

## 부록 2 :

몬테카를로 방법과 광선추적기법을 이용한  
아트리움 자연채광 성능 예측 프로그램

```

////////////////////////////////////
//몬테카를로 방법과 광선추적기법을 이용한 아트리움의 자연채광 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_triangle;
class C_lightinit;
static const double PI=3.1415926535;
static const double
f_coeff_a1[8]={1.3525, -1.2219, -1.1, -0.5484, -0.6, -1.0156, -1.0, -1.05},
f_coeff_a2[8]={-0.2576, -0.773, -0.2515, -0.6654, -0.3566, -0.367, 0.0211, 0.0289},
f_coeff_a3[8]={-0.2690, 1.4148, 0.8952, -0.2672, -2.5, 1.0078, 0.5025, 0.4260},
f_coeff_a4[8]={-1.4366, 1.1016, 0.0156, 0.7117, 2.3250, 1.4051, -0.5119, 0.3590},
f_coeff_b1[8]={-0.767, -0.2054, 0.2782, 0.7234, 0.2937, 0.2875, -0.3, -0.325},
f_coeff_b2[8]={0.0007, 0.0367, -0.1812, -0.6219, 0.0496, -0.5328, 0.1922, 0.1156},
f_coeff_b3[8]={1.2734, -3.9128, -4.5, -5.6812, -5.6812, -3.85, 0.7023, 0.7781},
f_coeff_b4[8]={-0.1233, 0.9156, 1.1766, 2.6297, 1.8415, 3.375, -1.6317, 0.0025},
f_coeff_c1[8]={2.8, 6.975, 24.7219, 33.3389, 21.0, 14.0, 19.0, 31.0625},
f_coeff_c2[8]={0.6004, 0.1774, -13.0812, -18.3, -4.7656, -0.9999, -5.0, -14.5},
f_coeff_c3[8]={1.2375, 6.4477, -37.7, -62.25, -21.5906, -7.1406, 1.2438, -46.1148},
f_coeff_c4[8]={1.0, -0.1239, 34.8438, 52.0781, 7.2492, 7.5469, -1.9094, 55.375},
f_coeff_d1[8]={1.8734, -1.5798, -5.0, -3.5, -3.5, -3.4, -4.0, -7.2312},
f_coeff_d2[8]={0.6297, -0.5081, 1.5218, 0.0016, -0.1554, -0.1078, 0.025, 0.405},
f_coeff_d3[8]={0.9738, -1.7812, 3.9229, 1.1477, 1.4062, -1.075, 0.3844, 13.35},
f_coeff_d4[8]={0.2809, 0.108, -2.6204, 0.1062, 0.3988, 1.5702, 0.2656, 0.6234},
f_coeff_e1[8]={0.0356, 0.2624, -0.0156, 0.4659, 0.0032, -0.0672, 1.0468, 1.5},
f_coeff_e2[8]={-0.1246, 0.0672, 0.1597, -0.3296, 0.0766, 0.4016, -0.3788, 0.6426},
f_coeff_e3[8]={-0.5718, -0.219, 0.4199, -0.0876, -0.0656, 0.3017, -2.4517, 1.8564},
f_coeff_e4[8]={0.9938, -0.4285, -0.5562, -0.0329, -0.1294, -0.4844, 1.4656, 0.5636};
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[2]-f_zipos[0]))-((f_yipos[2]-f_yipos[0])*(f_zipos[1]-
    f_zipos[0]));
    f_viy=((f_zipos[1]-f_zipos[0])*(f_xipos[2]-f_xipos[0]))-((f_zipos[2]-f_zipos[0])*(f_xipos[1]-
    f_xipos[0]));
    f_viz=((f_xipos[1]-f_xipos[0])*(f_yipos[2]-f_yipos[0]))-((f_xipos[2]-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.);
}
int F_light_pos(double f_longi_t, double f_lati_t, int i_month_t, int i_date_t, double f_time_t,
            double& f_height_t, double& f_azimuth_t)
{
    int i_totalday;
    double f_b, f_sdelta, f_e_t, f_w_t;
    i_totalday=(int)((153*(i_month_t-1)+2*(i_month_t<3)-9*(i_month_t>=3))/5)+i_date_t;
    f_b=(double)(360.*((double)i_totalday-81.)/365.);
    f_sdelta=dasin(0.397949*dsin(f_b));
    f_e_t=0.1645*dsin(2.*f_b) - 0.1255*dcos(f_b)-0.025*dsin(f_b);
    f_w_t=((f_time_t+f_e_t+(f_longi_t-135.)/15)-12.)*15.;
    f_height_t=dasin(dsin(f_lati_t)*dsin(f_sdelta)+dcos(f_lati_t)*dcos(f_sdelta)*dcos(f_w_t));
    f_azimuth_t=dasin(dsin(f_w_t)*dcos(f_sdelta)/dcos(f_height_t));
    f_height_t=f_height_t*0.01745;
    f_azimuth_t=f_azimuth_t*0.01745*(-1.);
    return i_totalday;
}
double calc_dirac_nom_illum(int i_e_t, double f_idn_t, double f_w_t, double f_z_t, double f_delta_t)
{
    double f_dirac_a[8]={57.20, 98.99, 109.83, 110.34, 106.36, 107.19, 105.75, 101.18},
    f_dirac_b[8]={-4.55, -3.46, -4.90, -5.84, -3.97, -1.25, 0.77, 1.58},
    f_dirac_c[8]={-2.98, -1.21, -1.71, -1.99, -1.75, -1.51, -1.26, -1.10},
    f_dirac_d[8]={117.12, 12.38, -8.81, -4.56, -6.16, -26.73, -34.44, -8.29};
}

```

```

double f_edn_t;
f_edn_t=maxdata(0, f_idn_t*(f_direc_a[i_e_t]+f_direc_b[i_e_t]*f_w_t+f_direc_c[i_e_t]*exp(5.73*f_z_t-5.)
+f_direc_d[i_e_t]*f_delta_t));
return f_edn_t;
}
int calc_es(double f_es_t)
{
int i_e_t;
if (f_es_t<=1.065) {i_e_t=0;}
else if (f_es_t<=1.23) {i_e_t=1;}
else if (f_es_t<=1.5) {i_e_t=2;}
else if (f_es_t<=1.95) {i_e_t=3;}
else if (f_es_t<=2.8) {i_e_t=4;}
else if (f_es_t<=4.5) {i_e_t=5;}
else if (f_es_t<=6.2) {i_e_t=6;}
else i_e_t=7;
return i_e_t;
}
double calc_hor_diffu_illum(int i_e_t, double f_id_t, double f_w_t, double f_z_t, double f_delta_t)
{
double f_diffu_a[8]={97.24, 107.22, 104.97, 102.39, 100.71, 106.42, 141.88, 152.23},
f_diffu_b[8]={-0.46, 1.15, 2.96, 5.59, 5.94, 3.83, 1.90, 0.35},
f_diffu_c[8]={12.0, 0.59, -5.53, -13.95, -22.75, -36.15, -53.24, -45.27},
f_diffu_d[8]={-8.91, -3.95, -8.77, -13.90, -23.74, -28.83, -14.03, -7.98};
double f_ekh_t;
f_ekh_t=f_id_t*(f_diffu_a[i_e_t]+f_diffu_b[i_e_t]*f_w_t+f_diffu_c[i_e_t]*exp(f_z_t)+f_diffu_d[i_e_t]
*log(f_delta_t));
return f_ekh_t;
}
double calc_zenith_lumin(int i_e_t, double f_id_t, double f_z_t, double f_delta_t)
{
double f_zenith_a[8]={40.86, 26.58, 19.34, 13.25, 14.47, 19.76, 28.39, 42.91},
f_zenith_c[8]={26.77, 14.73, 2.28, -1.39, -5.09, -3.88, -9.67, -19.62},
f_zenith_cl[8]={-29.59, 58.46, 100.0, 124.79, 160.09, 154.61, 151.58, 130.80},
f_zenith_d[8]={-45.75, -21.25, 0.25, 15.66, 9.13, -19.21, -69.39, -164.08};
double f_lvz_t;
f_lvz_t=f_id_t*(f_zenith_a[i_e_t]+f_zenith_c[i_e_t]*cos(f_z_t)+f_zenith_cl[i_e_t]*exp(f_z_t*(-3))
+f_zenith_d[i_e_t]*f_delta_t);
return f_lvz_t;
}
double calc_slope_diffu_illum(int i_e_t, double f_ekh_t, double f_slopeangle_t, double f_a_t,
double f_z_t, double f_delta_t)
{
double f_diffu_f11[8]={0.11, 0.429, 0.809, 1.014, 1.282, 1.426, 1.485, 1.170},
f_diffu_f12[8]={0.570, 0.363, -0.054, -0.252, -0.420, -0.653, -1.214, -0.3},
f_diffu_f13[8]={-0.081, -0.307, -0.442, -0.531, -0.689, -0.779, -0.784, -0.615},
f_diffu_f21[8]={-0.095, 0.05, 0.181, 0.275, 0.38, 0.425, 0.411, 0.518},
f_diffu_f22[8]={0.158, 0.008, -0.169, -0.35, -0.559, -0.785, -0.629, -1.892},
f_diffu_f23[8]={-0.018, -0.065, -0.092, -0.096, -0.114, -0.097, -0.082, -0.055};
double f_eks_t, f_b_t, f_F1, f_F2;
f_b_t=maxdata(0.087, cos(f_z_t));
f_F1=f_diffu_f11[i_e_t]+f_diffu_f12[i_e_t]*f_delta_t+f_diffu_f13[i_e_t]*f_z_t;
f_F2=f_diffu_f21[i_e_t]+f_diffu_f22[i_e_t]*f_delta_t+f_diffu_f23[i_e_t]*f_z_t;
f_eks_t=f_ekh_t*((1.-f_F1)*(1.+cos(f_slopeangle_t))/2.+f_F1*f_a_t/f_b_t+f_F2*sin(f_slopeangle_t));
return f_eks_t;
}
double calc_relative_lumin(int i_e_t, double f_zenith_t, double f_azimuth_t, double f_pzenith_t,
double f_pazimuth_t, double f_delta_t, double f_lvz_t, double f_lve_t)
{
double f_lv_t, f_sx_t, f_sy_t, f_sz_t, f_px_t, f_py_t, f_pz_t, f_r_t, f_a_t, f_b_t, f_c_t, f_d_t, f_e_t;
f_sx_t=sin(f_zenith_t)*cos(f_azimuth_t);
f_px_t=sin(f_pzenith_t)*cos(f_pazimuth_t);
f_sy_t=sin(f_zenith_t)*sin(f_azimuth_t);
f_py_t=sin(f_pzenith_t)*sin(f_pazimuth_t);
f_sz_t=cos(f_zenith_t);
f_pz_t=cos(f_pzenith_t);
f_r_t=acos((f_sx_t*f_px_t+f_sy_t*f_py_t+f_sz_t*f_pz_t)/(sqrt(f_sx_t*f_sx_t+f_sy_t*f_sy_t+f_sz_t*f_sz_t)
*sqrt(f_px_t*f_px_t+f_py_t*f_py_t+f_pz_t*f_pz_t)));
if (i_e_t!=0)
{
f_a_t=f_coeff_a1[i_e_t]+f_coeff_a2[i_e_t]*f_zenith_t+f_delta_t*(f_coeff_a3[i_e_t]
+f_coeff_a4[i_e_t]*f_zenith_t);
f_b_t=f_coeff_b1[i_e_t]+f_coeff_b2[i_e_t]*f_zenith_t+f_delta_t*(f_coeff_b3[i_e_t]
+f_coeff_b4[i_e_t]*f_zenith_t);
f_c_t=f_coeff_c1[i_e_t]+f_coeff_c2[i_e_t]*f_zenith_t+f_delta_t*(f_coeff_c3[i_e_t]
+f_coeff_c4[i_e_t]*f_zenith_t);
f_d_t=f_coeff_d1[i_e_t]+f_coeff_d2[i_e_t]*f_zenith_t+f_delta_t*(f_coeff_d3[i_e_t]
+f_coeff_d4[i_e_t]*f_zenith_t);
f_e_t=f_coeff_e1[i_e_t]+f_coeff_e2[i_e_t]*f_zenith_t+f_delta_t*(f_coeff_e3[i_e_t]
+f_coeff_e4[i_e_t]*f_zenith_t);
f_lv_t=(1.+f_a_t*exp(f_b_t/cos(f_pzenith_t)))*(1.+f_c_t*exp(f_d_t*f_r_t)+f_e_t*cos(f_r_t)*cos(f_r_t));
f_lv_t=f_lv_t*f_lvz_t/f_lve_t;
return f_lv_t;
}
}

```

```

}
else
{
f_a_t=f_coeff_a1[0]+f_coeff_a2[0]*f_zenith_t+f_delta_t*(f_coeff_a3[0]+f_coeff_a4[0]*f_zenith_t);
f_b_t=f_coeff_b1[0]+f_coeff_b2[0]*f_zenith_t+f_delta_t*(f_coeff_b3[0]+f_coeff_b4[0]*f_zenith_t);
f_c_t=exp(pow((f_delta_t*(f_coeff_c1[0]+f_coeff_c2[0]*f_zenith_t), f_coeff_c3[0]))-1.);
f_d_t=(-1.)*exp(f_delta_t*(f_coeff_d1[0]+f_coeff_d2[0]*f_zenith_t))+f_coeff_d3[0]
+f_delta_t*f_coeff_d4[0];
f_e_t=f_coeff_e1[0]+f_coeff_e2[0]*f_zenith_t+f_delta_t*(f_coeff_e3[0]+f_coeff_e4[0]*f_zenith_t);
f_lv_t=(1.+f_a_t*exp(f_b_t/cos(f_zenith_t)))*(1.+f_c_t*exp(f_d_t*f_r_t)+f_e_t*cos(f_r_t)*cos(f_r_t));
return f_lv_t;
}
}

////////////////////////////////////
// 입력된 평면이 사각형일 때 적용되는 클래스//
////////////////////////////////////
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;
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.);
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;
}
if((f_vx*f_xt+f_vy*f_yt+f_vz*f_zt+f_c)!=0.) return (-1.);
if(f_zpos[0]!=f_zpos[3]){if((f_zt<f_zpos[0])&&(f_zt<f_zpos[3]))||((f_zt>f_zpos[3])
&&(f_zt>f_zpos[0])) return (-1.);}
if(f_xpos[0]!=f_xpos[3]){if((f_xt<f_xpos[0])&&(f_xt<f_xpos[3]))||((f_xt>f_xpos[3])
&&(f_xt>f_xpos[0])) return (-1.);}
}

```

```

if(f_ypos[0]!=f_ypos[3]){if(((f_yt<f_ypos[0])&&(f_yt<f_ypos[3]))||((f_yt>f_ypos[3])
&&(f_yt>f_ypos[0]))) return (-1.);}
f_temp=(f_xt-f_xpos[0])*(f_xt-f_xpos[3])+(f_yt-f_ypos[0])*(f_yt-f_ypos[3])+(f_zt-f_zpos[0])
*(f_zt-f_zpos[3]);
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;
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[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;
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.;
do
{
f_theta_t=((double)rand())/32767.*PI/2.;
}while(cos(f_theta_t)<((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[1]*f_ypolar_axis[0]*f_xpolar_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_ypolar_axis[2]-f_xpolar_axis[2]*f_ypolar_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_tynum_t)
{
i_tnum_t=i_xnum;
i_tynum_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_tynum_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_txnum_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[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]));
if(i_txnum_t==i_xnum) i_txnum_t=i_txnum_t-1;
if(i_tynum_t==i_ynum) i_tynum_t=i_tynum_t-1;
}
void F_calc_origin_xyspace(double f_xt, double f_yt, double f_zt, int& i_txnum_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[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_txnum_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[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]));
if(i_txnum_t==i_xnum) i_txnum_t=i_txnum_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, int& i_reject_diffu_t)
{
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((i_reject_diffu_t==0) && (((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<0.0) 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_reflec==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.;
        do
        {
            f_theta=((double)rand()/32767.)*PI/2.+PI/2.;
        }while(cos(f_theta-PI/2.)<((double)rand()/32767.));
    }
    else
    {
    }
}
else
{
    if(((double)rand()/32767.)<f_spec)
    {
        f_theta=PI-f_theta;
    }
    else
    {
        f_pusai=((double)rand()/32767.)*PI*2.;
        do
        {
            f_theta=((double)rand()/32767.)*PI/2.;
        }while(cos(f_theta)<((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_xpolar_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[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]));
    return (f_area_t);
}
double F_put_value(){return f_value;}
double F_get_reflec(){return f_reflec;}
};

//////////////////////////////////////
// 입력된 평면이 삼각형일 때 적용되는 클래스//
//////////////////////////////////////
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_xint, f_yint, f_zint;
    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_temp=(f_xt-f_xpos[0])*(f_xt-(f_xpos[1]+f_xpos[2]-f_xpos[0]))+(f_yt-f_ypos[0])*(f_yt-(f_ypos[1]
        +f_ypos[2]-f_ypos[0]))+(f_zt-f_zpos[0])*(f_zt-(f_zpos[1]+f_zpos[2]-f_zpos[0]));
    if(f_temp>0.) return (-1.);
    f_xpos[3]=f_xpos[1]+f_xpos[2]-f_xpos[0];
    f_ypos[3]=f_ypos[1]+f_ypos[2]-f_ypos[0];
    f_zpos[3]=f_zpos[1]+f_zpos[2]-f_zpos[0];
    if((f_xt<f_xpos[0]&&f_xt<f_xpos[3])||f_xt>f_xpos[3]&&f_xt>f_xpos[0])return(-1.);
    if((f_yt<f_ypos[0]&&f_yt<f_ypos[3])||f_yt>f_ypos[3]&&f_yt>f_ypos[0])return(-1.);
    if((f_zt<f_zpos[0]&&f_zt<f_zpos[3])||f_zt>f_zpos[3]&&f_zt>f_zpos[0])return(-1.);
    f_r=(f_xt-f_xpos[1]+(f_xpos[2]-f_xpos[0])*(f_ypos[1]-f_yt))/(f_xpos[1]-f_ypos[0])/(f_xpos[1]
        -f_xpos[0])-(f_xpos[2]-f_xpos[0])*(f_ypos[1]-f_ypos[0])/(f_ypos[2]-f_ypos[0]);
    f_xint=f_xpos[1]+(f_xpos[1]-f_xpos[0])*f_r;
    f_yint=f_ypos[1]+(f_ypos[1]-f_ypos[0])*f_r;
    f_zint=f_zpos[1]+(f_zpos[1]-f_zpos[0])*f_r;
    if(sqrt((f_xpos[1]-f_xint)*(f_xpos[1]-f_xint)+(f_ypos[1]-f_yint)*(f_ypos[1]-f_yint)+(f_zpos[1]
        -f_zint)*(f_zpos[1]-f_zint))>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]))<sqrt((f_xt-f_xint)*(f_xt-f_xint)+(f_yt-f_yint)*(f_yt-f_yint)
        +(f_zt-f_zint)*(f_zt-f_zint)))return(-1.);
    if(acos(((f_xpos[0]-f_xpos[1])*(f_xpos[2]-f_xpos[1])+(f_ypos[0]-f_ypos[1])*(f_ypos[2]-f_ypos[1])
        +(f_zpos[0]-f_zpos[1])*(f_zpos[2]-f_zpos[1]))/sqrt((f_xpos[0]-f_xpos[1])*(f_xpos[0]-f_xpos[1])
        *(f_xpos[2]-f_xpos[1])+(f_ypos[0]-f_ypos[1])*(f_ypos[2]-f_ypos[1])+(f_zpos[0]-f_zpos[1])
        *(f_zpos[2]-f_zpos[1]))>sqrt((f_xpos[2]-f_xpos[1])*(f_xpos[2]-f_xpos[1])+(f_ypos[2]-f_ypos[1])
        *(f_ypos[2]-f_ypos[1])+(f_zpos[2]-f_zpos[1])*(f_zpos[2]-f_zpos[1]))<acos(((f_xpos[0]-f_xpos[1])*(f_xt-f_xpos[1])
        +(f_ypos[0]-f_ypos[1])*(f_yt-f_ypos[1])+(f_zpos[0]-f_zpos[1])*(f_zt-f_zpos[1]))/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_xt-f_xpos[1])*(f_xt-f_xpos[1])+(f_yt-f_ypos[1])*(f_yt
        -f_ypos[1])+(f_zt-f_zpos[1])*(f_zt-f_zpos[1]))))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_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.;
do
{
f_theta_t=((double)rand()/32767.)*PI/2.;
}while(cos(f_theta_t)<(((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[1]-f_ypolar_axis[2]*f_xpolar_axis[0]*f_zpolar_axis[1]-f_xpolar_axis[0]*f_ypolar_axis[2]-f_ypolar_axis[1]*f_xpolar_axis[0]*f_zpolar_axis[1]-f_zpolar_axis[0]*f_ypolar_axis[1]*f_xpolar_axis[0]);
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_txnum_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_tynum_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_txnum_t==i_xnum)
i_txnum_t=i_txnum_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, int i_reject_diffu_t)
{
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(i_reject_diffu_t==0&&((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<0.0)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_reflec==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.;
do
{
f_theta=((double)rand()/32767.)*PI/2.+PI/2.;
}while(cos(f_theta-PI/2.)<((double)rand()/32767.));
}
else
{
}
}
else
{
if(((double)rand()/32767.)<f_spec)
{
f_theta=PI-f_theta;
}
else
{
f_pusai=((double)rand()/32767.)*PI*2.;
do
{
f_theta=((double)rand()/32767.)*PI/2.;
}while(cos(f_theta)<((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;}
};

//////////////////////////////////////
// 태양의 위치, 외부 조도 계산과 직사일광 광자 생성 클래스//
//////////////////////////////////////
class C_lightinit
{
private:
int i_lightindex, i_month, i_date, i_e;
double f_time, f_totalrad, f_direcrad, f_diffurad, f_longi, f_lati, f_height, f_azimuth, f_dew_temp,
f_zenith, f_es, f_delta, f_w, f_direc_area, f_slope, f_plain_xmin, f_plain_xmax;
double f_light_zmax, f_light_zmin, f_light_vx, f_light_vy, f_light_vz, f_light_vd, f_edn, f_edh, f_ekh,
f_lvz, f_lve, f_eks, f_a, f_ax, f_b, f_c_t, f_ground_reflec;
public:
C_lightinit(){}
~C_lightinit(){}
void F_lightinput()
{
ifstream o_infilelight("l.txt");
o_infilelight>>f_totalrad;
f_totalrad=f_totalrad/0.0036;
o_infilelight>>f_longi;
o_infilelight>>f_lati;
o_infilelight>>i_month;
o_infilelight>>i_date;
o_infilelight>>f_time;
o_infilelight>>f_dew_temp;
o_infilelight>>f_ground_reflec;
o_infilelight.close();
}
void F_get_direcrad_diffurad()
{
int i_julian_date;
double f_kt;
i_julian_date=F_light_pos(f_longi, f_lati, i_month, i_date, f_time, f_height, f_azimuth);
f_zenith=(PI/2.)-f_height;
f_kt=f_totalrad/(1350.*(1.+0.33*cos(PI*2.*(double)i_julian_date/365.))*cos(f_zenith));
if(f_kt<=0.22)f_diffurad=(1.-0.09*f_kt)*f_totalrad;
else
{
if(f_kt<=0.8)f_diffurad=(0.9511-0.1604*f_kt+4.3888*f_kt*f_kt-16.638*f_kt*f_kt*f_kt
+12.336*f_kt*f_kt*f_kt*f_kt)*f_totalrad;
else f_diffurad=0.165*f_totalrad;
}
f_direcrad=(f_totalrad-f_diffurad)/sin(f_height);
}
void F_lightposi()
{
f_es=((f_diffurad+f_direcrad)/f_diffurad+1.104*f_zenith*f_zenith*f_zenith)
/(1.+1.104*f_zenith*f_zenith*f_zenith);
f_delta=f_diffurad*(1./cos(f_zenith))/(1350.*(1.+0.33*cos(360.*(int)((153.*(i_month-1)+2.

```

```

        *(i_month<3)-9.*(i_month>=3))/5.+i_date)/365.));
f_w=exp(0.07*f_dew_temp-0.075);
i_e=calc_es(f_es);
f_edn=calc_direct_illumination(i_e, f_directrad, f_w, f_zenith, f_delta);
f_edn=f_edn+f_edn*f_ground_reflec;
f_edh=f_edn*cos(f_zenith);
f_ekh=calc_hor_diffusion_illumination(i_e, f_diffurad, f_w, f_zenith, f_delta);
f_lvz=calc_zenith_luminance(i_e, f_diffurad, f_zenith, f_delta);
f_lvz=f_lvz+f_lvz*f_ground_reflec;
f_ekh=f_ekh+f_ekh*f_ground_reflec;
f_lve=f_ekh/PI;
}
void F_get_edh_ekh(double& f_edh_t, double& f_ekh_t)
{
    f_edh_t=f_edh;
    f_ekh_t=f_ekh;
}
void F_getdirec_maxmin(double f_xmax_t, double f_ymax_t, double f_zmax_t, double f_xmin_t,
    double f_ymin_t, double f_zmin_t)
{
    int i_j, i_k, i_l;
    double f_light_r, f_xm[2], f_ym[2], f_zm[2], f_xm0, f_xm1, f_xm2, f_xm3, f_ym0, f_ym1, f_ym2, f_ym3,
        f_zm0, f_zm1, f_zm2, f_zm3, f_l, f_m, f_n;
    f_xm[0]=f_xmax_t;
    f_xm[1]=f_xmin_t;
    f_ym[0]=f_ymax_t;
    f_ym[1]=f_ymin_t;
    f_zm[0]=f_zmax_t;
    f_zm[1]=f_zmin_t;
    f_light_r=sqrt(f_xmax_t*f_xmax_t+f_ymax_t*f_ymax_t)/cos(f_height);
    f_light_r=1.;
    f_light_vx=f_light_r*sin(f_zenith)*cos(f_azimuth);
    f_light_vy=f_light_r*sin(f_zenith)*sin(f_azimuth);
    f_light_vz=f_light_r*cos(f_zenith);
    if(f_azimuth>=0.)
    {
        if(f_azimuth<=(PI/2.))
        {
            f_light_vd=(f_light_vx*f_xmax_t+f_light_vy*f_ymax_t+f_light_vz*f_zmax_t)*(-1.);
        }
        else
        {
            f_light_vd=(f_light_vx*f_xmin_t+f_light_vy*f_ymax_t+f_light_vz*f_zmax_t)*(-1.);
        }
    }
    else
    {
        if(f_azimuth>=(PI/2.*(-1.)))
        {
            f_light_vd=(f_light_vx*f_xmax_t+f_light_vy*f_ymin_t+f_light_vz*f_zmax_t)*(-1.);
        }
        else
        {
            f_light_vd=(f_light_vx*f_xmin_t+f_light_vy*f_ymin_t+f_light_vz*f_zmax_t)*(-1.);
        }
    }
    f_light_r=(f_light_vx*f_xmax_t+f_light_vy*f_ymax_t+f_light_vz*f_zmax_t+f_light_vd)/(f_light_vx
        *f_light_vx+f_light_vy*f_light_vy+f_light_vz*f_light_vz)*(-1.);
    f_light_zmax=f_zmax_t+f_light_vz*f_light_r;
    f_light_zmin=f_light_zmax;
    for(i_j=0; i_j<=1; i_j++)
    {
        for(i_k=0; i_k<=1; i_k++)
        {
            for(i_l=0; i_l<=1; i_l++)
            {
                f_light_r=(f_light_vx*f_xm[i_j]+f_light_vy*f_ym[i_k]+f_light_vz*f_zm[i_l]+f_light_vd)
                    /(f_light_vx*f_light_vx+f_light_vy*f_light_vy+f_light_vz*f_light_vz)*(-1.);
                f_light_zmax=maxdata(f_light_zmax, (f_zm[i_l]+f_light_vz*f_light_r));
                f_light_zmin=mindata(f_light_zmin, (f_zm[i_l]+f_light_vz*f_light_r));
            }
        }
    }
}
if(f_azimuth>=0.)
{
    if(f_azimuth<=(PI/2.))
    {
        f_light_r=(f_light_vx*f_xmax_t+f_light_vy*f_ymin_t+f_light_vz*f_zmax_t+f_light_vd)/(f_light_vx
            *f_light_vx+f_light_vy*f_light_vy+f_light_vz*f_light_vz)*(-1.);
        f_ym1=f_ymin_t+f_light_vy*f_light_r;
        f_xm1=f_xmax_t+f_light_vx*f_light_r;
        f_zm1=f_zmax_t+f_light_vz*f_light_r;
        f_light_r=(f_light_vx*f_xmax_t+f_light_vy*f_ymin_t+f_light_vz*f_zmin_t+f_light_vd)/(f_light_vx

```

```

*f_light_vx+f_light_vy*f_light_vy+f_light_vz*f_light_vz)*(-1.);
f_xm0=f_xmax_t+f_light_vx*f_light_r;
f_ym0=f_ymin_t+f_light_vy*f_light_r;
f_zm0=f_zmin_t+f_light_vz*f_light_r;
f_l=f_xm1-f_xm0;
f_m=f_ym1-f_ym0;
f_n=f_zm1-f_zm0;
f_xm1=(f_light_zmax-f_zm0)*f_l/f_n+f_xm0;
f_ym1=(f_light_zmax-f_zm0)*f_m/f_n+f_ym0;
f_xm0=(f_light_zmin-f_zm0)*f_l/f_n+f_xm0;
f_ym0=(f_light_zmin-f_zm0)*f_m/f_n+f_ym0;
f_zm0=f_light_zmin;
f_zm1=f_light_zmax;
f_light_r=(f_light_vx*f_xmin_t+f_light_vy*f_ymax_t+f_light_vz*f_zmax_t+f_light_vd)/(f_light_vx
*f_light_vx+f_light_vy*f_light_vy+f_light_vz*f_light_vz)*(-1.);
f_xm3=f_xmin_t+f_light_vx*f_light_r;
f_ym3=f_ymax_t+f_light_vy*f_light_r;
f_zm3=f_zmax_t+f_light_vz*f_light_r;
f_light_r=(f_light_vx*f_xmin_t+f_light_vy*f_ymax_t+f_light_vz*f_zmin_t+f_light_vd)/(f_light_vx
*f_light_vx+f_light_vy*f_light_vy+f_light_vz*f_light_vz)*(-1.);
f_xm2=f_xmin_t+f_light_vx*f_light_r;
f_ym2=f_ymax_t+f_light_vy*f_light_r;
f_zm2=f_zmin_t+f_light_vz*f_light_r;
f_l=f_xm3-f_xm2;
f_m=f_ym3-f_ym2;
f_n=f_zm3-f_zm2;
f_xm3=(f_light_zmax-f_zm2)*f_l/f_n+f_xm2;
f_ym3=(f_light_zmax-f_zm2)*f_m/f_n+f_ym2;
f_xm2=(f_light_zmin-f_zm2)*f_l/f_n+f_xm2;
f_ym2=(f_light_zmin-f_zm2)*f_m/f_n+f_ym2;
f_zm2=f_light_zmin;
f_zm3=f_light_zmax;
f_direct_area=sqrt((f_xm0-f_xm2)*(f_xm0-f_xm2)+(f_ym2-f_ym0)*(f_ym2-f_ym0))*sqrt((f_xm0-f_xm1)
*(f_xm0-f_xm1)+(f_ym0-f_ym1)*(f_ym0-f_ym1)+(f_zm1-f_zm0)*(f_zm1-f_zm0));
f_ax=(f_ym0-f_ym2)/(f_xm0-f_xm2);
f_b=f_ym0-f_xm0*f_ax;
f_c_t=sqrt((f_xm0-f_xm1)*(f_xm0-f_xm1)+(f_ym0-f_ym1)*(f_ym0-f_ym1)+(f_zm0-f_zm1)*(f_zm0-f_zm1))
/cos(f_height);
f_plain_xmax=maxdata(f_xm0, f_xm2);
f_plain_xmin=mindata(f_xm0, f_xm2);
}
else
{
f_light_r=(f_light_vx*f_xmax_t+f_light_vy*f_ymax_t+f_light_vz*f_zmax_t+f_light_vd)/(f_light_vx
*f_light_vx+f_light_vy*f_light_vy+f_light_vz*f_light_vz)*(-1.);
f_ym1=f_ymax_t+f_light_vy*f_light_r;
f_xm1=f_xmax_t+f_light_vx*f_light_r;
f_zm1=f_zmax_t+f_light_vz*f_light_r;
f_light_r=(f_light_vx*f_xmax_t+f_light_vy*f_ymax_t+f_light_vz*f_zmin_t+f_light_vd)/(f_light_vx
*f_light_vx+f_light_vy*f_light_vy+f_light_vz*f_light_vz)*(-1.);
f_xm0=f_xmax_t+f_light_vx*f_light_r;
f_ym0=f_ymax_t+f_light_vy*f_light_r;
f_zm0=f_zmin_t+f_light_vz*f_light_r;
f_l=f_xm1-f_xm0;
f_m=f_ym1-f_ym0;
f_n=f_zm1-f_zm0;
f_xm1=(f_light_zmax-f_zm0)*f_l/f_n+f_xm0;
f_ym1=(f_light_zmax-f_zm0)*f_m/f_n+f_ym0;
f_xm0=(f_light_zmin-f_zm0)*f_l/f_n+f_xm0;
f_ym0=(f_light_zmin-f_zm0)*f_m/f_n+f_ym0;
f_zm0=f_light_zmin;
f_zm1=f_light_zmax;
f_light_r=(f_light_vx*f_xmin_t+f_light_vy*f_ymin_t+f_light_vz*f_zmax_t+f_light_vd)/(f_light_vx
*f_light_vx+f_light_vy*f_light_vy+f_light_vz*f_light_vz)*(-1.);
f_xm3=f_xmin_t+f_light_vx*f_light_r;
f_ym3=f_ymin_t+f_light_vy*f_light_r;
f_zm3=f_zmax_t+f_light_vz*f_light_r;
f_light_r=(f_light_vx*f_xmin_t+f_light_vy*f_ymin_t+f_light_vz*f_zmin_t+f_light_vd)/(f_light_vx
*f_light_vx+f_light_vy*f_light_vy+f_light_vz*f_light_vz)*(-1.);
f_xm2=f_xmin_t+f_light_vx*f_light_r;
f_ym2=f_ymin_t+f_light_vy*f_light_r;
f_zm2=f_zmin_t+f_light_vz*f_light_r;
f_l=f_xm3-f_xm2;
f_m=f_ym3-f_ym2;
f_n=f_zm3-f_zm2;
f_xm3=(f_light_zmax-f_zm2)*f_l/f_n+f_xm2;
f_ym3=(f_light_zmax-f_zm2)*f_m/f_n+f_ym2;
f_xm2=(f_light_zmin-f_zm2)*f_l/f_n+f_xm2;
f_ym2=(f_light_zmin-f_zm2)*f_m/f_n+f_ym2;
f_zm2=f_light_zmin;
f_zm3=f_light_zmax;
f_direct_area=sqrt((f_xm0-f_xm2)*(f_xm0-f_xm2)+(f_ym2-f_ym0)*(f_ym2-f_ym0))*sqrt((f_xm0-f_xm1)
*(f_xm0-f_xm1)+(f_ym0-f_ym1)*(f_ym0-f_ym1)+(f_zm1-f_zm0)*(f_zm1-f_zm0));
f_ax=(f_ym0-f_ym2)/(f_xm0-f_xm2);

```

```

f_b=f_y0-f_x0*f_ax;
f_c_t=sqrt((f_x0-f_xm1)*(f_x0-f_xm1)+(f_y0-f_ym1)*(f_y0-f_ym1)+(f_z0-f_zm1)*(f_z0-f_zm1))
/cos(f_height);
f_plain_xmax=maxdata(f_xm0, f_xm2);
f_plain_xmin=mindata(f_xm0, f_xm2);
}
else
{
if (f_azimuth>=(PI/2.*(-1.)))
{
f_light_r=(f_light_vx*f_xmin_t+f_light_vy*f_ymin_t+f_light_vz*f_zmax_t+f_light_vd)/(f_light_vx
*f_light_vx+f_light_vy*f_light_vy+f_light_vz*f_light_vz)*(-1.);
f_ym1=f_ymin_t+f_light_vy*f_light_r;
f_xm1=f_xmin_t+f_light_vx*f_light_r;
f_zm1=f_zmax_t+f_light_vz*f_light_r;
f_light_r=(f_light_vx*f_xmin_t+f_light_vy*f_ymin_t+f_light_vz*f_zmin_t+f_light_vd)/(f_light_vx
*f_light_vx+f_light_vy*f_light_vy+f_light_vz*f_light_vz)*(-1.);
f_xm0=f_xmin_t+f_light_vx*f_light_r;
f_y0=f_ymin_t+f_light_vy*f_light_r;
f_z0=f_zmin_t+f_light_vz*f_light_r;
f_l=f_xm1-f_xm0;
f_m=f_ym1-f_y0;
f_n=f_zm1-f_z0;
f_xm1=(f_light_zmax-f_z0)*f_l/f_n+f_xm0;
f_ym1=(f_light_zmax-f_z0)*f_m/f_n+f_y0;
f_xm0=(f_light_zmin-f_z0)*f_l/f_n+f_xm0;
f_y0=(f_light_zmin-f_z0)*f_m/f_n+f_y0;
f_z0=f_light_zmin;
f_zm1=f_light_zmax;
f_light_r=(f_light_vx*f_xmax_t+f_light_vy*f_ymax_t+f_light_vz*f_zmax_t+f_light_vd)/(f_light_vx
*f_light_vx+f_light_vy*f_light_vy+f_light_vz*f_light_vz)*(-1.);
f_xm3=f_xmax_t+f_light_vx*f_light_r;
f_ym3=f_ymax_t+f_light_vy*f_light_r;
f_zm3=f_zmax_t+f_light_vz*f_light_r;
f_light_r=(f_light_vx*f_xmax_t+f_light_vy*f_ymax_t+f_light_vz*f_zmin_t+f_light_vd)/(f_light_vx
*f_light_vx+f_light_vy*f_light_vy+f_light_vz*f_light_vz)*(-1.);
f_xm2=f_xmax_t+f_light_vx*f_light_r;
f_ym2=f_ymax_t+f_light_vy*f_light_r;
f_zm2=f_zmin_t+f_light_vz*f_light_r;
f_l=f_xm3-f_xm2;
f_m=f_ym3-f_ym2;
f_n=f_zm3-f_zm2;
f_xm3=(f_light_zmax-f_zm2)*f_l/f_n+f_xm2;
f_ym3=(f_light_zmax-f_zm2)*f_m/f_n+f_y2;
f_xm2=(f_light_zmin-f_zm2)*f_l/f_n+f_xm2;
f_ym2=(f_light_zmin-f_zm2)*f_m/f_n+f_y2;
f_zm2=f_light_zmin;
f_zm3=f_light_zmax;
f_direct_area=sqrt((f_xm0-f_xm2)*(f_xm0-f_xm2)+(f_ym2-f_y0)*(f_ym2-f_y0))*sqrt((f_xm0-f_xm1)
*(f_xm0-f_xm1)+(f_y0-f_ym1)*(f_y0-f_ym1)+(f_zm1-f_zm0)*(f_zm1-f_zm0));
f_ax=(f_y0-f_ym2)/(f_xm0-f_xm2);
f_b=f_y0-f_xm0*f_ax;
f_c_t=sqrt((f_xm0-f_xm1)*(f_xm0-f_xm1)+(f_y0-f_ym1)*(f_y0-f_ym1)+(f_z0-f_zm1)*(f_z0-f_zm1))
/cos(f_height);
f_plain_xmax=maxdata(f_xm0, f_xm2);
f_plain_xmin=mindata(f_xm0, f_xm2);
}else
{
f_light_r=(f_light_vx*f_xmin_t+f_light_vy*f_ymax_t+f_light_vz*f_zmax_t+f_light_vd)/(f_light_vx
*f_light_vx+f_light_vy*f_light_vy+f_light_vz*f_light_vz)*(-1.);
f_ym1=f_ymax_t+f_light_vy*f_light_r;
f_xm1=f_xmin_t+f_light_vx*f_light_r;
f_zm1=f_zmax_t+f_light_vz*f_light_r;
f_light_r=(f_light_vx*f_xmin_t+f_light_vy*f_ymax_t+f_light_vz*f_zmin_t+f_light_vd)/(f_light_vx
*f_light_vx+f_light_vy*f_light_vy+f_light_vz*f_light_vz)*(-1.);
f_xm0=f_xmin_t+f_light_vx*f_light_r;
f_y0=f_ymax_t+f_light_vy*f_light_r;
f_z0=f_zmin_t+f_light_vz*f_light_r;
f_l=f_xm1-f_xm0;
f_m=f_ym1-f_y0;
f_n=f_zm1-f_z0;
f_xm1=(f_light_zmax-f_zm0)*f_l/f_n+f_xm0;
f_ym1=(f_light_zmax-f_zm0)*f_m/f_n+f_y0;
f_xm0=(f_light_zmin-f_zm0)*f_l/f_n+f_xm0;
f_y0=(f_light_zmin-f_zm0)*f_m/f_n+f_y0;
f_z0=f_light_zmin;
f_zm1=f_light_zmax;
f_light_r=(f_light_vx*f_xmax_t+f_light_vy*f_ymin_t+f_light_vz*f_zmax_t+f_light_vd)/(f_light_vx
*f_light_vx+f_light_vy*f_light_vy+f_light_vz*f_light_vz)*(-1.);
f_xm3=f_xmax_t+f_light_vx*f_light_r;
f_ym3=f_ymin_t+f_light_vy*f_light_r;
f_zm3=f_zmax_t+f_light_vz*f_light_r;
}
}
}
}

```

```

    f_light_r=(f_light_vx*f_xmax_t+f_light_vy*f_ymin_t+f_light_vz*f_zmin_t+f_light_vd)/(f_light_vx
        *f_light_vx*f_light_vy*f_light_vy+f_light_vz*f_light_vz)*(-1.);
    f_xm2=f_xmax_t+f_light_vx*f_light_r;
    f_ym2=f_ymin_t+f_light_vy*f_light_r;
    f_zm2=f_zmin_t+f_light_vz*f_light_r;
    f_l=f_xm3-f_xm2;
    f_m=f_ym3-f_ym2;
    f_n=f_zm3-f_zm2;
    f_xm3=(f_light_zmax-f_zm2)*f_l/f_n+f_xm2;
    f_ym3=(f_light_zmax-f_zm2)*f_m/f_n+f_ym2;
    f_xm2=(f_light_zmin-f_zm2)*f_l/f_n+f_xm2;
    f_ym2=(f_light_zmin-f_zm2)*f_m/f_n+f_ym2;
    f_zm2=f_light_zmin;
    f_zm3=f_light_zmax;
    f_direct_area=sqrt((f_xm0-f_xm2)*(f_xm0-f_xm2)+(f_ym2-f_ym0)*(f_ym2-f_ym0))*sqrt((f_xm0-f_xm1)
        *(f_xm0-f_xm1)+(f_ym0-f_ym1)*(f_ym0-f_ym1)+(f_zm1-f_zm0)*(f_zm1-f_zm0));
    f_ax=(f_ym0-f_ym2)/(f_xm0-f_xm2);
    f_b=f_ym0-f_xm0*f_ax;
    f_c_t=sqrt((f_xm0-f_xm1)*(f_xm0-f_xm1)+(f_ym0-f_ym1)*(f_ym0-f_ym1)+(f_zm0-f_zm1)*(f_zm0-f_zm1))
        /cos(f_height);
    f_plain_xmax=maxdata(f_xm0, f_xm2);
    f_plain_xmin=mindata(f_xm0, f_xm2);
}
}
f_light_vx=f_light_vx*(-1.);
f_light_vy=f_light_vy*(-1.);
f_light_vz=f_light_vz*(-1.);
}
void F_calc_slopeangle(double f_d_p_vx, double f_d_p_vy, double f_d_p_vz)
{
    f_slope=acos(1.0*f_d_p_vz/(sqrt(f_d_p_vx*f_d_p_vx+f_d_p_vy*f_d_p_vy+f_d_p_vz*f_d_p_vz)));
    if(f_d_p_vz<0.)f_slope=PI-f_slope;
    if (f_slope==0.)f_eks=f_ekh;
    else{
        if((f_d_p_vx*f_light_vx+f_d_p_vy*f_light_vy+f_d_p_vz*f_light_vz)>=0.)f_a=0.;
        else f_a=(f_d_p_vx*f_light_vx*(-1.)+f_d_p_vy*f_light_vy*(-1.)+f_d_p_vz*f_light_vz*(-1.))
            /(sqrt(f_d_p_vx*f_d_p_vx+f_d_p_vy*f_d_p_vy+f_d_p_vz*f_d_p_vz)*sqrt(f_light_vx*f_light_vx
                +f_light_vy*f_light_vy+f_light_vz*f_light_vz));
        f_eks=calc_slope_diffu_illumination(i_e, f_ekh, f_slope, f_a, f_zenith, f_delta);
    }
}
double F_direct_photon_weight(double f_photon_t)
{
    return (f_edn*f_direct_area/f_photon_t);
}
double F_diffu_photon_weight(double f_d_area_t, double f_photon_t)
{
    return (f_eks*f_d_area_t/f_photon_t);
}
double F_ground_photon_weight(double f_d_area_t, double f_photon_t)
{
    return ((f_edh+f_ekh)*f_ground_reflec*f_d_area_t/f_photon_t);
}
void F_gen_direct_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)
{
    void randomize();
    f_d_p_vx=f_light_vx;
    f_d_p_vy=f_light_vy;
    f_d_p_vz=f_light_vz;
    f_d_p_px=(f_plain_xmax-f_plain_xmin)*(((double)rand())/32767.)+f_plain_xmin;
    f_d_p_py=f_ax*f_d_p_px+f_b;
    f_d_p_pz=f_c_t*(((double)rand())/32767.)+f_light_zmin;
}
double F_get_diffu_lumin(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_pazimuth, f_pzenith, f_relative_lumin_t;
    f_d_p_px=f_d_p_px+f_d_p_vx;
    f_d_p_py=f_d_p_py+f_d_p_vy;
    f_d_p_pz=f_d_p_pz+f_d_p_vz;
    f_pzenith=atan(sqrt(f_d_p_vx*f_d_p_vx+f_d_p_vy*f_d_p_vy)/f_d_p_vz);
    if(f_d_p_vz<0.)f_pzenith=PI+f_pzenith;
    f_pazimuth=atan(f_d_p_vy/f_d_p_vx);
    if(f_d_p_vx<0.&&f_d_p_vy>0.)f_pazimuth=PI+f_pazimuth;
    if(f_d_p_vx<0.&&f_d_p_vy<0.)f_pazimuth=f_pazimuth-PI;
    f_relative_lumin_t= calc_relative_lumin(i_e, f_zenith, f_azimuth, f_pzenith, f_pazimuth, f_delta,
        f_lvz, f_lve);
    return f_relative_lumin_t;
}
};

```

```

//////////
// MAIN FUNCTION//
//////////
void main()
{
    C_plain_rectangle* p_numofplain_r[100];
    double* p_value_r[100][10][10];
    C_plain_triangle* p_numofplain_t[50];
    double* p_value_t[50][10][10];
    int i_num, i_i, i_ij, i_ij2, i_typeslope, i_typeplain, i_r, i_t, i_txnum, i_tynum, i_plain_index,
        i_plain_num, i_photon_continue, i_once, i_once_t, i_diffu, i_first_plain, i_reject_diffu,
        i_photon_continue_d, i_first_plain_d, i_plain_index_d, i_plain_num_d;
    double f_txpos[4], f_typos[4], f_tzpos[4], f_ttrans, f_treflec, f_tspec, f_i, f_dist, f_dist_temp,
        f_area, f_d_area;
    double f_max_xpos, f_max_ypos, f_max_zpos, f_min_xpos, f_min_ypos, f_min_zpos, f_phonum,
        f_relative_lumin;
    double f_dirac_phovx, f_dirac_phovy, f_dirac_phovz, f_dirac_phopx, f_dirac_phopy, f_dirac_phopz,
        f_phoweight, f_origin_phopx, f_origin_phopy, f_origin_phopz, f_origin_phovx,
        f_ground_phoweight;
    double f_dist_d, f_dist_temp_d, f_Edh, f_Ekh, f_ratio, f_origin_phovy, f_origin_phovz;
    ifstream o_in("a.txt");
    ofstream o_out("a.out");
    o_in>>i_num;
    i_r=0; i_t=0;
    f_max_xpos=0.0; f_max_ypos=0.0; f_max_zpos=0.0; f_min_xpos=30000.0; f_min_ypos=30000.0;
    f_min_zpos=30000.0;
    for (i_i=0; i_i<i_num; i_i++)
    {
        o_in >> i_typeplain;
        switch (i_typeplain)
        {
            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];
                    f_max_xpos=maxdata(f_max_xpos, f_txpos[i_ij]);
                    f_max_ypos=maxdata(f_max_ypos, f_typos[i_ij]);
                    f_max_zpos=maxdata(f_max_zpos, f_tzpos[i_ij]);
                    f_min_xpos=mindata(f_min_xpos, f_txpos[i_ij]);
                    f_min_ypos=mindata(f_min_ypos, f_typos[i_ij]);
                    f_min_zpos=mindata(f_min_zpos, 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:
            {
                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];
                    f_max_xpos=maxdata(f_max_xpos, f_txpos[i_ij]);
                    f_max_ypos=maxdata(f_max_ypos, f_typos[i_ij]);
                    f_max_zpos=maxdata(f_max_zpos, f_tzpos[i_ij]);
                    f_min_xpos=mindata(f_min_xpos, f_txpos[i_ij]);
                    f_min_ypos=mindata(f_min_ypos, f_typos[i_ij]);
                    f_min_zpos=mindata(f_min_zpos, 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;
}
if(i_i==0)i_typeslope=i_typeplain;
}
switch(i_typeslope)
{
case 3:
{
p_numofplain_t[0]->F_getslope_vector(f_direct_phovx, f_direct_phovy, f_direct_phovz);
break;
}
case 4:
{
p_numofplain_r[0]->F_getslope_vector(f_direct_phovx, f_direct_phovy, f_direct_phovz);
break;
}
default : break;
}
}
C_lightinit o_light;
o_light.F_lightinput();
o_light.F_get_directrad_diffurad();
o_light.F_lightposi();
i_diffu=1;
i_reject_diffu=0;
o_light.F_getdirect_maxmin(f_max_xpos, f_max_ypos, f_max_zpos, f_min_xpos, f_min_ypos, f_min_zpos);
o_light.F_calc_slopeangle(f_direct_phovx, f_direct_phovy, f_direct_phovz);

////////////////////////////////////
// 직사일광에 대한 모델링//
////////////////////////////////////
f_phonum=100000.;
f_phoweight=o_light.F_direct_photon_weight(f_phonum);
for(f_i=0.;f_i<f_phonum;f_i++)
{
i_photon_continue=1;
i_first_plain=-1;
o_light.F_gen_direct_pho(f_direct_phovx, f_direct_phovy, f_direct_phovz, f_direct_phopx, f_direct_phopy,
f_direct_phopz);
while(i_photon_continue>0)
{
f_dist=-1.;
i_plain_index=-1;
i_plain_num=-1;
f_dist_temp=-1.;
for(i_i=0;i_i<i_t;i_i++)
{
if(i_i!=0||i_first_plain!=-1)
f_dist_temp=p_numofplain_t[i_i]->F_calc_photon(f_direct_phovx, f_direct_phovy, f_direct_phovz,
f_direct_phopx, f_direct_phopy, f_direct_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_i;
i_first_plain=1;
}
}
}
}
for(i_i=0;i_i<i_r;i_i++)

```

```

{
  if(i_i!=0||i_first_plain!=-1)
    f_dist_temp=p_numofplain_r[i_i]->F_calc_photon(f_direct_phovx, f_direct_phovy, f_direct_phovz,
                                                  f_direct_phopx, f_direct_phopy, f_direct_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_i;
      i_first_plain=1;
    }
  }
  if(f_dist>0)
  {
    switch(i_plain_index)
    {
    case 4:
    {
      p_numofplain_r[i_plain_num]->F_exact_grid(i_tnxnum, i_tynum);
      if(i_tnxnum==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_direct_phovx,
                               f_direct_phovy, f_direct_phovz, f_direct_phopx, f_direct_phopy,
                               f_direct_phopz, i_reject_diffu);
      }
      else
      {
        p_numofplain_r[i_plain_num]->F_calc_xyspace(f_direct_phovx, f_direct_phovy, f_direct_phovz,
                                                    f_direct_phopx, f_direct_phopy, f_direct_phopz, i_tnxnum, i_tynum);
        *p_value_r[i_plain_num][i_tnxnum][i_tynum]= *p_value_r[i_plain_num][i_tnxnum][i_tynum]
                                                    +f_phoweight;
        i_photon_continue=p_numofplain_r[i_plain_num]->F_trans_polaraxis(f_direct_phovx,
                               f_direct_phovy, f_direct_phovz, f_direct_phopx, f_direct_phopy,
                               f_direct_phopz, i_reject_diffu);
      }
    }
    break;
  }
  case 3:
  {
    p_numofplain_t[i_plain_num]->F_exact_grid(i_tnxnum, i_tynum);
    if(i_tnxnum==1&&i_tynum==1)
    {
      p_numofplain_t[i_plain_num]->F_add_value(f_phoweight, 1.0);
      i_photon_continue=p_numofplain_t[i_plain_num]->F_trans_polaraxis(f_direct_phovx,
                               f_direct_phovy, f_direct_phovz, f_direct_phopx, f_direct_phopy,
                               f_direct_phopz, i_reject_diffu);
    }
    else
    {
      p_numofplain_t[i_plain_num]->F_calc_xyspace(f_direct_phovx, f_direct_phovy, f_direct_phovz,
                                                  f_direct_phopx, f_direct_phopy, f_direct_phopz, i_tnxnum, i_tynum);
      *p_value_t[i_plain_num][i_tnxnum][i_tynum]= *p_value_t[i_plain_num][i_tnxnum][i_tynum]
                                                  +f_phoweight;
      i_photon_continue=p_numofplain_t[i_plain_num]->F_trans_polaraxis(f_direct_phovx,
                               f_direct_phovy, f_direct_phovz, f_direct_phopx, f_direct_phopy, f_direct_phopz,
                               i_reject_diffu);
    }
  }
  break;
}
default: break;
}
else
{
  i_photon_continue=-1;
}
}
}
o_light.F_get_edh_ekh(f_Edh, f_Ekh);
o_out<<"Sunlight Edh = "<<f_Edh<<"\n";
for(i_i=0; i_i<i_r; i_i++)
{
  p_numofplain_r[i_i]->F_xyinum(i_tnxnum, i_tynum);
  f_area=p_numofplain_r[i_i]->F_get_area();
  if(i_tnxnum==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_tnxnum; 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]
        *i_txnum*i_tynum/(f_area*f_Edh))<<"\n";
    *p_value_r[i_i][i_ij][i_ij2]=0.;
    }
    }
}
for(i_i=0;i_i<i_t;i_i++)
{
p_numofplain_t[i_i]->F_xynum(i_txnum, i_tynum);
if(i_txnum==1&&i_tynum==1)
{
f_area=p_numofplain_t[i_i]->F_nogrid_get_area();
o_out<<"triangle plain["<<i_i<<"] = "<<(p_numofplain_t[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++)
{
f_area=p_numofplain_t[i_i]->F_grid_get_area(i_ij, i_ij2);
o_out<<"triangle plain["<<i_i<<"]["<<i_ij<<"]["<<i_ij2<<"] =
    "<<(*p_value_t[i_i][i_ij][i_ij2]/f_area)<<" lx\n";
*p_value_t[i_i][i_ij][i_ij2]=0.;
}
}
}
}
}
switch(i_typeslope)
{
case 4:
{
f_d_area=p_numofplain_r[0]->F_get_area();
break;
}
case 3:
{
f_d_area=p_numofplain_t[0]->F_nogrid_get_area();
break;
}
default : break;
}
}

////////////////////////////////////
// 천공광에 대한 모델링//
////////////////////////////////////
f_pho_num=50000.;
f_pho_weight=o_light.F_diffu_photon_weight(f_d_area, f_pho_num);
f_ground_pho_weight=o_light.F_ground_photon_weight(f_d_area, f_pho_num);
f_ratio=0.;
for (f_i=0.;f_i<f_pho_num;f_i=f_i+1.)
{
i_photon_continue=1;
i_diffu=-1;
i_once=0;
i_once_t=0;
f_ratio=f_ratio+1.;
switch(i_typeslope)
{
case 4:
{
p_numofplain_r[0]->F_gen_diffu_pho(f_direct_pho_vx, f_direct_pho_vy, f_direct_pho_vz, f_direct_pho_px,
    f_direct_pho_py, f_direct_pho_pz);

f_origin_pho_px=f_direct_pho_px;
f_origin_pho_py=f_direct_pho_py;
f_origin_pho_pz=f_direct_pho_pz;
f_origin_pho_vx=f_direct_pho_vx;
f_origin_pho_vy=f_direct_pho_vy;
f_origin_pho_vz=f_direct_pho_vz;
break;
}
case 3:
{
p_numofplain_t[0]->F_gen_diffu_pho(f_direct_pho_vx, f_direct_pho_vy, f_direct_pho_vz, f_direct_pho_px,
    f_direct_pho_py, f_direct_pho_pz);

f_origin_pho_px=f_direct_pho_px;
f_origin_pho_py=f_direct_pho_py;
f_origin_pho_pz=f_direct_pho_pz;
f_origin_pho_vx=f_direct_pho_vx;
}
}
}
}

```

```

    f_origin_pho_vy=f_dirac_pho_vy;
    f_origin_pho_vz=f_dirac_pho_vz;
    break;
}
default : break;
}
while(i_photon_continue>0)
{
f_dist=-1.;
i_plain_index=-1;
i_plain_num=-1;
for(i_i=0;i_i<i_r;i_i++)
{
    f_dist_temp=p_numofplain_r[i_i]->F_calc_photon(f_dirac_pho_vx, f_dirac_pho_vy, f_dirac_pho_vz,
                                                    f_dirac_pho_px, f_dirac_pho_py, f_dirac_pho_pz);
    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_i;
        }
    }
}
for(i_i=0;i_i<i_t;i_i++)
{
    f_dist_temp=p_numofplain_t[i_i]->F_calc_photon(f_dirac_pho_vx, f_dirac_pho_vy, f_dirac_pho_vz,
                                                    f_dirac_pho_px, f_dirac_pho_py, f_dirac_pho_pz);
    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_i;
        }
    }
}
if(f_dist>0.)
{
    i_once_t=1;
    switch(i_plain_index)
    {
    case 4:
    {
        if(i_diffu==1) i_photon_continue=p_numofplain_r[i_plain_num]->F_trans_polaraxis(f_dirac_pho_vx,
                                                    f_dirac_pho_vy, f_dirac_pho_vz, f_dirac_pho_px, f_dirac_pho_py, f_dirac_pho_pz,
                                                    i_reject_diffu);
        else
        {
            p_numofplain_r[i_plain_num]->F_exact_grid(i_tynum, i_tynum);
            if(i_tynum==1&&i_tynum==1)
            {
                p_numofplain_r[i_plain_num]->F_add_value(f_pho_weight, f_relative_lumin);
                i_photon_continue=p_numofplain_r[i_plain_num]->F_trans_polaraxis(f_dirac_pho_vx,
                                                            f_dirac_pho_vy, f_dirac_pho_vz, f_dirac_pho_px, f_dirac_pho_py,
                                                            f_dirac_pho_pz, i_reject_diffu);
            }
        }
    }
    case 3:
    {
        if(i_diffu==1) i_photon_continue=p_numofplain_t[i_plain_num]->F_trans_polaraxis(f_dirac_pho_vx,
                                                    f_dirac_pho_vy, f_dirac_pho_vz, f_dirac_pho_px, f_dirac_pho_py, f_dirac_pho_pz,
                                                    i_reject_diffu);
        else
        {
            p_numofplain_t[i_plain_num]->F_exact_grid(i_tynum, i_tynum);
            if(i_tynum==1&&i_tynum==1)
            {
                p_numofplain_t[i_plain_num]->F_add_value(f_pho_weight, f_relative_lumin);
            }
        }
    }
    }
}
}
}

```

```

        i_photon_continue=p_numofplain_t[i_plain_num]->F_trans_polaraxis(f_direct_phovx,
        f_direct_phovy, f_direct_phovz, f_direct_phopx, f_direct_phopy,
        f_direct_phopz, i_reject_diffu);
    }
    else
    {
        p_numofplain_t[i_plain_num]->F_calc_yspace(f_direct_phovx, f_direct_phovy, f_direct_phovz,
        f_direct_phopx, f_direct_phopy, f_direct_phopz, 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_relative_lumin*f_phoweight;
        i_photon_continue=p_numofplain_t[i_plain_num]->F_trans_polaraxis(f_direct_phovx,
        f_direct_phovy, f_direct_phovz, f_direct_phopx, f_direct_phopy, f_direct_phopz,
        i_reject_diffu);
    }
    }
    break;
}
default: break;
}
}
else
{
    if(i_once>=1)i_photon_continue=-1;
    else
    {
        if(i_once_t==0)
        {
            f_i=f_i-1.;
            i_photon_continue=-1;
        }else
        {
            if(f_direct_phovz<0.)
            {
                f_i=f_i-1.;
                i_reject_diffu=1;
                i_photon_continue=-1;
            }else
            {
                void randomize();
                f_relative_lumin=o_light.F_get_diffu_lumin(f_direct_phovx, f_direct_phovy, f_direct_phovz,
                f_direct_phopx, f_direct_phopy, f_direct_phopz);
                if((f_relative_lumin-(int)f_relative_lumin)>=((double)rand())/32767.)
                {
                    f_relative_lumin=(int)f_relative_lumin+1.;
                }else
                {
                    f_relative_lumin=(int)f_relative_lumin;
                }
                if(f_relative_lumin>=1.)
                {
                    i_reject_diffu=0;
                    p_numofplain_r[0]->F_exact_grid(i_txnum, i_tynum);
                    if(i_txnum==1&&i_tynum==1)
                    {
                        p_numofplain_r[0]->F_add_value(f_phoweight, f_relative_lumin);
                    }
                    else
                    {
                        p_numofplain_r[0]->F_calc_origin_yspace(f_origin_phopx,f_origin_phopy, f_origin_phopz,
                        i_txnum, i_tynum);
                        *p_value_r[0][i_txnum][i_tynum]= *p_value_r[0][i_txnum][i_tynum]
                        +f_relative_lumin*f_phoweight;
                    }
                }

                f_direct_phopx=f_origin_phopx;
                f_direct_phopy=f_origin_phopy;
                f_direct_phopz=f_origin_phopz;
                f_direct_phovx=f_origin_phovx;
                f_direct_phovy=f_origin_phovy;
                f_direct_phovz=f_origin_phovz;
                i_photon_continue_d=1;
                i_first_plain_d=-1;
                if((p_numofplain_r[0]->F_get_reflec())<(((double)rand())/32767.))i_photon_continue_d=-1;
                while(i_photon_continue_d>0)
                {
                    f_dist_d=-1.;
                    i_plain_index_d=-1;
                    i_plain_num_d=-1;
                    f_dist_temp_d=-1.;
                    for(i_i=0;i_i<i_t;i_i++)
                    {
                        if(i_i!=0||i_first_plain_d!=-1)
                            f_dist_temp_d=p_numofplain_t[i_i]->F_calc_photon(f_direct_phovx, f_direct_phovy,
                            f_direct_phovz, f_direct_phopx, f_direct_phopy, f_direct_phopz);
                    }
                }
            }
        }
    }
}

```

```

if(f_dist_temp_d>0.)
{
  if ((f_dist_d<0.)||(f_dist_d>=f_dist_temp_d))
  {
    f_dist_d=f_dist_temp_d;
    i_plain_index_d=3;
    i_plain_num_d=i_i;
    i_first_plain_d=1;
  }
}
for(i_i=0;i_i<i_r;i_i++)
{
  if(i_i!=0||i_first_plain_d!=-1)
  f_dist_temp_d=p_numofplain_r[i_i]->F_calc_photon(f_dirac_phovx, f_dirac_phovy,
  f_dirac_phovz, f_dirac_phopx, f_dirac_phopy, f_dirac_phopz);
  if(f_dist_temp_d>0.)
  {
    if((f_dist_d<0.)||(f_dist_d>=f_dist_temp_d))
    {
      f_dist_d=f_dist_temp_d;
      i_plain_index_d=4;
      i_plain_num_d=i_i;
      i_first_plain_d=1;
    }
  }
  if(f_dist_d>0)
  {
    switch(i_plain_index_d)
    {
      case 4:
      {
        p_numofplain_r[i_plain_num_d]->F_exact_grid(i_tnum, i_tnum);
        if(i_tnum==1&&i_tnum==1)
        {
          p_numofplain_r[i_plain_num_d]->F_add_value(f_phoweight, f_relativelumin);
          i_photon_continue_d=p_numofplain_r[i_plain_num_d]->F_trans_polaraxis(f_dirac_phovx,
          f_dirac_phovy, f_dirac_phovz, f_dirac_phopx, f_dirac_phopy,
          f_dirac_phopz, i_reject_diffu);
        }
        else
        {
          p_numofplain_r[i_plain_num_d]->F_calc_xyspace(f_dirac_phovx,f_dirac_phovy,
          f_dirac_phovz, f_dirac_phopx,f_dirac_phopy,
          f_dirac_phopz,i_tnum,i_tnum);
          *p_value_r[i_plain_num_d][i_tnum][i_tnum]=
          *p_value_r[i_plain_num_d][i_tnum][i_tnum]
          +f_phoweight*f_relativelumin;
          i_photon_continue_d=p_numofplain_r[i_plain_num_d]->F_trans_polaraxis(f_dirac_phovx,
          f_dirac_phovy, f_dirac_phovz, f_dirac_phopx, f_dirac_phopy,
          f_dirac_phopz, i_reject_diffu);
        }
        break;
      }
      case 3:
      {
        p_numofplain_t[i_plain_num_d]->F_exact_grid(i_tnum, i_tnum);
        if(i_tnum==1&&i_tnum==1)
        {
          p_numofplain_t[i_plain_num_d]->F_add_value(f_phoweight, f_relativelumin);
          i_photon_continue_d=p_numofplain_t[i_plain_num_d]->F_trans_polaraxis(f_dirac_phovx,
          f_dirac_phovy, f_dirac_phovz, f_dirac_phopx, f_dirac_phopy,
          f_dirac_phopz, i_reject_diffu);
        }
        else
        {
          p_numofplain_t[i_plain_num_d]->F_calc_xyspace(f_dirac_phovx,f_dirac_phovy,
          f_dirac_phovz, f_dirac_phopx,f_dirac_phopy,f_dirac_phopz,i_tnum,i_tnum);
          *p_value_t[i_plain_num_d][i_tnum][i_tnum]=
          *p_value_t[i_plain_num_d][i_tnum][i_tnum]
          +f_phoweight*f_relativelumin;
          i_photon_continue_d=p_numofplain_t[i_plain_num_d]->F_trans_polaraxis(f_dirac_phovx,
          f_dirac_phovy, f_dirac_phovz, f_dirac_phopx, f_dirac_phopy,
          f_dirac_phopz, i_reject_diffu);
        }
        break;
      }
      default: break;
    }
  }
  else
  {
    i_photon_continue_d=-1;
  }
}

```

```

    }
    }
    }
    else
    {
        i_reject_diffu=1;
    }
    }
    i_diffu=1;
    i_once=1;
    if(f_relative_lumin==1.) f_ratio=0.;
    if(f_relative_lumin>1.)
    {
        f_pho_num=f_pho_num+(1.-f_relative_lumin)*f_ratio;
        f_ratio=0.;
    }
    i_photon_continue=-1;
}
}
}
}
}
o_out<<"\n\n\n"<<"Skylight Ekh = "<<f_Ekh<<"\n";
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]*i_txnum*i_tynum/(f_area*f_Ekh))<<"\n";
            }
        }
    }
}
for(i_i=0;i_i<i_t;i_i++)
{
    p_numofplain_t[i_i]->F_xynum(i_txnum, i_tynum);
    if(i_txnum==1&&i_tynum==1)
    {
        f_area=p_numofplain_t[i_i]->F_nogrid_get_area();
        o_out<<"triangle plain["<<i_i<<"] = "<<(p_numofplain_t[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++)
            {
                f_area=p_numofplain_t[i_i]->F_grid_get_area(i_ij, i_ij2);
                o_out<<"triangle plain["<<i_i<<"["<<i_ij<<"["<<i_ij2<<"] =
                    "<<(*p_value_t[i_i][i_ij][i_ij2]/f_area)<<" lx\n";
            }
        }
    }
}
}
}
o_in.close();
o_out.close();
}

```