_d_p_vx=((f_ypolar_axis[1]*f_zpolar_axis[2]-f_ypolar_axis[2]*f_zpolar_axis[1])*f_xt-
(f_xpolar_axis[1]*f_zpolar_axis[2]-
f_xpolar_axis[2]*f_zpolar_axis[1])*f_yt+(f_xpolar_axis[1]*f_ypolar_axis[2]-
f_xpolar_axis[2]*f_ypolar_axis[1])*f_zt)*f_r;
f_d_p_vy=((f_ypolar_axis[2]*f_zpolar_axis[0]-
f_ypolar_axis[0]*f_zpolar_axis[2])*f_xt+(f_xpolar_axis[0]*f_zpolar_axis[2]-
f_xpolar_axis[2]*f_zpolar_axis[0])*f_yt-(f_xpolar_axis[0]*f_ypolar_axis[2]-
f_xpolar_axis[2]*f_ypolar_axis[0])*f_zt)*f_r;
f_d_p_vz=((f_ypolar_axis[0]*f_zpolar_axis[1]-f_ypolar_axis[1]*f_zpolar_axis[0])*f_xt-
(f_xpolar_axis[0]*f_zpolar_axis[1]-

```

```

        f_xpolar_axis[1]*f_zpolar_axis[0])*f_yt+(f_xpolar_axis[0]*f_ypolar_axis[1]-
        f_xpolar_axis[1]*f_ypolar_axis[0])*f_zt)*f_r;
        return (1);
    }
void F_xynum(int& i_xnum_t, int& i_ynum_t)
{
    i_xnum_t=i_xnum;
    i_ynum_t=i_ynum;
}
double F_get_area()
{
    double f_area_t;
    f_area_t=sqrt((f_xpos[0]-f_xpos[1])*(f_xpos[0]-f_xpos[1])+(f_ypos[0]-
    f_ypos[1])*(f_ypos[0]-f_ypos[1])+(f_zpos[0]-f_zpos[1])*(f_zpos[0]-
    f_zpos[1]))*sqrt((f_xpos[0]-f_xpos[3])*(f_xpos[0]-f_xpos[3])+(f_ypos[0]-
    f_ypos[3])*(f_ypos[0]-f_ypos[3])+(f_zpos[0]-f_zpos[3])*(f_zpos[0]-f_zpos[3]));
    return (f_area_t);
}
double F_put_value(){return f_value;}
void F_delete_value(){f_value=0.;}
double F_get_reflec(){return f_reflec;}
};

class C_plain_triangle
{
private:
    int i_xnum, i_ynum;
    double f_vx, f_vy, f_vz, f_c, f_xpos[4], f_ypos[4], f_zpos[4], f_trans, f_reflec, f_spec,
    f_value;
public:
    C_plain_triangle(){}
    ~C_plain_triangle(){}
    void F_initcalc()
    {
        F_vcalc(f_vx, f_vy, f_vz, f_c, f_xpos, f_ypos, f_zpos);
    }
    void F_getdata(double f_xpos_t[4], double f_ypos_t[4], double f_zpos_t[4], int i_xnum_t,
    int i_ynum_t, double f_trans_t, double f_reflec_t, double f_spec_t)
    {
        int i_it;
        for(i_it=0;i_it<=2;i_it++)
        {
            f_xpos[i_it]=f_xpos_t[i_it];
            f_ypos[i_it]=f_ypos_t[i_it];
            f_zpos[i_it]=f_zpos_t[i_it];
        }
        i_xnum=i_xnum_t;
        i_ynum=i_ynum_t;
        f_trans=f_trans_t;
        f_reflec=f_reflec_t;
        f_spec=f_spec_t;
        f_value=0.;
    }
    void F_getslope_vector(double& f_d_p_vx, double& f_d_p_vy, double& f_d_p_vz)
    {
        f_d_p_vx=f_vx;

```



```

        f_d_p_vy=f_vy;
        f_d_p_vz=f_vz;
    }
    double F_calc_photon(double f_d_p_vx, double f_d_p_vy, double f_d_p_vz, double f_d_p_px,
    double f_d_p_py, double f_d_p_pz)
    {
    double f_temp, f_xt, f_yt, f_zt, f_r, f_first_xvt, f_first_yvt, f_first_zvt, f_xvt, f_yvt,
    f_zvt;
    int i_temp;
    f_temp=f_vx*f_d_p_vx+f_vy*f_d_p_vy+f_vz*f_d_p_vz;
    if(f_temp>=0.) return (-1.);
    if((f_vx*f_d_p_px+f_vy*f_d_p_py+f_vz*f_d_p_pz+f_c)==0.) return(-1.);
    f_r=(f_vx*f_d_p_px+f_vy*f_d_p_py+f_vz*f_d_p_pz+f_c)/(f_vx*f_vx+f_vy*f_vy+f_vz*f_vz)*(-1.);
    f_xt=f_d_p_px+f_vx*f_r;
    f_yt=f_d_p_py+f_vy*f_r;
    f_zt=f_d_p_pz+f_vz*f_r;
    f_temp=(f_xt-f_d_p_px)*f_d_p_vx+(f_yt-f_d_p_py)*f_d_p_vy+(f_zt-f_d_p_pz)*f_d_p_vz;
    if(f_temp<=0.) return (-1.);
    if(f_xt==f_d_p_px&&f_yt==f_d_p_py&&f_zt==f_d_p_pz) return (-1.);
    f_r=(f_vx*f_d_p_px+f_vy*f_d_p_py+f_vz*f_d_p_pz+f_c)/(f_vx*f_d_p_vx+f_vy*f_d_p_vy+f_vz*f_d_p_vz)*(-1.);
    f_xt=f_d_p_px+f_d_p_vx*f_r;
    f_yt=f_d_p_py+f_d_p_vy*f_r;
    f_zt=f_d_p_pz+f_d_p_vz*f_r;
    for(i_temp=0;i_temp<5;i_temp++)
    {
        f_r=(f_vx*f_xt+f_vy*f_yt+f_vz*f_zt+f_c)/(f_vx*f_d_p_vx+f_vy*f_d_p_vy+f_vz*f_d_p_vz)*(-1.);
        f_xt=f_xt+f_d_p_vx*f_r;
        f_yt=f_yt+f_d_p_vy*f_r;
        f_zt=f_zt+f_d_p_vz*f_r;
    }
    f_first_xvt=(f_ypos[0]-f_ypos[2])*(f_zt-f_zpos[2])-(f_zpos[0]-f_zpos[2])*(f_yt-f_ypos[2]);
    f_first_yvt=(f_zpos[0]-f_zpos[2])*(f_xt-f_xpos[2])-(f_xpos[0]-f_xpos[2])*(f_zt-f_zpos[2]);
    f_first_zvt=(f_xpos[0]-f_xpos[2])*(f_yt-f_ypos[2])-(f_ypos[0]-f_ypos[2])*(f_xt-f_xpos[2]);
    for(i_temp=0;i_temp<=1;i_temp++)
    {
        f_xvt=(f_ypos[i_temp+1]-f_ypos[i_temp])*(f_zt-f_zpos[i_temp])-(f_zpos[i_temp+1]-f_zpos[i_temp])*(f_yt-f_ypos[i_temp]);
        f_yvt=(f_zpos[i_temp+1]-f_zpos[i_temp])*(f_xt-f_xpos[i_temp])-(f_xpos[i_temp+1]-f_xpos[i_temp])*(f_zt-f_zpos[i_temp]);
        f_zvt=(f_xpos[i_temp+1]-f_xpos[i_temp])*(f_yt-f_ypos[i_temp])-(f_ypos[i_temp+1]-f_ypos[i_temp])*(f_xt-f_xpos[i_temp]);
        f_temp=f_first_xvt*f_xvt+f_first_yvt*f_yvt+f_first_zvt*f_zvt;
        if(f_temp<=0.) return (-1.);
        return (sqrt((f_d_p_px-f_xt)*(f_d_p_px-f_xt)+(f_d_p_py-f_yt)*(f_d_p_py-f_yt)+(f_d_p_pz-f_zt)*(f_d_p_pz-f_zt)));
    }
    void F_add_value(double f_pho_weight_t,double
    f_relative_lumin_t){f_value=f_value+f_pho_weight_t*f_relative_lumin_t;}
    void F_gen_diffu_pho(double& f_d_p_vx, double& f_d_p_vy, double& f_d_p_vz, double& f_d_p_px,
    double& f_d_p_py, double& f_d_p_pz)
    {
        double f_theta_t, f_pusai_t, f_random, f_xx_t, f_xy_t, f_xz_t, f_yx_t, f_yy_t, f_yz_t,
        f_slope_t;
        double f_xt, f_yt, f_zt, f_r, f_xpolar_axis[3], f_ypolar_axis[3], f_zpolar_axis[3];

```

```

void randomize();
f_slope_t=sqrt((f_xpos[2]-f_xpos[0])*(f_xpos[2]-f_xpos[0])+(f_ypos[2]-
f_ypos[0])*(f_ypos[2]-f_ypos[0])+(f_zpos[2]-f_zpos[0])*(f_zpos[2]-
f_zpos[0]))/sqrt((f_xpos[1]-f_xpos[0])*(f_xpos[1]-f_xpos[0])+(f_ypos[1]-
f_ypos[0])*(f_ypos[1]-f_ypos[0])+(f_zpos[1]-f_zpos[0])*(f_zpos[1]-f_zpos[0]));
f_random=((double)rand())/32767.;
f_xx_t=(f_xpos[0]-f_xpos[1])*f_random;
f_xy_t=(f_ypos[0]-f_ypos[1])*f_random;
f_xz_t=(f_zpos[0]-f_zpos[1])*f_random;
do
{
    f_random=((double)rand())/32767.;
    f_yx_t=(f_xpos[2]-f_xpos[0])*f_random;
    f_yy_t=(f_ypos[2]-f_ypos[0])*f_random;
    f_yz_t=(f_zpos[2]-f_zpos[0])*f_random;
}while((sqrt(f_yx_t*f_yx_t+f_yy_t*f_yy_t+f_yz_t*f_yz_t)/sqrt((f_xpos[2]-
f_xpos[0])*(f_xpos[2]-f_xpos[0])+(f_ypos[2]-f_ypos[0])*(f_ypos[2]-
f_ypos[0])+(f_zpos[2]-f_zpos[0])*(f_zpos[2]-f_zpos[0]))<(((double)rand())/32767.));
f_d_p_px=f_xx_t+f_yx_t+f_xpos[1];
f_d_p_py=f_xy_t+f_yy_t+f_ypos[1];
f_d_p_pz=f_xz_t+f_yz_t+f_zpos[1];
f_xpolar_axis[0]=f_xpos[0]-f_d_p_px;
f_xpolar_axis[1]=f_ypos[0]-f_d_p_py;
f_xpolar_axis[2]=f_zpos[0]-f_d_p_pz;
f_r=sqrt(f_xpolar_axis[0]*f_xpolar_axis[0]+f_xpolar_axis[1]*f_xpolar_axis[1]+f_xpolar_a
xis[2]*f_xpolar_axis[2]);
f_xpolar_axis[0]=f_xpolar_axis[0]/f_r;
f_xpolar_axis[1]=f_xpolar_axis[1]/f_r;
f_xpolar_axis[2]=f_xpolar_axis[2]/f_r;
f_r=sqrt(f_vx*f_vx+f_vy*f_vy+f_vz*f_vz);
f_zpolar_axis[0]=f_vx/f_r;
f_zpolar_axis[1]=f_vy/f_r;
f_zpolar_axis[2]=f_vz/f_r;
f_ypolar_axis[0]=f_zpolar_axis[1]*f_xpolar_axis[2]-f_zpolar_axis[2]*f_xpolar_axis[1];
f_ypolar_axis[1]=f_zpolar_axis[2]*f_xpolar_axis[0]-f_zpolar_axis[0]*f_xpolar_axis[2];
f_ypolar_axis[2]=f_zpolar_axis[0]*f_xpolar_axis[1]-f_zpolar_axis[1]*f_xpolar_axis[0];
f_xt=f_xpolar_axis[0]*f_d_p_vx+f_xpolar_axis[1]*f_d_p_vy+f_xpolar_axis[2]*f_d_p_vz;
f_yt=f_ypolar_axis[0]*f_d_p_vx+f_ypolar_axis[1]*f_d_p_vy+f_ypolar_axis[2]*f_d_p_vz;
f_zt=f_zpolar_axis[0]*f_d_p_vx+f_zpolar_axis[1]*f_d_p_vy+f_zpolar_axis[2]*f_d_p_vz;
f_pusai_t=((double)rand()/32767.)*PI*2.;
f_theta_t=asin(sqrt((double)rand()/32767.));
f_xt=sin(f_theta_t)*cos(f_pusai_t);
f_yt=sin(f_theta_t)*sin(f_pusai_t);
f_zt=cos(f_theta_t);
f_r=1./(f_xpolar_axis[0]*f_ypolar_axis[1]*f_zpolar_axis[2]-
f_ypolar_axis[2]*f_zpolar_axis[1]-f_xpolar_axis[1]*f_ypolar_axis[0]*f_zpolar_axis[2]-
f_zpolar_axis[0]*f_ypolar_axis[2])+f_xpolar_axis[2]*(f_ypolar_axis[0]*f_zpolar_axis[1]-
f_zpolar_axis[0]*f_ypolar_axis[1]));
f_d_p_vx=((f_ypolar_axis[1]*f_zpolar_axis[2]-f_ypolar_axis[2]*f_zpolar_axis[1])*f_xt-
(f_xpolar_axis[1]*f_zpolar_axis[2]-
f_xpolar_axis[2]*f_zpolar_axis[1])*f_yt+(f_xpolar_axis[1]*f_ypolar_axis[2]-
f_xpolar_axis[2]*f_ypolar_axis[1])*f_zt)*f_r;
f_d_p_vy=((f_ypolar_axis[2]*f_zpolar_axis[0]-
f_ypolar_axis[0]*f_zpolar_axis[2])*f_xt+(f_xpolar_axis[0]*f_zpolar_axis[2]-
f_xpolar_axis[2]*f_zpolar_axis[0])*f_yt-(f_xpolar_axis[0]*f_ypolar_axis[2]-
f_xpolar_axis[2]*f_ypolar_axis[0])*f_zt)*f_r;

```

```

    f_d_p_vz=((f_ypolar_axis[0]*f_zpolar_axis[1]-f_ypolar_axis[1]*f_zpolar_axis[0])*f_xt-
    (f_xpolar_axis[0]*f_zpolar_axis[1]-
    f_xpolar_axis[1]*f_zpolar_axis[0])*f_yt+(f_xpolar_axis[0]*f_ypolar_axis[1]-
    f_xpolar_axis[1]*f_ypolar_axis[0])*f_zt)*f_r;
}
void F_exact_grid(int& i_tnum_t, int& i_tnum_t)
{
    i_tnum_t=i_xnum;
    i_tnum_t=i_ynum;
}
void F_calc_xyspace(double f_d_p_vx, double f_d_p_vy, double f_d_p_vz, double f_d_p_px,
double f_d_p_py, double f_d_p_pz, int& i_tnum_t, int& i_tnum_t)
{
    double f_xt, f_yt, f_zt, f_r, f_hx, f_hy;
    f_r=(f_vx*f_d_p_px+f_vy*f_d_p_py+f_vz*f_d_p_pz+f_c)/(f_vx*f_d_p_vx+f_vy*f_d_p_vy+f_vz*f
    _d_p_vz)*(-1.);
    f_xt=f_d_p_px+f_d_p_vx*f_r;
    f_yt=f_d_p_py+f_d_p_vy*f_r;
    f_zt=f_d_p_pz+f_d_p_vz*f_r;
    f_r=((f_xpos[1]-f_xpos[0])*f_xt-f_xpos[0])+(f_ypos[1]-f_ypos[0])*f_yt-
    f_ypos[0])+(f_zpos[1]-f_zpos[0])*f_zt-f_zpos[0])/((f_xpos[1]-f_xpos[0])*(f_xpos[1]-
    f_xpos[0])+(f_ypos[1]-f_ypos[0])*(f_ypos[1]-f_ypos[0])+(f_zpos[1]-
    f_zpos[0])*(f_zpos[1]-f_zpos[0]));
    f_hy=sqrt((f_xpos[0]+(f_xpos[1]-f_xpos[0])*f_r-f_xt)*(f_xpos[0]+(f_xpos[1]-
    f_xpos[0])*f_r-f_xt)+(f_ypos[0]+(f_ypos[1]-f_ypos[0])*f_r-f_yt)*(f_ypos[0]+(f_ypos[1]-
    f_ypos[0])*f_r-f_yt)+(f_zpos[0]+(f_zpos[1]-f_zpos[0])*f_r-f_zt)*(f_zpos[0]+(f_zpos[1]-
    f_zpos[0])*f_r-f_zt));
    f_r=((f_xpos[2]-f_xpos[0])*f_xt-f_xpos[0])+(f_ypos[2]-f_ypos[0])*f_yt-
    f_ypos[0])+(f_zpos[2]-f_zpos[0])*f_zt-f_zpos[0])/((f_xpos[2]-f_xpos[0])*(f_xpos[2]-
    f_xpos[0])+(f_ypos[2]-f_ypos[0])*(f_ypos[2]-f_ypos[0])+(f_zpos[2]-
    f_zpos[0])*(f_zpos[2]-f_zpos[0]));
    f_hx=sqrt((f_xpos[0]+(f_xpos[2]-f_xpos[0])*f_r-f_xt)*(f_xpos[0]+(f_xpos[2]-
    f_xpos[0])*f_r-f_xt)+(f_ypos[0]+(f_ypos[2]-f_ypos[0])*f_r-f_yt)*(f_ypos[0]+(f_ypos[2]-
    f_ypos[0])*f_r-f_yt)+(f_zpos[0]+(f_zpos[2]-f_zpos[0])*f_r-f_zt)*(f_zpos[0]+(f_zpos[2]-
    f_zpos[0])*f_r-f_zt));
    i_tnum_t=(int)(f_hx/(sqrt((f_xpos[1]-f_xpos[0])*(f_xpos[1]-f_xpos[0])+(f_ypos[1]-
    f_ypos[0])*(f_ypos[1]-f_ypos[0])+(f_zpos[1]-f_zpos[0])*(f_zpos[1]-f_zpos[0]))/(double)
    i_xnum));
    i_tnum_t=(int)(f_hy/(sqrt((f_xpos[2]-f_xpos[0])*(f_xpos[2]-f_xpos[0])+(f_ypos[2]-
    f_ypos[0])*(f_ypos[2]-f_ypos[0])+(f_zpos[2]-f_zpos[0])*(f_zpos[2]-f_zpos[0]))/(double)
    i_ynum));
    if(i_tnum_t==i_xnum)
        i_tnum_t=i_tnum_t-1;
    if(i_tnum_t==i_ynum)
        i_tnum_t=i_tnum_t-1;
}
int F_trans_polaraxis(double& f_d_p_vx, double& f_d_p_vy, double& f_d_p_vz, double&
f_d_p_px, double& f_d_p_py, double& f_d_p_pz)
{
    double f_t, f_xt, f_yt, f_zt, f_r, f_xpolar_axis[3], f_ypolar_axis[3], f_zpolar_axis[3],
    f_theta, f_pusai;
    int i_temp;
    void randomize();
    if(((double)rand()/32767.)>(f_reflec)) return (-1);
    for(i_temp=0;i_temp<5;i_temp++)
    {

```

```

f_t=(f_vx*f_d_p_px+f_vy*f_d_p_py+f_vz*f_d_p_pz+f_c)/(f_vx*f_d_p_vx+f_vy*f_d_p_vy+f_vz*f_d_p_vz)*(-1.);
f_d_p_px=f_d_p_px+f_d_p_vx*f_t;
f_d_p_py=f_d_p_py+f_d_p_vy*f_t;
f_d_p_pz=f_d_p_pz+f_d_p_vz*f_t;
}
f_xpolar_axis[0]=f_xpos[0]-f_d_p_px;
f_xpolar_axis[1]=f_ypos[0]-f_d_p_py;
f_xpolar_axis[2]=f_zpos[0]-f_d_p_pz;
f_r=sqrt(f_xpolar_axis[0]*f_xpolar_axis[0]+f_xpolar_axis[1]*f_xpolar_axis[1]+f_xpolar_axis[2]*f_xpolar_axis[2]);
f_xpolar_axis[0]=f_xpolar_axis[0]/f_r;
f_xpolar_axis[1]=f_xpolar_axis[1]/f_r;
f_xpolar_axis[2]=f_xpolar_axis[2]/f_r;
f_r=sqrt(f_vx*f_vx+f_vy*f_vy+f_vz*f_vz);
f_zpolar_axis[0]=f_vx/f_r;
f_zpolar_axis[1]=f_vy/f_r;
f_zpolar_axis[2]=f_vz/f_r;
f_ypolar_axis[0]=f_zpolar_axis[1]*f_xpolar_axis[2]-f_zpolar_axis[2]*f_xpolar_axis[1];
f_ypolar_axis[1]=f_zpolar_axis[2]*f_xpolar_axis[0]-f_zpolar_axis[0]*f_xpolar_axis[2];
f_ypolar_axis[2]=f_zpolar_axis[0]*f_xpolar_axis[1]-f_zpolar_axis[1]*f_xpolar_axis[0];
f_xt=f_xpolar_axis[0]*f_d_p_vx+f_xpolar_axis[1]*f_d_p_vy+f_xpolar_axis[2]*f_d_p_vz;
f_yt=f_ypolar_axis[0]*f_d_p_vx+f_ypolar_axis[1]*f_d_p_vy+f_ypolar_axis[2]*f_d_p_vz;
f_zt=f_zpolar_axis[0]*f_d_p_vx+f_zpolar_axis[1]*f_d_p_vy+f_zpolar_axis[2]*f_d_p_vz;
if(f_zt==0.)f_theta=PI/2.;
else f_theta=atan(sqrt(f_xt*f_xt+f_yt*f_yt)/f_zt);
if(f_zt<0.)f_theta=PI+f_theta;
f_pusai=atan(f_yt/f_xt);
if(f_xt<0.&&f_yt>0.)f_pusai=PI+f_pusai;
if(f_xt<0.&&f_yt<0.)f_pusai=f_pusai-PI;
f_r=sqrt(f_xt*f_xt+f_yt*f_yt+f_zt*f_zt);
if(((f_trans)==1.0)||(((double)rand()/32767.)<(1.018*f_trans*(cos(PI-f_theta)+sin(PI-f_theta)*sin(PI-f_theta)*sin(PI-f_theta)*cos(PI-f_theta))))))
{
    if(((double)rand()/32767.)>f_spec)
    {
        f_pusai=((double)rand()/32767.)*PI*2.;
        f_theta=asin(sqrt((double)rand()/32767.))+PI/2.;
    }
    else
    {
    }
}
else
{
    if(((double)rand()/32767.)<f_spec)
    {
        f_theta=PI-f_theta;
    }
    else
    {
        f_pusai=((double)rand()/32767.)*PI*2.;
        f_theta=asin(sqrt((double)rand()/32767.));
    }
}
}
f_xt=(sin(f_theta)*cos(f_pusai))*f_r;

```

```

        f_yt=(sin(f_theta)*sin(f_pusai))*f_r;
        f_zt=(cos(f_theta))*f_r;
        f_r=1./(f_xpolar_axis[0]*(f_ypolar_axis[1]*f_zpolar_axis[2]-
        f_ypolar_axis[2]*f_zpolar_axis[1])-f_xpolar_axis[1]*(f_ypolar_axis[0]*f_zpolar_axis[2]-
        f_zpolar_axis[0]*f_ypolar_axis[2])+f_xpolar_axis[2]*(f_ypolar_axis[0]*f_zpolar_axis[1]-
        f_zpolar_axis[0]*f_ypolar_axis[1]));
        f_d_p_vx=((f_ypolar_axis[1]*f_zpolar_axis[2]-f_ypolar_axis[2]*f_zpolar_axis[1])*f_xt-
        (f_xpolar_axis[1]*f_zpolar_axis[2]-
        f_xpolar_axis[2]*f_zpolar_axis[1])*f_yt+(f_xpolar_axis[1]*f_ypolar_axis[2]-
        f_xpolar_axis[2]*f_ypolar_axis[1])*f_zt)*f_r;
        f_d_p_vy=((f_ypolar_axis[2]*f_zpolar_axis[0]-
        f_ypolar_axis[0]*f_zpolar_axis[2])*f_xt+(f_xpolar_axis[0]*f_zpolar_axis[2]-
        f_xpolar_axis[2]*f_zpolar_axis[0])*f_yt-(f_xpolar_axis[0]*f_ypolar_axis[2]-
        f_xpolar_axis[2]*f_ypolar_axis[0])*f_zt)*f_r;
        f_d_p_vz=((f_ypolar_axis[0]*f_zpolar_axis[1]-f_ypolar_axis[1]*f_zpolar_axis[0])*f_xt-
        (f_xpolar_axis[0]*f_zpolar_axis[1]-
        f_xpolar_axis[1]*f_zpolar_axis[0])*f_yt+(f_xpolar_axis[0]*f_ypolar_axis[1]-
        f_xpolar_axis[1]*f_ypolar_axis[0])*f_zt)*f_r;
        return (1);
    }
    void F_xynum(int& i_xnum_t, int& i_ynum_t)
    {
        i_xnum_t=i_xnum;
        i_ynum_t=i_ynum;
    }
    double F_nogrid_get_area()
    {
        double f_area_t;
        f_area_t=sqrt((f_xpos[0]-f_xpos[1])*(f_xpos[0]-f_xpos[1])+(f_ypos[0]-
        f_ypos[1])*(f_ypos[0]-f_ypos[1])+(f_zpos[0]-f_zpos[1])*(f_zpos[0]-
        f_zpos[1]))*sqrt((f_xpos[0]-f_xpos[2])*(f_xpos[0]-f_xpos[2])+(f_ypos[0]-
        f_ypos[2])*(f_ypos[0]-f_ypos[2])+(f_zpos[0]-f_zpos[2])*(f_zpos[0]-f_zpos[2]))/2.;
        return (f_area_t);
    }
    double F_put_value(){return f_value;}
    void F_delete_value(){f_value=0.;}
};

class C_plain_cylinder
{
private:
    int i_xnum, i_ynum;
    double f_xpos[2], f_ypos[2], f_zpos[2], f_radius, f_trans, f_reflec, f_spec, f_value;
    double f_a, f_b, f_c, f_lx, f_mx, f_nx, f_lz, f_mz, f_nz, f_A, f_B, f_C, f_D, f_E, f_F, f_G,
    f_H, f_I, f_J;
public:
    C_plain_cylinder(){}
    ~C_plain_cylinder(){}
    double F_calc_photon(double f_p_vx, double f_p_vy, double f_p_vz, double f_p_px, double
    f_p_py, double f_p_pz)
    {
        double f_k, f_temp, f_r0, f_r1, f_r2, f_ra, f_rb, f_rc, f_xt, f_yt, f_zt, f_xtd, f_ytd,
        f_ztd;
        f_ra=f_A*f_p_vx*f_p_vx+f_B*f_p_vy*f_p_vy+f_C*f_p_vz*f_p_vz+f_D*f_p_vx*f_p_vy+f_E*f_p_vx
        *f_p_vz+f_F*f_p_vy*f_p_vz;
        f_rb=(2.0*f_A*f_p_px+f_D*f_p_py+f_E*f_p_pz+f_G)*f_p_vx+(2.0*f_B*f_p_py+f_D*f_p_px+f_F*f

```

```

    _p_pz+f_H)*f_p_vy+(2.0*f_C*f_p_pz+f_E*f_p_px+f_F*f_p_py+f_I)*f_p_vz;
    f_rc=f_A*f_p_px*f_p_px+f_B*f_p_py*f_p_py+f_C*f_p_pz*f_p_pz+f_D*f_p_px*f_p_py+f_E*f_p_px
    *f_p_pz+f_F*f_p_py*f_p_pz+f_G*f_p_px+f_H*f_p_py+f_I*f_p_pz+f_J;
    f_temp=f_rb*f_rb-4.0*f_ra*f_rc;
    if (f_temp<0.) return (-1.);
    f_r1=(-f_rb+sqrt(f_temp))/(2.0*f_ra);
    f_r2=(-f_rb-sqrt(f_temp))/(2.0*f_ra);
    if(f_rb>0.)f_r0=f_r1;
    else f_r0=f_r2;
    f_xt=f_p_px+f_p_vx*f_r0;
    f_yt=f_p_py+f_p_vy*f_r0;
    f_zt=f_p_pz+f_p_vz*f_r0;
    f_k=((f_xpos[1]-f_xpos[0])*(f_xt-f_xpos[0])+(f_ypos[1]-f_ypos[0])*(f_yt-
    f_ypos[0])+(f_zpos[1]-f_zpos[0])*(f_zt-f_zpos[0]))/((f_xpos[1]-f_xpos[0])*(f_xpos[1]-
    f_xpos[0])+(f_ypos[1]-f_ypos[0])*(f_ypos[1]-f_ypos[0])+(f_zpos[1]-
    f_zpos[0])*(f_zpos[1]-f_zpos[0]));
    f_xtd=(f_xpos[1]-f_xpos[0])*f_k+f_xpos[0];
    f_ytd=(f_ypos[1]-f_ypos[0])*f_k+f_ypos[0];
    f_ztd=(f_zpos[1]-f_zpos[0])*f_k+f_zpos[0];
    f_temp=(f_xtd-f_xpos[0])*(f_xtd-f_xpos[1])+(f_ytd-f_ypos[0])*(f_ytd-f_ypos[1])+(f_ztd-
    f_zpos[0])*(f_ztd-f_zpos[1]);
    if(f_temp>0.) return (-1.);
    return(sqrt((f_p_px-f_xt)*(f_p_px-f_xt)+(f_p_py-f_yt)*(f_p_py-f_yt)+(f_p_pz-
    f_zt)*(f_p_pz-f_zt)));
}
int F_trans_polaraxis(double& f_p_vx, double& f_p_vy, double& f_p_vz, double& f_p_px,
double& f_p_py, double& f_p_pz)
{
    double f_theta, f_pusai, f_vx, f_vy, f_vz;
    double f_temp, f_r0, f_r1, f_r2, f_ra, f_rb, f_rc, f_xt, f_yt, f_zt, f_r,
    f_xpolar_axis[3], f_ypolar_axis[3], f_zpolar_axis[3];
    void randomize();
    f_ra=f_A*f_p_vx*f_p_vx+f_B*f_p_vy*f_p_vy+f_C*f_p_vz*f_p_vz+f_D*f_p_vx*f_p_vy+f_E*f_p_vx
    *f_p_vz+f_F*f_p_vy*f_p_vz;
    f_rb=(2.0*f_A*f_p_px+f_D*f_p_py+f_E*f_p_pz+f_G)*f_p_vx+(2.0*f_B*f_p_py+f_D*f_p_px+f_F*f
    _p_pz+f_H)*f_p_vy+(2.0*f_C*f_p_pz+f_E*f_p_px+f_F*f_p_py+f_I)*f_p_vz;
    f_rc=f_A*f_p_px*f_p_px+f_B*f_p_py*f_p_py+f_C*f_p_pz*f_p_pz+f_D*f_p_px*f_p_py+f_E*f_p_px
    *f_p_pz+f_F*f_p_py*f_p_pz+f_G*f_p_px+f_H*f_p_py+f_I*f_p_pz+f_J;
    f_temp=f_rb*f_rb-4.0*f_ra*f_rc;
    f_r1=(-f_rb+sqrt(f_temp))/(2.0*f_ra);
    f_r2=(-f_rb-sqrt(f_temp))/(2.0*f_ra);
    if(f_rb>0.)f_r0=f_r1;
    else f_r0=f_r2;
    f_p_px=f_p_px+f_p_vx*f_r0;
    f_p_py=f_p_py+f_p_vy*f_r0;
    f_p_pz=f_p_pz+f_p_vz*f_r0;
    f_vx=2.0*f_A*f_p_px+f_D*f_p_py+f_E*f_p_pz+f_G;
    f_vy=2.0*f_B*f_p_py+f_D*f_p_px+f_F*f_p_pz+f_H;
    f_vz=2.0*f_C*f_p_pz+f_E*f_p_px+f_F*f_p_py+f_I;
    f_xpolar_axis[0]=f_a;
    f_xpolar_axis[1]=f_b;
    f_xpolar_axis[2]=f_c;
    f_r=sqrt(f_vx*f_vx+f_vy*f_vy+f_vz*f_vz);
    f_zpolar_axis[0]=f_vx/f_r;
    f_zpolar_axis[1]=f_vy/f_r;
    f_zpolar_axis[2]=f_vz/f_r;
}

```

```

f_ypolar_axis[0]=f_zpolar_axis[1]*f_xpolar_axis[2]-f_zpolar_axis[2]*f_xpolar_axis[1];
f_ypolar_axis[1]=f_zpolar_axis[2]*f_xpolar_axis[0]-f_zpolar_axis[0]*f_xpolar_axis[2];
f_ypolar_axis[2]=f_zpolar_axis[0]*f_xpolar_axis[1]-f_zpolar_axis[1]*f_xpolar_axis[0];
f_xt=f_xpolar_axis[0]*f_p_vx+f_xpolar_axis[1]*f_p_vy+f_xpolar_axis[2]*f_p_vz;
f_yt=f_ypolar_axis[0]*f_p_vx+f_ypolar_axis[1]*f_p_vy+f_ypolar_axis[2]*f_p_vz;
f_zt=f_zpolar_axis[0]*f_p_vx+f_zpolar_axis[1]*f_p_vy+f_zpolar_axis[2]*f_p_vz;
if(f_zt==0.)f_theta=PI/2.;
else f_theta=atan(sqrt(f_xt*f_xt+f_yt*f_yt)/f_zt);
if(f_zt<0.0)f_theta=PI+f_theta;
f_pusai=atan(f_yt/f_xt);
if(f_xt<0.0&&f_yt>0.)f_pusai=PI+f_pusai;
if(f_xt<0.0&&f_yt<0.)f_pusai=f_pusai-PI;
f_r=sqrt(f_xt*f_xt+f_yt*f_yt+f_zt*f_zt);
if(((f_trans)==1.0)||(((double)rand()/32767.)<(1.018*f_trans*(cos(PI-f_theta)+sin(PI-
f_theta)*sin(PI-f_theta)*sin(PI-f_theta)*cos(PI-f_theta))))))
{
    if(((double)rand()/32767.)>f_spec)
    {
        f_pusai=((double)rand()/32767.)*PI*2.;
        f_theta=asin(sqrt((double)rand()/32767.))+PI/2.;
    }
    else
    {
    }
}
else
{
    if(((double)rand()/32767.)<f_spec)
    {
        f_theta=PI-f_theta;
    }
    else
    {
        f_pusai=((double)rand()/32767.)*PI*2.;
        f_theta=asin(sqrt((double)rand()/32767.));
    }
}
f_xt=(sin(f_theta)*cos(f_pusai))*f_r;
f_yt=(sin(f_theta)*sin(f_pusai))*f_r;
f_zt=(cos(f_theta))*f_r;
f_r=1./(f_xpolar_axis[0]*(f_ypolar_axis[1]*f_zpolar_axis[2]-
f_ypolar_axis[2]*f_zpolar_axis[1])-f_xpolar_axis[1]*(f_ypolar_axis[0]*f_zpolar_axis[2]-
f_zpolar_axis[0]*f_ypolar_axis[2])+f_xpolar_axis[2]*(f_ypolar_axis[0]*f_zpolar_axis[1]-
f_zpolar_axis[0]*f_ypolar_axis[1]));
f_p_vx=((f_ypolar_axis[1]*f_zpolar_axis[2]-f_ypolar_axis[2]*f_zpolar_axis[1])*f_xt-
(f_xpolar_axis[1]*f_zpolar_axis[2]-
f_xpolar_axis[2]*f_zpolar_axis[1])*f_yt+(f_xpolar_axis[1]*f_ypolar_axis[2]-
f_xpolar_axis[2]*f_ypolar_axis[1])*f_zt)*f_r;
f_p_vy=((f_ypolar_axis[2]*f_zpolar_axis[0]-
f_ypolar_axis[0]*f_zpolar_axis[2])*f_xt+(f_xpolar_axis[0]*f_zpolar_axis[2]-
f_xpolar_axis[2]*f_zpolar_axis[0])*f_yt-(f_xpolar_axis[0]*f_ypolar_axis[2]-
f_xpolar_axis[2]*f_ypolar_axis[0])*f_zt)*f_r;
f_p_vz=((f_ypolar_axis[0]*f_zpolar_axis[1]-f_ypolar_axis[1]*f_zpolar_axis[0])*f_xt-
(f_xpolar_axis[0]*f_zpolar_axis[1]-
f_xpolar_axis[1]*f_zpolar_axis[0])*f_yt+(f_xpolar_axis[0]*f_ypolar_axis[1]-
f_xpolar_axis[1]*f_ypolar_axis[0])*f_zt)*f_r;

```

```

    return (1);
}
void F_getdata(double f_xpos_s[4], double f_ypos_s[4], double f_zpos_s[4], double&
f_radius_s, int& i_xnum_s, int& i_ynum_s, double& f_trans_s, double& f_reflec_s, double&
f_spec_s)
{
    int i_is;
    for(i_is=0;i_is<=1;i_is++)
    {
        f_xpos[i_is]=f_xpos_s[i_is];
        f_ypos[i_is]=f_ypos_s[i_is];
        f_zpos[i_is]=f_zpos_s[i_is];
    }
    f_radius=f_radius_s;
    i_xnum=i_xnum_s;
    i_ynum=i_ynum_s;
    f_trans=f_trans_s;
    f_reflec=f_reflec_s;
    f_spec=f_spec_s;
}
void F_initcalc()
{
    double f_mxa, f_mxb, f_mxc;
    f_a=f_xpos[1]-f_xpos[0];
    f_b=f_ypos[1]-f_ypos[0];
    f_c=f_zpos[1]-f_zpos[0];
    f_a=f_a/sqrt(f_a*f_a+f_b*f_b+f_c*f_c);
    f_b=f_b/sqrt(f_a*f_a+f_b*f_b+f_c*f_c);
    f_c=f_c/sqrt(f_a*f_a+f_b*f_b+f_c*f_c);
    f_lz=-f_a;
    f_mz=-f_b;
    f_nz=f_c;
    if(f_a==0.0)
    {
        f_lx=1.0;
        f_mx=0.0;
        f_nx=0.0;
    }else if (f_c==0.0)
    {
        f_mx=-f_b/((f_a*f_a+f_b*f_b)/f_a);
        f_lx=-f_b*f_mx/f_a;
        if(f_a>0.0)
        {
            f_nx=sqrt(1.0-f_lx*f_lx-f_mx*f_mx);
        }else
        {
            f_nx=-sqrt(1.0-f_lx*f_lx-f_mx*f_mx);
        }
    }else if (f_b==0.0)
    {
        f_mx=0.0;
        if(f_c>0.0)
        {
            f_nx=sqrt(1.0-(f_c*f_c+f_a*f_a)/(f_a*f_a));
            f_lx=f_c*f_nx/f_a;
        }else
    }
}

```



```

        {
            f_nx=-sqrt(1.0-(f_c*f_c+f_a*f_a)/(f_a*f_a));
            f_lx=f_c*f_nx/f_a;
        }
    }else
    {
        f_mxa=(f_a*f_a)/(f_b*f_b)+(1.0/(f_c*f_c))*((f_a*f_a+f_b*f_b)/f_b)*((f_a*f_a+f_b*f_b)/f_b)+1.0;
        f_mxb=(2.0*f_a/f_b)*(1.0+((f_a*f_a+f_b*f_b)/(f_c*f_c)));
        f_mxc=(f_a*f_a)/(f_c*f_c);
        if(f_a>=0.0)
        {
            if(f_nz>=0.0)
            {
                f_mx=(-f_mxb+sqrt(f_mxb*f_mxb-4.0*f_mxa*f_mxc))/(2.0*f_mxa);
            }else
            {
                f_mx=(-f_mxb-sqrt(f_mxb*f_mxb-4.0*f_mxa*f_mxc))/(2.0*f_mxa);
            }
            f_lx=f_a*f_mx/f_b+1.;
            f_nx=((f_a*f_a+f_b*f_b)/f_b)*f_mx+f_a/f_c;
        }else
        {
            if(f_nz>=0.0)
            {
                f_mx=(-f_mxb+sqrt(f_mxb*f_mxb-4.0*f_mxa*f_mxc))/(2.0*f_mxa);
            }else
            {
                f_mx=(-f_mxb-sqrt(f_mxb*f_mxb-4.0*f_mxa*f_mxc))/(2.0*f_mxa);
            }
            f_lx=f_a*f_mx/f_b-1.;
            f_nx=((f_a*f_a+f_b*f_b)/f_b)*f_mx-f_a/f_c;
        }
    }
    f_A=f_lx*f_lx+(f_mz*f_nx-f_nz*f_mx)*(f_mz*f_nx-f_nz*f_mx);
    f_B=f_mx*f_mx+(f_nz*f_lx-f_lz*f_nx)*(f_nz*f_lx-f_lz*f_nx);
    f_C=f_nx*f_nx+(f_lz*f_mx-f_mz*f_lx)*(f_lz*f_mx-f_mz*f_lx);
    f_D=2.*(f_lx*f_mx+(f_mz*f_nx-f_nz*f_mx)*(f_nz*f_lx-f_lz*f_nx));
    f_E=2.*(f_lx*f_nx+(f_mz*f_nx-f_nz*f_mx)*(f_lz*f_mx-f_mz*f_lx));
    f_F=2.*(f_mx*f_nx+(f_nz*f_lx-f_lz*f_nx)*(f_lz*f_mx-f_mz*f_lx));
    f_J=-f_radius*f_radius;
    f_G=(2.0*f_A*f_xpos[0] + f_D*f_ypos[0] + f_E*f_zpos[0]);
    f_H=(2.0*f_B*f_ypos[0] + f_D*f_xpos[0] + f_F*f_zpos[0]);
    f_I=(2.0*f_C*f_zpos[0] + f_E*f_xpos[0] + f_F*f_ypos[0]);
    f_J=f_A*f_xpos[0]*f_xpos[0]+f_B*f_ypos[0]*f_ypos[0]+f_C*f_zpos[0]*f_zpos[0]+f_D*f_xpos[0]*f_ypos[0]+f_E*f_xpos[0]*f_zpos[0]+f_F*f_ypos[0]*f_zpos[0]+f_J;
}
void F_add_value(){f_value++;}
void F_exact_grid(int& i_txnum_t, int& i_tynum_t)
{
    i_txnum_t=i_xnum;
    i_tynum_t=i_ynum;
}
};

class C_light_cylinder

```

```

{
private:
    int i_xnum, i_ynum;
    double f_flux, f_xpos[2], f_ypos[2], f_zpos[2], f_radius, f_trans, f_reflec, f_spec,
    f_value;
    double f_a, f_b, f_c, f_lx, f_mx, f_nx, f_lz, f_mz, f_nz, f_A, f_B, f_C, f_D, f_E, f_F, f_G,
    f_H, f_I, f_J;
    double f_x0, f_y0, f_z0;
public:
    C_light_cylinder() {}
    ~C_light_cylinder() {}
    double F_calc_photon(double f_p_vx, double f_p_vy, double f_p_vz, double f_p_px, double
    f_p_py, double f_p_pz)
    {
        double f_k, f_temp, f_r0, f_r1, f_r2, f_ra, f_rb, f_rc, f_xt, f_yt, f_zt, f_xtd, f_ytd,
        f_ztd;
        f_ra=f_A*f_p_vx*f_p_vx+f_B*f_p_vy*f_p_vy+f_C*f_p_vz*f_p_vz+f_D*f_p_vx*f_p_vy+f_E*f_p_vx
        *f_p_vz+f_F*f_p_vy*f_p_vz;
        f_rb=(2.0*f_A*f_p_px+f_D*f_p_py+f_E*f_p_pz+f_G)*f_p_vx+(2.0*f_B*f_p_py+f_D*f_p_px+f_F*f
        _p_pz+f_H)*f_p_vy+(2.0*f_C*f_p_pz+f_E*f_p_px+f_F*f_p_py+f_I)*f_p_vz;
        f_rc=f_A*f_p_px*f_p_px+f_B*f_p_py*f_p_py+f_C*f_p_pz*f_p_pz+f_D*f_p_px*f_p_py+f_E*f_p_px
        *f_p_pz+f_F*f_p_py*f_p_pz+f_G*f_p_px+f_H*f_p_py+f_I*f_p_pz+f_J;
        f_temp=f_rb*f_rb-4.0*f_ra*f_rc;
        if (f_temp<0.) return (-1.);
        f_r1=(-f_rb+sqrt(f_temp))/(2.0*f_ra);
        f_r2=(-f_rb-sqrt(f_temp))/(2.0*f_ra);
        if(f_rb>0.)f_r0=f_r1;
        else f_r0=f_r2;
        f_xt=f_p_px+f_p_vx*f_r0;
        f_yt=f_p_py+f_p_vy*f_r0;
        f_zt=f_p_pz+f_p_vz*f_r0;
        f_k=((f_xpos[1]-f_xpos[0])*(f_xt-f_xpos[0])+(f_ypos[1]-f_ypos[0])*(f_yt-
        f_ypos[0])+(f_zpos[1]-f_zpos[0])*(f_zt-f_zpos[0]))/((f_xpos[1]-f_xpos[0])*(f_xpos[1]-
        f_xpos[0])+(f_ypos[1]-f_ypos[0])*(f_ypos[1]-f_ypos[0])+(f_zpos[1]-
        f_zpos[0])*(f_zpos[1]-f_zpos[0]));
        f_xtd=(f_xpos[1]-f_xpos[0])*f_k+f_xpos[0];
        f_ytd=(f_ypos[1]-f_ypos[0])*f_k+f_ypos[0];
        f_ztd=(f_zpos[1]-f_zpos[0])*f_k+f_zpos[0];
        f_temp=(f_xtd-f_xpos[0])*(f_xtd-f_xpos[1])+(f_ytd-f_ypos[0])*(f_ytd-f_ypos[1])+(f_ztd-
        f_zpos[0])*(f_ztd-f_zpos[1]);
        if(f_temp>0.) return (-1.);
        return(sqrt((f_p_px-f_xt)*(f_p_px-f_xt)+(f_p_py-f_yt)*(f_p_py-f_yt)+(f_p_pz-
        f_zt)*(f_p_pz-f_zt)));
    }
    int F_trans_polaraxis(double& f_p_vx, double& f_p_vy, double& f_p_vz, double& f_p_px,
    double& f_p_py, double& f_p_pz)
    {
        double f_theta, f_pusai, f_vx, f_vy, f_vz;
        double f_temp, f_r0, f_r1, f_r2, f_ra, f_rb, f_rc, f_xt, f_yt, f_zt, f_r,
        f_xpolar_axis[3], f_ypolar_axis[3], f_zpolar_axis[3];
        void randomize();
        f_ra=f_A*f_p_vx*f_p_vx+f_B*f_p_vy*f_p_vy+f_C*f_p_vz*f_p_vz+f_D*f_p_vx*f_p_vy+f_E*f_p_vx
        *f_p_vz+f_F*f_p_vy*f_p_vz;
        f_rb=(2.0*f_A*f_p_px+f_D*f_p_py+f_E*f_p_pz+f_G)*f_p_vx+(2.0*f_B*f_p_py+f_D*f_p_px+f_F*f
        _p_pz+f_H)*f_p_vy+(2.0*f_C*f_p_pz+f_E*f_p_px+f_F*f_p_py+f_I)*f_p_vz;
        f_rc=f_A*f_p_px*f_p_px+f_B*f_p_py*f_p_py+f_C*f_p_pz*f_p_pz+f_D*f_p_px*f_p_py+f_E*f_p_px
    }
}

```

```

*f_p_pz+f_F*f_p_py*f_p_pz+f_G*f_p_px+f_H*f_p_py+f_I*f_p_pz+f_J;
f_temp=f_rb*f_rb-4.0*f_ra*f_rc;
f_r1=(-f_rb+sqrt(f_temp))/(2.0*f_ra);
f_r2=(-f_rb-sqrt(f_temp))/(2.0*f_ra);
if(f_rb>0.)f_r0=f_r1;
else f_r0=f_r2;
f_p_px=f_p_px+f_p_vx*f_r0;
f_p_py=f_p_py+f_p_vy*f_r0;
f_p_pz=f_p_pz+f_p_vz*f_r0;
f_vx=2.0*f_A*f_p_px+f_D*f_p_py+f_E*f_p_pz+f_G;
f_vy=2.0*f_B*f_p_py+f_D*f_p_px+f_F*f_p_pz+f_H;
f_vz=2.0*f_C*f_p_pz+f_E*f_p_px+f_F*f_p_py+f_I;
f_xpolar_axis[0]=f_a;
f_xpolar_axis[1]=f_b;
f_xpolar_axis[2]=f_c;
f_r=sqrt(f_vx*f_vx+f_vy*f_vy+f_vz*f_vz);
f_zpolar_axis[0]=f_vx/f_r;
f_zpolar_axis[1]=f_vy/f_r;
f_zpolar_axis[2]=f_vz/f_r;
f_ypolar_axis[0]=f_zpolar_axis[1]*f_xpolar_axis[2]-f_zpolar_axis[2]*f_xpolar_axis[1];
f_ypolar_axis[1]=f_zpolar_axis[2]*f_xpolar_axis[0]-f_zpolar_axis[0]*f_xpolar_axis[2];
f_ypolar_axis[2]=f_zpolar_axis[0]*f_xpolar_axis[1]-f_zpolar_axis[1]*f_xpolar_axis[0];
f_xt=f_xpolar_axis[0]*f_p_vx+f_xpolar_axis[1]*f_p_vy+f_xpolar_axis[2]*f_p_vz;
f_yt=f_ypolar_axis[0]*f_p_vx+f_ypolar_axis[1]*f_p_vy+f_ypolar_axis[2]*f_p_vz;
f_zt=f_zpolar_axis[0]*f_p_vx+f_zpolar_axis[1]*f_p_vy+f_zpolar_axis[2]*f_p_vz;
if(f_zt==0.)f_theta=PI/2.;
else f_theta=atan(sqrt(f_xt*f_xt+f_yt*f_yt)/f_zt);
if(f_zt<0.0)f_theta=PI+f_theta;
f_pusai=atan(f_yt/f_xt);
if(f_xt<0.&&f_yt>0.)f_pusai=PI+f_pusai;
if(f_xt<0.&&f_yt<0.)f_pusai=f_pusai-PI;
f_r=sqrt(f_xt*f_xt+f_yt*f_yt+f_zt*f_zt);
if(((f_trans)==1.0)||(((double)rand()/32767.)<(1.018*f_trans*(cos(PI-f_theta)+sin(PI-
f_theta)*sin(PI-f_theta)*sin(PI-f_theta)*cos(PI-f_theta))))))
{
    if(((double)rand()/32767.)>f_spec)
    {
        f_pusai=((double)rand()/32767.)*PI*2.;
        f_theta=asin(sqrt((double)rand()/32767.))+PI/2.;
    }
    else
    {
    }
}
else
{
    if(((double)rand()/32767.)<f_spec)
    {
        f_theta=PI-f_theta;
    }
    else
    {
        f_pusai=((double)rand()/32767.)*PI*2.;
        f_theta=asin(sqrt((double)rand()/32767.));
    }
}
}

```

```

f_xt=(sin(f_theta)*cos(f_pusai))*f_r;
f_yt=(sin(f_theta)*sin(f_pusai))*f_r;
f_zt=(cos(f_theta))*f_r;
f_r=1./(f_xpolar_axis[0]*(f_ypolar_axis[1]*f_zpolar_axis[2]-
f_ypolar_axis[2]*f_zpolar_axis[1])-f_xpolar_axis[1]*(f_ypolar_axis[0]*f_zpolar_axis[2]-
f_zpolar_axis[0]*f_ypolar_axis[2])+f_xpolar_axis[2]*(f_ypolar_axis[0]*f_zpolar_axis[1]-
f_zpolar_axis[0]*f_ypolar_axis[1]));
f_p_vx=((f_ypolar_axis[1]*f_zpolar_axis[2]-f_ypolar_axis[2]*f_zpolar_axis[1])*f_xt-
(f_xpolar_axis[1]*f_zpolar_axis[2]-
f_xpolar_axis[2]*f_zpolar_axis[1])*f_yt+(f_xpolar_axis[1]*f_ypolar_axis[2]-
f_xpolar_axis[2]*f_ypolar_axis[1])*f_zt)*f_r;
f_p_vy=((f_ypolar_axis[2]*f_zpolar_axis[0]-
f_ypolar_axis[0]*f_zpolar_axis[2])*f_xt+(f_xpolar_axis[0]*f_zpolar_axis[2]-
f_xpolar_axis[2]*f_zpolar_axis[0])*f_yt-(f_xpolar_axis[0]*f_ypolar_axis[2]-
f_xpolar_axis[2]*f_ypolar_axis[0])*f_zt)*f_r;
f_p_vz=((f_ypolar_axis[0]*f_zpolar_axis[1]-f_ypolar_axis[1]*f_zpolar_axis[0])*f_xt-
(f_xpolar_axis[0]*f_zpolar_axis[1]-
f_xpolar_axis[1]*f_zpolar_axis[0])*f_yt+(f_xpolar_axis[0]*f_ypolar_axis[1]-
f_xpolar_axis[1]*f_ypolar_axis[0])*f_zt)*f_r;
return (1);
}
void F_getdata(double f_flux_s, double f_xpos_s[4], double f_ypos_s[4], double f_zpos_s[4],
double& f_radius_s, int& i_xnum_s, int& i_ynum_s, double& f_trans_s, double& f_reflec_s,
double& f_spec_s)
{
int i_is;
f_flux=f_flux_s;
for(i_is=0;i_is<=4;i_is++)
{
f_xpos[i_is]=f_xpos_s[i_is];
f_ypos[i_is]=f_ypos_s[i_is];
f_zpos[i_is]=f_zpos_s[i_is];
}
f_radius=f_radius_s;
i_xnum=i_xnum_s;
i_ynum=i_ynum_s;
f_trans=f_trans_s;
f_reflec=f_reflec_s;
f_spec=f_spec_s;
}
double F_photon_weight(double f_pho_num_t)
{
double f_flux_t;
f_flux_t=f_flux/f_pho_num_t;
return(f_flux_t);
}
void F_initcalc()
{
double f_mxa, f_mxb, f_mxc;
f_a=f_xpos[1]-f_xpos[0];
f_b=f_ypos[1]-f_ypos[0];
f_c=f_zpos[1]-f_zpos[0];
f_a=f_a/sqrt(f_a*f_a+f_b*f_b+f_c*f_c);
f_b=f_b/sqrt(f_a*f_a+f_b*f_b+f_c*f_c);
f_c=f_c/sqrt(f_a*f_a+f_b*f_b+f_c*f_c);
f_lz=-f_a;
}

```

```

f_mz=-f_b;
f_nz=f_c;
if (f_a==0.0)
{
    f_lx=1.0;
    f_mx=0.0;
    f_nx=0.0;
}else if (f_c==0.0)
{
    f_mx=-f_b/((f_a*f_a+f_b*f_b)/f_a);
    f_lx=-f_b*f_mx/f_a;
    if (f_a>0.0)
    {
        f_nx=sqrt(1.0-f_lx*f_lx-f_mx*f_mx);
    }else
    {
        f_nx=-sqrt(1.0-f_lx*f_lx-f_mx*f_mx);
    }
}else if (f_b==0.0)
{
    f_mx=0.0;
    if (f_c>0.0)
    {
        f_nx=sqrt(1.0-(f_c*f_c+f_a*f_a)/(f_a*f_a));
        f_lx=f_c*f_nx/f_a;
    }else
    {
        f_nx=-sqrt(1.0-(f_c*f_c+f_a*f_a)/(f_a*f_a));
        f_lx=f_c*f_nx/f_a;
    }
}else
{
    f_mxa=(f_a*f_a)/(f_b*f_b)+(1.0/(f_c*f_c))*((f_a*f_a+f_b*f_b)/f_b)*((f_a*f_a+f_b*f_b)/f_b)+1.0;
    f_mxb=(2.0*f_a/f_b)*(1.0+((f_a*f_a+f_b*f_b)/(f_c*f_c)));
    f_mxc=(f_a*f_a)/(f_c*f_c);
    if (f_a>0.0)
    {
        if (f_nz>=0.0)
        {
            f_mx=(-f_mxb+sqrt(f_mxb*f_mxb-4.0*f_mxa*f_mxc))/(2.0*f_mxa);
        }else
        {
            f_mx=(-f_mxb-sqrt(f_mxb*f_mxb-4.0*f_mxa*f_mxc))/(2.0*f_mxa);
        }
        f_lx=f_a*f_mx/f_b+1.;
        f_nx=((f_a*f_a+f_b*f_b)/f_b)*f_mx+f_a/f_c;
    }else
    {
        if (f_nz>=0.0)
        {
            f_mx=(-f_mxb+sqrt(f_mxb*f_mxb-4.0*f_mxa*f_mxc))/(2.0*f_mxa);
        }else
        {
            f_mx=(-f_mxb-sqrt(f_mxb*f_mxb-4.0*f_mxa*f_mxc))/(2.0*f_mxa);
        }
    }
}

```

```

        f_lx=f_a*f_mx/f_b-1.;
        f_nx=((f_a*f_a+f_b*f_b)/f_b)*f_mx-f_a)/f_c;
    }
}
f_A=f_lx*f_lx+(f_mz*f_nx-f_nz*f_mx)*(f_mz*f_nx-f_nz*f_mx);
f_B=f_mx*f_mx+(f_nz*f_lx-f_lz*f_nx)*(f_nz*f_lx-f_lz*f_nx);
f_C=f_nx*f_nx+(f_lz*f_mx-f_mz*f_lx)*(f_lz*f_mx-f_mz*f_lx);
f_D=2.*(f_lx*f_mx+(f_mz*f_nx-f_nz*f_mx)*(f_nz*f_lx-f_lz*f_nx));
f_E=2.*(f_lx*f_nx+(f_mz*f_nx-f_nz*f_mx)*(f_lz*f_mx-f_mz*f_lx));
f_F=2.*(f_mx*f_nx+(f_nz*f_lx-f_lz*f_nx)*(f_lz*f_mx-f_mz*f_lx));
f_J=-f_radius*f_radius;
f_G=-(2.0*f_A*f_xpos[0] + f_D*f_ypos[0] + f_E*f_zpos[0]);
f_H=-(2.0*f_B*f_ypos[0] + f_D*f_xpos[0] + f_F*f_zpos[0]);
f_I=-(2.0*f_C*f_zpos[0] + f_E*f_xpos[0] + f_F*f_ypos[0]);
f_J=f_A*f_xpos[0]*f_xpos[0]+f_B*f_ypos[0]*f_ypos[0]+f_C*f_zpos[0]*f_zpos[0]+f_D*f_xpos[
0]*f_ypos[0]+f_E*f_xpos[0]*f_zpos[0]+f_F*f_ypos[0]*f_zpos[0]+f_J;
}
void F_gen_photon(double& f_p_vx, double& f_p_vy, double& f_p_vz, double& f_p_px, double&
f_p_py, double& f_p_pz)
{
    double f_theta_t, f_pusai_t, f_random, f_vx, f_vy, f_vz;
    double f_xt, f_yt, f_zt, f_r, f_xpolar_axis[3], f_ypolar_axis[3], f_zpolar_axis[3];
    void randomize();
    f_z0=sqrt((f_xpos[1]-f_xpos[0])*(f_xpos[1]-f_xpos[0])+(f_ypos[1]-f_ypos[0])*(f_ypos[1]-
f_ypos[0])+(f_zpos[1]-f_zpos[0])*(f_zpos[1]-f_zpos[0]))*((double)rand()/32767.);
    f_random=2.0*PI*((double)rand())/32767.;
    f_x0=cos(f_random)*f_radius;
    if(f_random>PI) f_y0=-sqrt(f_radius*f_radius-f_x0*f_x0);
    else f_y0=sqrt(f_radius*f_radius-f_x0*f_x0);
    f_p_px=(f_x0*f_lx + f_y0*f_mx + f_z0*f_nx)+f_xpos[0];
    f_p_py=(f_x0*(-f_b*f_nx-f_c*f_mx) + f_y0*(f_c*f_lx+f_a*f_nx) +f_z0*(-
f_a*f_mx+f_b*f_lx)) + f_ypos[0];
    f_p_pz=(-f_a*f_x0-f_b*f_y0+f_c*f_z0)+f_zpos[0];
    f_vx=2.0*f_A*f_p_px+f_D*f_p_py+f_E*f_p_pz+f_G;
    f_vy=2.0*f_B*f_p_py+f_D*f_p_px+f_F*f_p_pz+f_H;
    f_vz=2.0*f_C*f_p_pz+f_E*f_p_px+f_F*f_p_py+f_I;
    f_xpolar_axis[0]=f_a;
    f_xpolar_axis[1]=f_b;
    f_xpolar_axis[2]=f_c;
    f_r=sqrt(f_vx*f_vx+f_vy*f_vy+f_vz*f_vz);
    f_zpolar_axis[0]=f_vx/f_r;
    f_zpolar_axis[1]=f_vy/f_r;
    f_zpolar_axis[2]=f_vz/f_r;
    f_ypolar_axis[0]=f_zpolar_axis[1]*f_xpolar_axis[2]-f_zpolar_axis[2]*f_xpolar_axis[1];
    f_ypolar_axis[1]=f_zpolar_axis[2]*f_xpolar_axis[0]-f_zpolar_axis[0]*f_xpolar_axis[2];
    f_ypolar_axis[2]=f_zpolar_axis[0]*f_xpolar_axis[1]-f_zpolar_axis[1]*f_xpolar_axis[0];
    f_pusai_t=((double)rand()/32767.)*PI*2.;
    f_theta_t=asin(sqrt((double)rand()/32767.));
    f_xt=sin(f_theta_t)*cos(f_pusai_t);
    f_yt=sin(f_theta_t)*sin(f_pusai_t);
    f_zt=cos(f_theta_t);
    f_r=1./(f_xpolar_axis[0]*(f_ypolar_axis[1]*f_zpolar_axis[2]-
f_ypolar_axis[2]*f_zpolar_axis[1])-f_xpolar_axis[1]*(f_ypolar_axis[0]*f_zpolar_axis[2]-
f_zpolar_axis[0]*f_ypolar_axis[2])+f_xpolar_axis[2]*(f_ypolar_axis[0]*f_zpolar_axis[1]-
f_zpolar_axis[0]*f_ypolar_axis[1]));
    f_p_vx=((f_ypolar_axis[1]*f_zpolar_axis[2]-f_ypolar_axis[2]*f_zpolar_axis[1])*f_xt-

```

```

        (f_xpolar_axis[1]*f_zpolar_axis[2]-
        f_xpolar_axis[2]*f_zpolar_axis[1])*f_yt+(f_xpolar_axis[1]*f_ypolar_axis[2]-
        f_xpolar_axis[2]*f_ypolar_axis[1])*f_zt)*f_r;
        f_p_vy=((f_ypolar_axis[2]*f_zpolar_axis[0]-
        f_ypolar_axis[0]*f_zpolar_axis[2])*f_xt+(f_xpolar_axis[0]*f_zpolar_axis[2]-
        f_xpolar_axis[2]*f_zpolar_axis[0])*f_yt-(f_xpolar_axis[0]*f_ypolar_axis[2]-
        f_xpolar_axis[2]*f_ypolar_axis[0])*f_zt)*f_r;
        f_p_vz=((f_ypolar_axis[0]*f_zpolar_axis[1]-f_ypolar_axis[1]*f_zpolar_axis[0])*f_xt-
        (f_xpolar_axis[0]*f_zpolar_axis[1]-
        f_xpolar_axis[1]*f_zpolar_axis[0])*f_yt+(f_xpolar_axis[0]*f_ypolar_axis[1]-
        f_xpolar_axis[1]*f_ypolar_axis[0])*f_zt)*f_r;
    }
    void F_add_value(){f_value++;}
    void F_exact_grid(int& i_tnum_t, int& i_tynum_t)
    {
        i_tnum_t=i_xnum;
        i_tynum_t=i_ynum;
    }
};

void main()
{
    C_plain_rectangle* p_numofplain_r[200];
    double* p_value_r[200][10][10];
    C_plain_triangle* p_numofplain_t[50];
    double* p_value_t[50][10][10];
    C_plain_cylinder* p_numofplain_c[10];
    double* p_value_c[10][10][10];
    C_light_cylinder* p_numoflight_lc[50];
    double* p_value_lc[50][10][10];
    int i_num, i_i, i_ij, i_ij2, i_typeplain, i_light_num, i_r, i_t, i_c, i_lc, i_photon_continue,
    i_plain_num, i_plain_index, i_tnum, i_tynum;
    double f_tflux, f_txpos[4], f_typos[4], f_tzpos[4], f_tradius, f_ttrans, f_treflec, f_tspec;
    double f_phonum, f_i, f_phovx, f_phovy, f_phovz, f_phopx, f_phopy, f_phopz;
    double f_dist, f_dist_temp, f_trdius, f_phoweight, f_area;
    ifstream o_infilelight("l.txt");
    ifstream o_in("a.txt");
    ofstream o_out("a.out");
    o_in>>i_num; //평면의 갯수
    i_r=0; i_t=0; i_c=0; i_lc=0;
    for (i_i=0;i_i<i_num;i_i++)
    {
        o_in >> i_typeplain; //면의 형태
        switch (i_typeplain)
        {
            case 1:
            {
                p_numofplain_c[i_c]= new C_plain_cylinder;
                for (i_ij=0; i_ij<=1; i_ij++)
                {
                    o_in>>f_txpos[i_ij];
                    o_in>>f_typos[i_ij];
                    o_in>>f_tzpos[i_ij];
                }
            }
        }
    }
}

```

```

o_in>>f_tradius;
o_in>>i_txnum;
o_in>>i_tynum;
o_in>>f_ttrans;
o_in>>f_treflec;
o_in>>f_tspec;
p_numofplain_c[i_c]->F_getdata(f_txpos, f_typos, f_tzpos, f_trdius, i_txnum,
i_tynum, f_ttrans, f_treflec, f_tspec);
p_numofplain_c[i_c]->F_initcalc();
if(i_txnum>1||i_tynum>1)
{
for(i_ij=0;i_ij<i_txnum;i_ij++)
{
for(i_ij2=0;i_ij2<i_tynum;i_ij2++)
{
p_value_c[i_c][i_ij][i_ij2]= new double;
*p_value_c[i_c][i_ij][i_ij2]=0.;
}
}
}
i_c++;
break;
}
case 3:
{
p_numofplain_t[i_t]= new C_plain_triangle;
for (i_ij=0; i_ij<=2; i_ij++)
{
o_in>>f_txpos[i_ij];
o_in>>f_typos[i_ij];
o_in>>f_tzpos[i_ij];
}
o_in>>i_txnum;
o_in>>i_tynum;
o_in>>f_ttrans;
o_in>>f_treflec;
o_in>>f_tspec;
p_numofplain_t[i_t]->F_getdata(f_txpos, f_typos, f_tzpos, i_txnum, i_tynum,
f_ttrans, f_treflec, f_tspec);
p_numofplain_t[i_t]->F_initcalc();
if(i_txnum>1||i_tynum>1)
{
for(i_ij=0;i_ij<i_txnum;i_ij++)
{
for(i_ij2=0;i_ij2<i_tynum;i_ij2++)
{
p_value_t[i_t][i_ij][i_ij2]= new double;
*p_value_t[i_t][i_ij][i_ij2]=0.;
}
}
}
}
i_t++;
break;
}
case 4:
{

```



```

        if(i_r>=100)
        {
            i_r=i_r+1-1;
        }
        p_numofplain_r[i_r]= new C_plain_rectangle;
        for (i_ij=0; i_ij<=3; i_ij++)
        {
            o_in>>f_txpos[i_ij];
            o_in>>f_typos[i_ij];
            o_in>>f_tzpos[i_ij];
        }
        o_in>>i_txnum;
        o_in>>i_tynum;
        o_in>>f_ttrans;
        o_in>>f_treflec;
        o_in>>f_tspec;
        p_numofplain_r[i_r]->F_getdata(f_txpos, f_typos, f_tzpos, i_txnum, i_tynum,
        f_ttrans, f_treflec, f_tspec);
        p_numofplain_r[i_r]->F_initcalc();
        if(i_txnum>1||i_tynum>1)
        {
            for(i_ij=0;i_ij<i_txnum;i_ij++)
            {
                for(i_ij2=0;i_ij2<i_tynum;i_ij2++)
                {
                    p_value_r[i_r][i_ij][i_ij2]= new double;
                    *p_value_r[i_r][i_ij][i_ij2]=0.;
                }
            }
            i_r++;
            break;
        }
        default : break;
    }
}
o_infilelight>>i_light_num;
for (i_i=0;i_i<i_light_num;i_i++)
{
    o_infilelight>>i_typeplain;
    switch (i_typeplain)
    {
        case 1:
        {
            p_numoflight_lc[i_i]= new C_light_cylinder;
            o_infilelight>>f_tflux;
            for (i_ij=0; i_ij<=1; i_ij++)
            {
                o_infilelight>>f_txpos[i_ij];
                o_infilelight>>f_typos[i_ij];
                o_infilelight>>f_tzpos[i_ij];
            }
            o_infilelight>>f_tradius;
            o_infilelight>>i_txnum;
            o_infilelight>>i_tynum;
            o_infilelight>>f_ttrans;

```

```

o_infilelight>>f_treflec;
o_infilelight>>f_tspec;
p_numoflight_lc[i_i]->F_getdata(f_tflux, f_txpos, f_typos, f_tzpos, f_tradius,
i_txnum, i_tynum, f_ttrans, f_treflec, f_tspec);
p_numoflight_lc[i_i]->F_initcalc();
if(i_txnum>1||i_tynum>1)
{
    for(i_ij=0;i_ij<i_txnum;i_ij++)
    {
        for(i_ij2=0;i_ij2<i_tynum;i_ij2++)
        {
            p_value_lc[i_i][i_ij][i_ij2]= new double;
            *p_value_lc[i_i][i_ij][i_ij2]=0.;
        }
    }
    i_lc++;
    break;
}
default : break;
}
}
f_phonum=10000.0;
for (i_i=0;i_i<i_light_num;i_i++)
{
    f_phonum_weight=p_numoflight_lc[i_i]->F_photon_weight(f_phonum);
    for (f_i=0.0;f_i<f_phonum;f_i++)
    {
        p_numoflight_lc[i_i]->F_gen_photon(f_phonum_vx, f_phonum_vy, f_phonum_vz, f_phonum_px, f_phonum_py,
f_phonum_pz);
        i_photon_continue=1;
        while(i_photon_continue>0)
        {
            f_dist=-1.;
            i_plain_index=-1;
            i_plain_num=-1;
            f_dist_temp=-1.;
            for(i_ij=0;i_ij<i_c;i_ij++)
            {
                f_dist_temp=p_numofplain_c[i_ij]->F_calc_photon(f_phonum_vx, f_phonum_vy, f_phonum_vz,
f_phonum_px, f_phonum_py, f_phonum_pz);
                if(f_dist_temp>0.000000001)
                {
                    if ((f_dist<0.)||(f_dist>=f_dist_temp))
                    {
                        f_dist=f_dist_temp;
                        i_plain_index=1;
                        i_plain_num=i_ij;
                    }
                }
            }
        }
        for(i_ij=0;i_ij<i_lc;i_ij++)
        {
            f_dist_temp=p_numoflight_lc[i_ij]->F_calc_photon(f_phonum_vx, f_phonum_vy, f_phonum_vz,
f_phonum_px, f_phonum_py, f_phonum_pz);
            if(f_dist_temp>0.000000001)

```

```

    {
        if ((f_dist<0.)||(f_dist>=f_dist_temp))
        {
            f_dist=f_dist_temp;
            i_plain_index=2;
            i_plain_num=i_ij;
        }
    }
}
for(i_ij=0;i_ij<i_t;i_ij++)
{
    f_dist_temp=p_numofplain_t[i_ij]->F_calc_photon(f_phovx, f_phovy, f_phovz,
    f_phopx, f_phopy, f_phopz);
    if(f_dist_temp>0.)
    {
        if ((f_dist<0.)||(f_dist>=f_dist_temp))
        {
            f_dist=f_dist_temp;
            i_plain_index=3;
            i_plain_num=i_ij;
        }
    }
}
for(i_ij=0;i_ij<i_r;i_ij++)
{
    f_dist_temp=p_numofplain_r[i_ij]->F_calc_photon(f_phovx, f_phovy, f_phovz,
    f_phopx, f_phopy, f_phopz);
    if(f_dist_temp>0.)
    {
        if((f_dist<0.)||(f_dist>=f_dist_temp))
        {
            f_dist=f_dist_temp;
            i_plain_index=4;
            i_plain_num=i_ij;
        }
    }
}
if(f_dist>0.00000001)
{
    switch(i_plain_index)
    {
        case 4:
        {
            p_numofplain_r[i_plain_num]->F_exact_grid(i_tnxum, i_tynum);
            if(i_tnxum==1&&i_tynum==1)
            {
                p_numofplain_r[i_plain_num]->F_add_value(f_phoweight, 1.0);
                i_photon_continue=p_numofplain_r[i_plain_num]-
                >F_trans_polaraxis(f_phovx, f_phovy, f_phovz, f_phopx, f_phopy,
                f_phopz);
            }
            else
            {
                p_numofplain_r[i_plain_num]->F_calc_xyspace(f_phovx, f_phovy,
                f_phovz, f_phopx, f_phopy, f_phopz, i_tnxum, i_tynum);
                *p_value_r[i_plain_num][i_tnxum][i_tynum]=

```

```

        *p_value_r[i_plain_num][i_txnum][i_tynum]+f_pho_weight;
        i_photon_continue=p_numofplain_r[i_plain_num]-
        >F_trans_polaraxis(f_pho_vx, f_pho_vy, f_pho_vz, f_pho_px, f_pho_py,
        f_pho_pz);
    }
    break;
}
case 3:
{
    p_numofplain_t[i_plain_num]->F_exact_grid(i_txnum, i_tynum);
    if(i_txnum==1&&i_tynum==1)
    {
        p_numofplain_t[i_plain_num]->F_add_value(f_pho_weight, 1.0);
        i_photon_continue=p_numofplain_t[i_plain_num]-
        >F_trans_polaraxis(f_pho_vx, f_pho_vy, f_pho_vz, f_pho_px, f_pho_py,
        f_pho_pz);
    }
    else
    {
        p_numofplain_t[i_plain_num]->F_calc_xyspace(f_pho_vx,      f_pho_vy,
        f_pho_vz, f_pho_px, f_pho_py, f_pho_pz, i_txnum, i_tynum);
        *p_value_t[i_plain_num][i_txnum][i_tynum]=
        *p_value_t[i_plain_num][i_txnum][i_tynum]+f_pho_weight;
        i_photon_continue=p_numofplain_t[i_plain_num]-
        >F_trans_polaraxis(f_pho_vx, f_pho_vy, f_pho_vz, f_pho_px, f_pho_py,
        f_pho_pz);
    }
    break;
}
case 2:
{
    p_numoflight_lc[i_plain_num]->F_exact_grid(i_txnum, i_tynum);
    if(i_txnum==1&&i_tynum==1)
    {
        i_photon_continue=p_numoflight_lc[i_plain_num]-
        >F_trans_polaraxis(f_pho_vx, f_pho_vy, f_pho_vz, f_pho_px, f_pho_py,
        f_pho_pz);
    }
    else
    {
        *p_value_lc[i_plain_num][i_txnum][i_tynum]=
        *p_value_lc[i_plain_num][i_txnum][i_tynum]+f_pho_weight;
        i_photon_continue=p_numoflight_lc[i_plain_num]-
        >F_trans_polaraxis(f_pho_vx, f_pho_vy, f_pho_vz, f_pho_px, f_pho_py,
        f_pho_pz);
    }
    break;
}
case 1:
{
    p_numofplain_c[i_plain_num]->F_exact_grid(i_txnum, i_tynum);
    if(i_txnum==1&&i_tynum==1)
    {
        i_photon_continue=p_numofplain_c[i_plain_num]-
        >F_trans_polaraxis(f_pho_vx, f_pho_vy, f_pho_vz, f_pho_px, f_pho_py,
        f_pho_pz);
    }
}

```

```

    }
    else
    {
        *p_value_c[i_plain_num][i_txnum][i_tynum]=
        *p_value_c[i_plain_num][i_txnum][i_tynum]+f_pho_weight;
        i_photon_continue=p_numofplain_c[i_plain_num]-
        >F_trans_polaraxis(f_pho_vx, f_pho_vy, f_pho_vz, f_pho_px, f_pho_py,
        f_pho_pz);
    }
    break;
}
default: break;
}
}else
{
    i_photon_continue=-1;
}
}
}
for(i_i=0;i_i<i_r;i_i++)
{
    p_numofplain_r[i_i]->F_xynum(i_txnum, i_tynum);
    f_area=p_numofplain_r[i_i]->F_get_area();
    if(i_txnum==1&&i_tynum==1)
    {
        o_out<<"rectangle plain["<<i_i<<" = "<<(p_numofplain_r[i_i]->F_put_value()/f_area)<<"
        lx\n";
    }else
    {
        for(i_ij=0;i_ij<i_txnum;i_ij++)
        {
            for(i_ij2=0;i_ij2<i_tynum;i_ij2++)
            {
                o_out<<"rectangle      plain["<<i_i<<"]["<<i_ij<<"]["<<i_ij2<<"      =
                "<<(*p_value_r[i_i][i_ij][i_ij2])<<" lx\n";
            }
        }
    }
}
for(i_i=0;i_i<i_t;i_i++)
{
    p_numofplain_t[i_i]->F_xynum(i_txnum, i_tynum);
    f_area=p_numofplain_t[i_i]->F_nogrid_get_area();
}
o_in.close();
o_out.close();
o_infilelight.close();
}

```