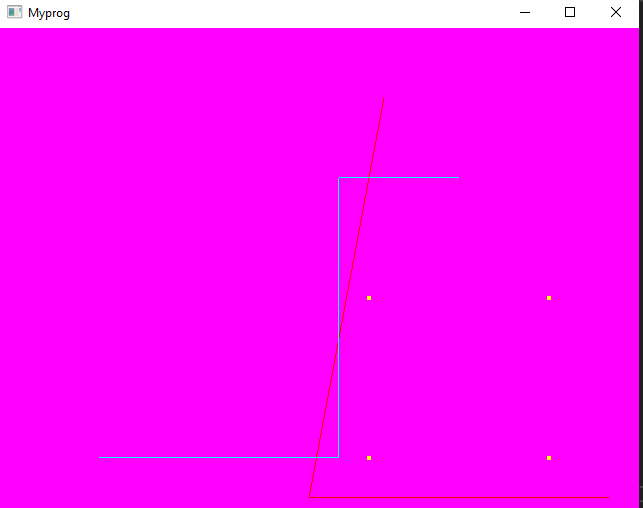
Лабораторная работа №1  
4210БеловВА

1.1  


Листинг1.1  
#include <GL/glut.h>

void init();

void draw();

void main(int argc, char \*\*argv) //Главная функция

{

glutInit(&argc, argv);

glutInitDisplayMode(GLUT\_SINGLE | GLUT\_RGB);

glutInitWindowSize(640, 480);

glutInitWindowPosition(50, 50);

glutCreateWindow("Myprog");

init();

glutDisplayFunc(draw);

glutMainLoop();

}

void init() //Функция инициализации

{

glClearColor(1.0, 0.0, 1.0, 1.0);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

gluOrtho2D(0.0, 2.0, 0.0, 1.0);

}

void draw()

{ //Функция рисования

glClear(GL\_COLOR\_BUFFER\_BIT);

glColor3f(1.0, 0.0, 0.0);

glViewport(10, 10, 600, 400);

glBegin(GL\_LINE\_LOOP);

glVertex2f(3.0, 0.0); glVertex2f(5.0, 0.0);

glVertex2f(1.0, 0.0); glVertex2f(3.0, 8.0);

glEnd();

glColor3f(0.0, 1.0, 1.0);

glBegin(GL\_LINE\_STRIP);

glVertex2f(0.3, 0.1); glVertex2f(1.1, 0.1);

glVertex2f(1.1, 0.8); glVertex2f(1.5, 0.8);

glEnd();

glColor3f(1.0, 1.0, 0.0);

glPointSize(4);

glBegin(GL\_POINTS);

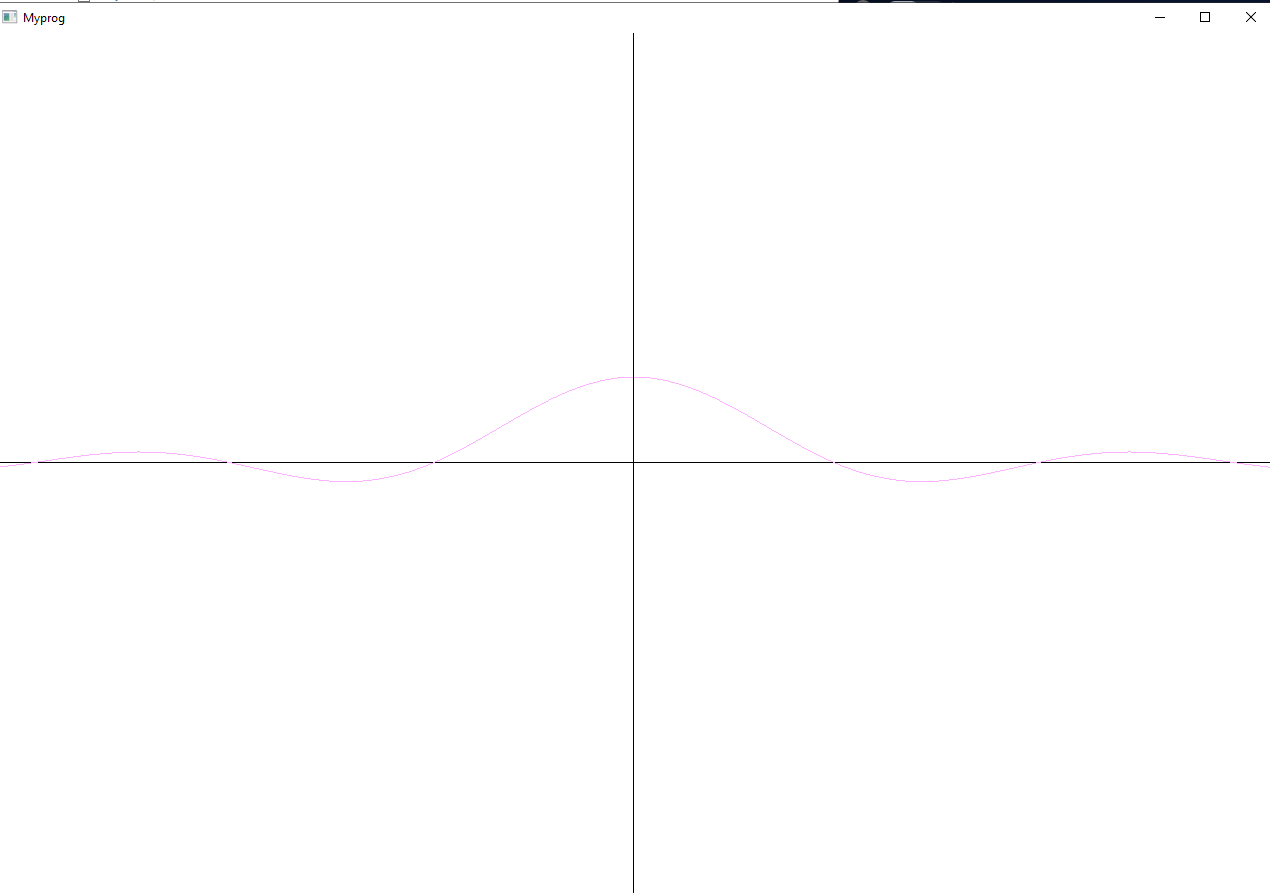
glVertex2f(1.2, 0.5); glVertex2f(1.8, 0.5);

glVertex2f(1.2, 0.1); glVertex2f(1.8, 0.1);

glEnd();

glFlush();

}

1.2  
  
Листинг 1.2  
#include <GL/glut.h>

#include <math.h>

#include <iostream>

void init();

void draw();

float mass[41];

float mass2[41];

float w;

void main(int argc, char \*\*argv) //Главная функция

{

std::cout << "y(x) = sin(wx) /wx" << '\n';

std::cout << "Vvedite parametr w: ";

std::cin >> w;

double x = -10;

for (int i = 0; i < 41;i++) {

if (x == 0) {

mass[i] = sin(w\*(x - 0.1)) / (w\*(x - 0.1));

continue;

}

mass[i] = sin(w\*x) / (w\*x);

std::cout << x << ' ' << mass[i] << '\n';

x += 0.25;

}

x = 0;

for (int i = 0; i < 41;i++) {

if (x == 0) {

mass2[i] = sin(w\*(x + 0.1)) / (w\*(x + 0.1));

x += 0.25;

continue;

}

mass2[i] = sin(w\*x) / (w\*x);

std::cout << x << ' ' << mass2[i] << '\n';

x += 0.25;

}

glutInit(&argc, argv);

glutInitDisplayMode(GLUT\_SINGLE | GLUT\_RGB);

glutInitWindowSize(1280, 860);

glutInitWindowPosition(50, 50);

glutCreateWindow("Myprog");

init();

glutDisplayFunc(draw);

glutMainLoop();

}

void init() //Функция инициализации

{

glClearColor(1.0, 1.0, 1.0, 1.0);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

gluOrtho2D(-5.0, 5.0, -5.0, 5.0);

}

void draw()

{ //Функция рисования

// 1) y(x) = sin(wx) /wx, где w – const;

glClear(GL\_COLOR\_BUFFER\_BIT);

glColor3f(0.0, 0.0, 0.0);

glBegin(GL\_LINE\_STRIP); //ОСИ

glVertex2f(-5.0, 0.0); glVertex2f(5.0, 0.0);

glEnd();

glBegin(GL\_LINE\_STRIP);

glVertex2f(0.0, -5.0); glVertex2f(0.0, 5.0);

glEnd();

glColor3f(1.0, 0.7, 1.0);

glBegin(GL\_LINE\_STRIP);

double x = -10;

for (int i = 0; i < 41;i ++) {

glVertex2f(x/2, mass[i]);

x += 0.25;

}

glEnd();

glBegin(GL\_LINE\_STRIP);

x = 0;

for (int i = 0; i < 41;i++) {

glVertex2f(x / 2, mass2[i]);

x += 0.25;

}

glEnd();

glColor3f(0.0, 0.0, 0.0);

glBegin(GL\_LINE\_STRIP);// выколотая точка

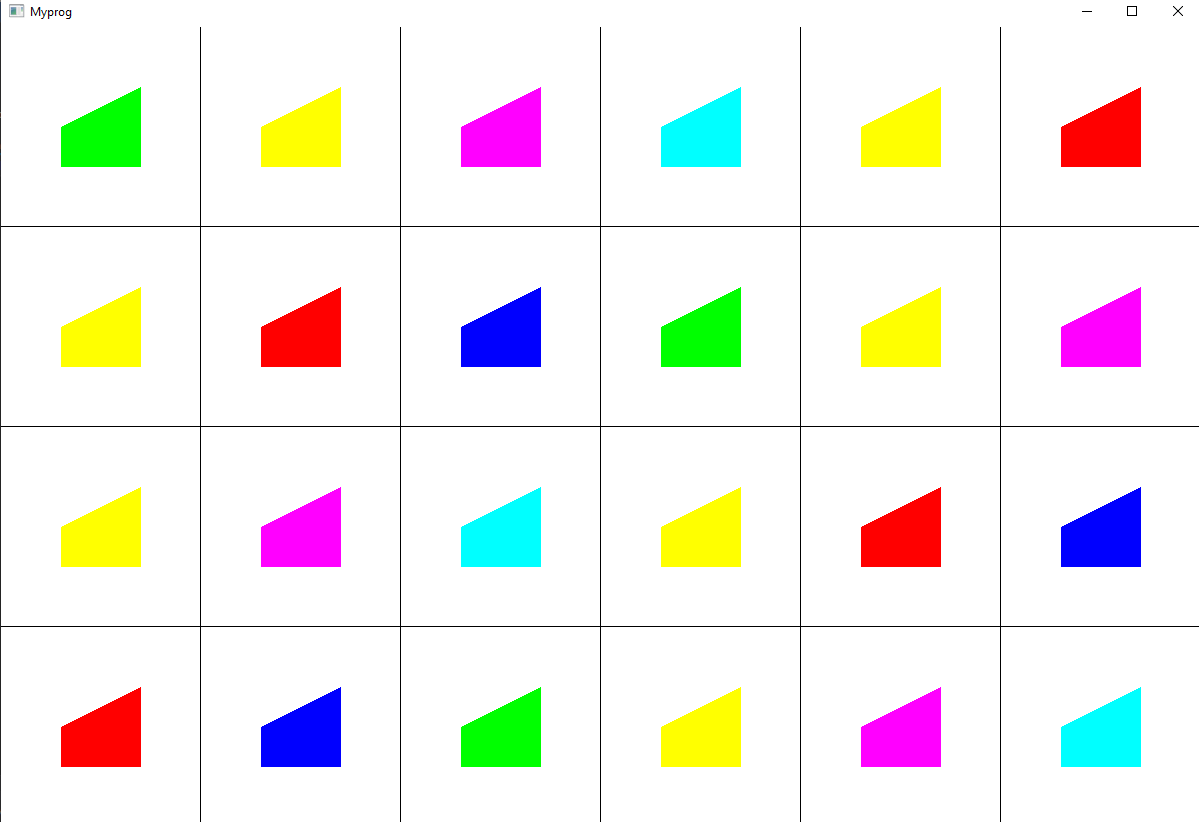
glVertex2f(0.0, -5.0); glVertex2f(0.0, 5.0);

glEnd();

glFlush();

}

2.

  
#include <GL/glut.h>

#include <math.h>

#include <iostream>

void init();

void draw();

float mass[21] = { 255,0,0,255,69,0,255,255,0,0,255,0,0,255,255,0,0,255,128,0,128};

void main(int argc, char \*\*argv) //Главная функция

{

glutInit(&argc, argv);

glutInitDisplayMode(GLUT\_SINGLE | GLUT\_RGB);

glutInitWindowSize(1200, 800);

glutInitWindowPosition(50, 50);

glutCreateWindow("Myprog");

init();

glutDisplayFunc(draw);

glutMainLoop();

}

void init() //Функция инициализации

{

glClearColor(1.0, 1.0, 1.0, 1.0);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

gluOrtho2D(-5.0, 5.0, -5.0, 5.0);

}

void draw()

{

GLfloat l = -1, r = 1, b = -1, t = 1;

GLint x = 0, y = 0, w = 200, h = 200;

int q = 0, qw = 1, e = 2;

glClear(GL\_COLOR\_BUFFER\_BIT);

gluOrtho2D(l, r, b, t);

for (x = 0; x <= 1200; x += w)

for (y = 0; y <= 800; y += h)

{

glViewport(x, y, w, h);

glColor3f(0.0, 0.0, 0.0);

glBegin(GL\_LINE\_LOOP);

glVertex2f(-5.0, -5.0);glVertex2f(-5.0, 5.0);

glVertex2f(5.0, 5.0);glVertex2f(5.0, -5.0);

glEnd();

glColor3f(mass[q], mass[qw], mass[e]);

glBegin(GL\_POLYGON);

glVertex2f(-2.0, -2.0);glVertex2f(-2.0, 0.0);glVertex2f(2.0,2.0);glVertex2f(2.0, 0.0);glVertex2f(2.0, -2.0);

glEnd();

q += 3; qw += 3; e += 3;

if (e > 21) { q = 0; qw = 1; e = 2; }

}

glFlush();

}