

POZAS MONTOYA KARLA JOSSAVET
PROYECTO

“EFECTO COMPTON”

OBJETIVO:

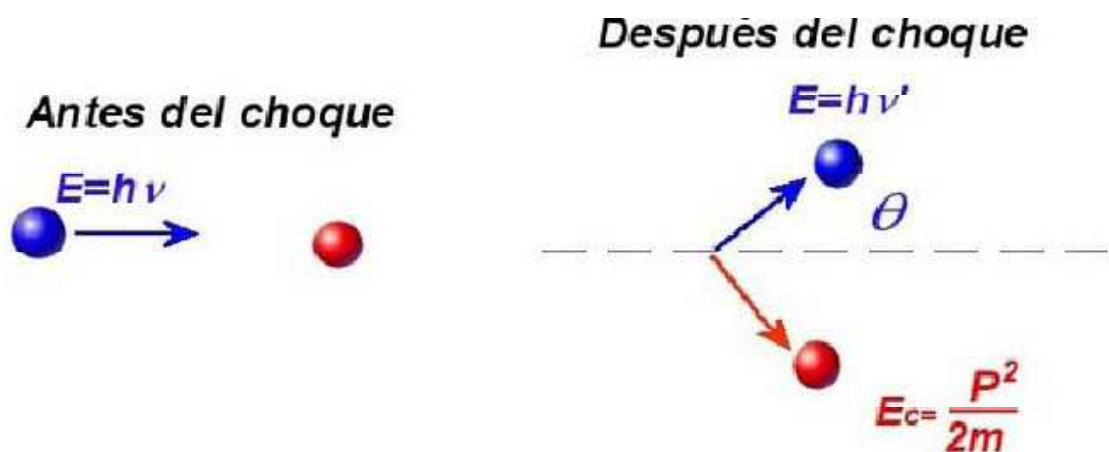
QUE EL ALUMNO PUEDA IMPLEMENTAR TODOS LOS CONOCIMIENTOS APRENDIDOS DURANTE EL CURSO PARA PODER REALIZAR UNA ANIMACIÓN QUE CONLLEVE TODO LO VISTO DURANTE EL CURSO.

OBJETIVO 1:

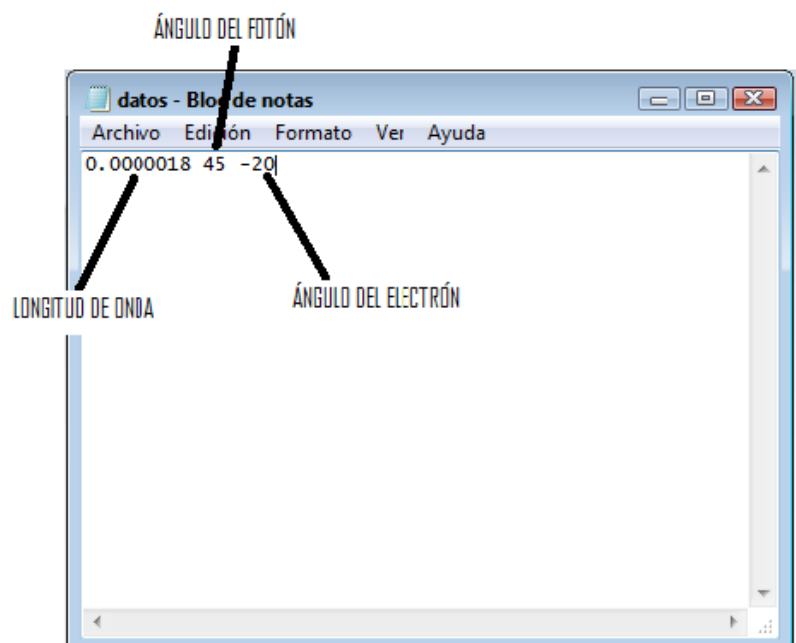
ACOPLAR ECUACIONES CON VISUALIZACIÓN CON OPENGL

INTRODUCCION:

EL EFECTO COMPTON ES LA DESVIACIÓN EN MOVIMIENTO DEL GOLPE DE UN FOTÓN (ROJO) CONTRA UN ELECTRÓN, DONDE AMBOS SALEN DISPARADOS EN DIFERENTES DIRECCIONES. VISUALMENTE SE VERA ASÍ:

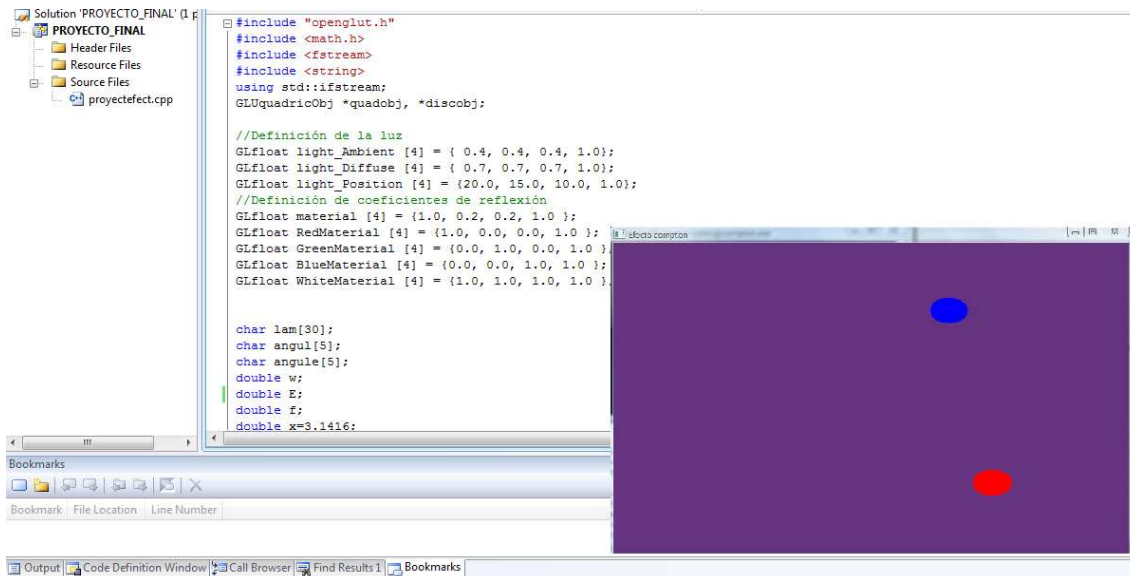


ENSEGUIDA MUESTRO EL ARCHIVO datos.txt EN EL CUAL SE REALIZÓ EL PROYECTO, CABE MENCIONAR QUE DESPUES DE VARIAS PRUEBAS, PUDE COMPROBAR QUE MIENTRAS MAS GRANDE ERA EL VALOR DE LONGITUD DE ONDA LA VELOCIDAD DEL FOTON ERA MUY LENTA. EL ORDEN DE LOS DATOS EN EL ARCHIVO ES EL SIGUIENTE:

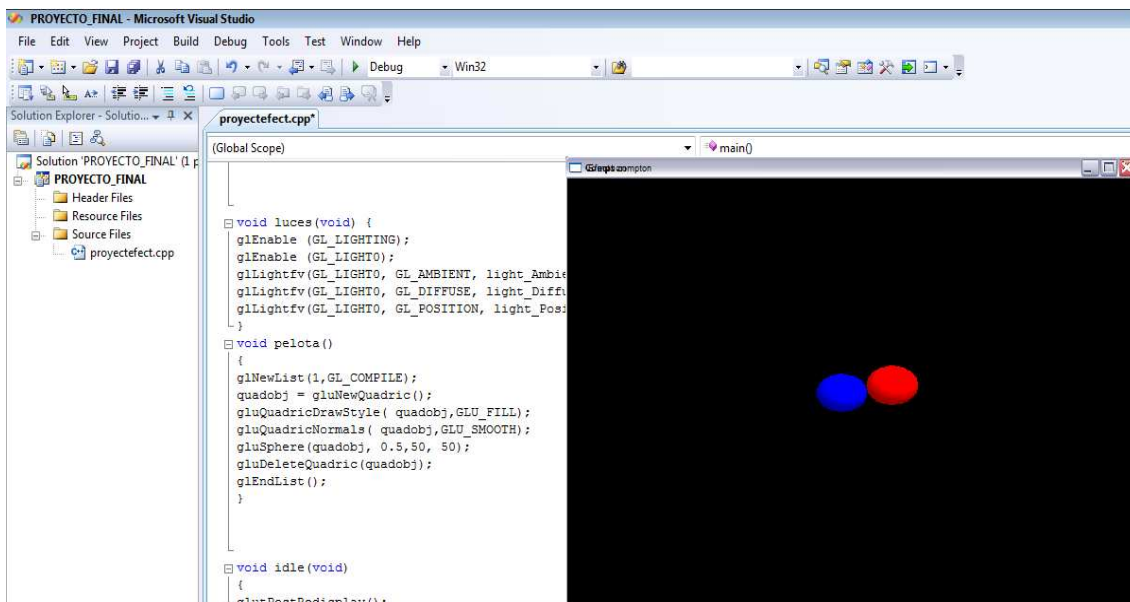


POZAS MONTOYA KARLA JOSSAVET
PROYECTO
DESARROLLO:

DESPUES DE CONSULTAR LAS ECUACIONES Y HACER LOS CALCULOS NECESARIOS, SE PROCEDE A REALIZAR EL PROGRAMA Y VERIFICAR SU FUNCIONAMIENTO, MOSTRANDO VISUALMENTE EL EFECTO COMPTON, ANTES DURANTE Y DESPUES DEL CHOQUE.

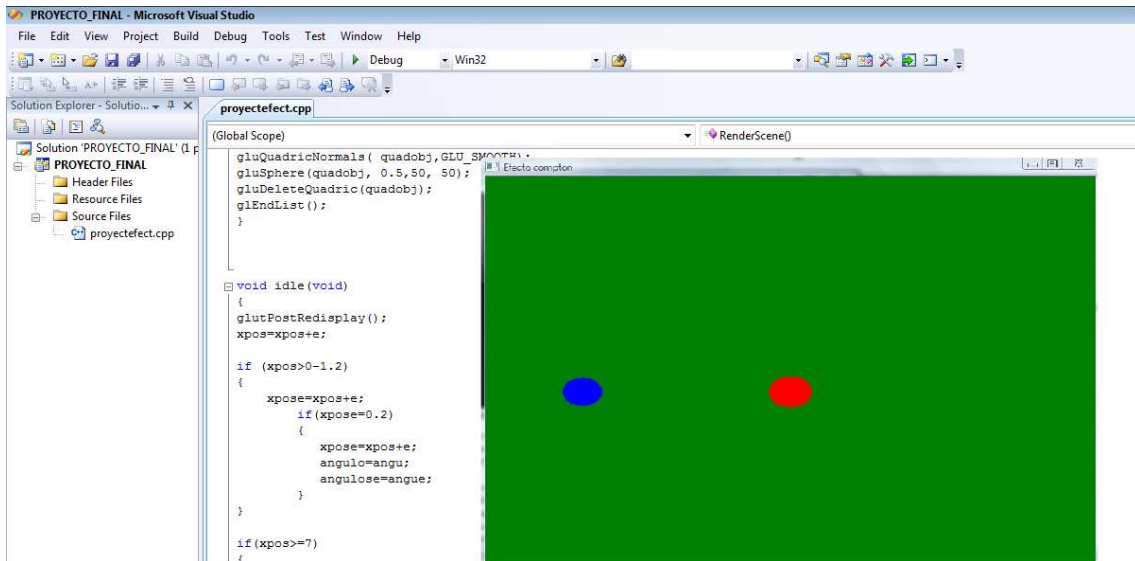


ANTES DE CHOCAR LAS PARTICULAS



SE DA EL CHOQUE ENTRE ELECTRÓN Y FOTÓN

POZAS MONTOYA KARLA JOSSAVET
PROYECTO



SE DISPERSAN LAS PARTICULAS DESPUÉS DEL CHOQUE

CODIGO:

```
#include "openglut.h"
#include <math.h>
#include <fstream>
#include <string>
using std::ifstream;
GLUquadricObj *quadobj, *discobj;

//Definición de la luz
GLfloat light_Ambient [4] = { 0.4, 0.4, 0.4, 1.0};
GLfloat light_Diffuse [4] = { 0.7, 0.7, 0.7, 1.0};
GLfloat light_Position [4] = {20.0, 15.0, 10.0, 1.0};
//Definición de coeficientes de reflexión
GLfloat material [4] = {1.0, 0.2, 0.2, 1.0 };
GLfloat RedMaterial [4] = {1.0, 0.0, 0.0, 1.0 };
GLfloat GreenMaterial [4] = {0.0, 1.0, 0.0, 1.0 };
GLfloat BlueMaterial [4] = {0.0, 0.0, 1.0, 1.0 };
GLfloat WhiteMaterial [4] = {1.0, 1.0, 1.0, 1.0 };

//Declaracion de variables necesarias
char lam[30];
char angul[5];
char angule[5];
double w;
double E;
double f;
double x=3.1416;
double xpos=-8;
double xpose;
double h=6.36E-34;
double la;
double angulo;
double angu;
```

POZAS MONTOYA KARLA JOSSAVET

PROYECTO

```
double xnueva;  
double anguloe;  
double angue;  
double t=1E-30;
```

```
//Cargando las ecuaciones de luz  
void luces(void) {  
glEnable (GL_LIGHTING);  
glEnable (GL_LIGHT0);  
glLightfv(GL_LIGHT0, GL_AMBIENT, light_Ambient );  
glLightfv(GL_LIGHT0, GL_DIFFUSE, light_Diffuse );  
glLightfv(GL_LIGHT0, GL_POSITION, light_Position );  
}  
void pelota()  
{  
glNewList(1, GL_COMPILE);  
quadobj = gluNewQuadric();  
gluQuadricDrawStyle( quadobj, GLU_FILL);  
gluQuadricNormals( quadobj, GLU_SMOOTH);  
gluSphere(quadobj, 0.5, 50, 50);  
gluDeleteQuadric(quadobj);  
glEndList();  
}  
  
void idle(void)  
{  
glutPostRedisplay();  
xpos=xpos+e;  
  
if (xpos>0-1.2)  
{  
    xpose=xpos+e;  
    if(xpose=0.2)  
    {  
        xpose=xpos+e;  
        angulo=angu;  
        angulose=angue;  
    }  
}  
  
if(xpos>=7)  
{  
    angulo=0;  
    xpos=-8;  
    anguloe=0;  
    xpose=0;  
}  
}  
  
//Función que lee los datos que ingresemos en el archivo datos.txt  
void LeerArchivo()  
{  
  
std:: ifstream texto("datos.txt");  
if (texto.is_open{})  
{  
    texto>>lam; // longitud de onda  
    texto>>angul; // ángulo del fotón  
    texto>>angule; // ángulo del electrón
```

POZAS MONTOYA KARLA JOSSAVET

PROYECTO

```
    texto.close();//cerramos el archivo

    la=atof(lam);
    angu=atof(angul);
    angue=atof(angule);
    f=(1/(la*t));
    w=2*x*f;//frecuencia
    E=h*w;//energia
}
}

void RenderScene(void)
{
    glClear(GL_COLOR_BUFFER_BIT|GL_DEPTH_BUFFER_BIT);
    LeerArchivo();
    glMaterialfv(GL_FRONT, GL_AMBIENT_AND_DIFFUSE, GreenMaterial );
    glMatrixMode(GL_MODELVIEW);
    glPushMatrix();
    glRotatef(angul,0.0,0.0,1.0);
    glTranslatef(0,0,0);
    glCallList(1);
    glPopMatrix();
    glMaterialfv(GL_FRONT, GL_AMBIENT_AND_DIFFUSE, RedMaterial );
    glMatrixMode(GL_MODELVIEW);
    glPushMatrix();
    glRotatef(angule,0.0,0.0,1.0);
    glTranslatef(0,0,0);
    glCallList(1);
    glPopMatrix();
    glFlush();
    glutSwapBuffers();
}
int main(void)
{
    int IdeWindow;
    glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGB | GLUT_DEPTH);
    glutInitWindowSize(800,600);
    glutInitWindowPosition(300,100);
    IdeWindow=glutCreateWindow("Efecto compton");
    glutDisplayFunc(RenderScene);
    glutIdleFunc( idle );
    /*
    glClearColor(0.60f, 0.40f, 0.70f, 1.0f);//color púrpura
    glClearColor(0.0f, 0.0f, 0.0f, 0.0f); //color negro
    */
    glClearColor(0.60f, 0.8f, 0.2f, 1.0f);//color verde
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    gluPerspective(15,800/600,.1,1000);
    glMatrixMode(GL_MODELVIEW);
    glLoadIdentity();
    gluLookAt(45,0,45, 0,0,0,5,5,-5);
    luces();
    pelota();
    LeerArchivo();
    glutMainLoop();
    glutDestroyWindow(IdeWindow);
    return 0;
}
```