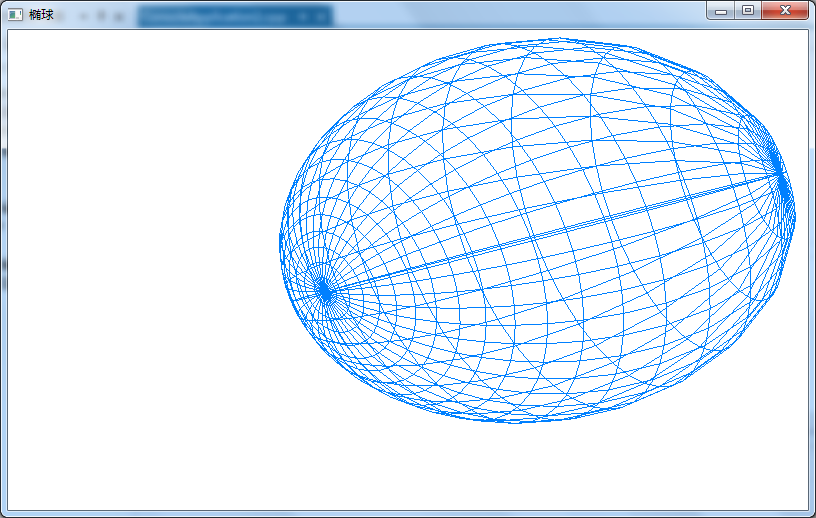
计算机图形学第二次作业

李博杰 PB10000603



#include "stdafx.h"

#include <gl/glut.h>

#include <math.h>

#define pi 3.1415926

typedef struct Point3f

{

GLfloat x;

GLfloat y;

GLfloat z;

} point;

void setPoint2(GLfloat rx, GLfloat ry, GLfloat rz, GLfloat a, GLfloat b, point\* p)

{

p->x = rx\*sin(a\*pi/180.0)\*cos(b\*pi/180.0);

p->y = ry\*sin(a\*pi/180.0)\*sin(b\*pi/180.0);

p->z = rz\*cos(a\*pi/180.0);

}

point\* getPointMatrix2(GLfloat rx, GLfloat ry, GLfloat rz, GLint slices)

{

int i,j,w=2\*slices,h=slices;

float a=0.0,b=0.0;

float hStep=180.0/(h-1);

float wStep=360.0/w;

point \*matrix = (point \*)malloc(w\*h\*sizeof(point));

if (!matrix)

return NULL;

for(a=0.0,i=0; i<h; i++,a+=hStep)

for(b=0.0,j=0; j<w; j++,b+=wStep)

setPoint2(rx, ry, rz, a, b, &matrix[i\*w+j]);

return matrix;

}

void drawSlice(point &p1, point &p2, point &p3, point &p4)

{

glBegin(GL\_LINE\_LOOP);

glVertex3f(p1.x,p1.y,p1.z);

glVertex3f(p2.x,p2.y,p2.z);

glVertex3f(p3.x,p3.y,p3.z);

glVertex3f(p4.x,p4.y,p4.z);

glEnd();

}

bool drawOval(GLfloat rx,GLfloat ry,GLfloat rz,GLint slices)

{

int i,j;

int w=2\*slices, h=slices;

point \*mx = getPointMatrix2(rx,ry,rz,slices);

if(!mx)

return false;

for(i=0; i<h; i++)

for(j=0; j<w; j++)

drawSlice(

mx[i\*w + j],

mx[i\*w + (j+1)%w],

mx[((i+1)%h)\*w + (j+1)%w],

mx[((i+1)%h)\*w + j]

);

free(mx);

return true;

}

void drawOval() {

glClear(GL\_COLOR\_BUFFER\_BIT | GL\_DEPTH\_BUFFER\_BIT);

glColor3f(0.0, 0.5, 1.0);

glMatrixMode(GL\_MODELVIEW);

glLoadIdentity();

glTranslatef(0.0, 0.0, -25.0);

glRotatef(40.0, 0, 1, 1);

drawOval(6,9,18,20);

glFlush();

}

void initGL(GLfloat width, GLfloat height )

{

glShadeModel(GL\_SMOOTH);

glClearColor(1.0, 1.0, 1.0, 0.0);

glMatrixMode(GL\_PROJECTION);

glViewport(0, 0, width, height);

gluPerspective(60.0, width/height, 1.0, 100.0);

}

int \_tmain(int argc, \_TCHAR\* argv[])

{

glutInit(&argc, (char \*\*)argv);

glutInitDisplayMode(GLUT\_RGB | GLUT\_SINGLE);

glutInitWindowPosition(50, 100);

glutInitWindowSize(800, 480);

glutCreateWindow("椭球");

initGL(800,480);

glutDisplayFunc(drawOval);

glutMainLoop();

return 0;

}