**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное бюджетное образовательное учреждение высшего**

**профессионального образования**

**«Казанский национальный исследовательский технический университет**

**им. А.Н. Туполева-КАИ»**

Институт компьютерных технологий и защиты информации

Лабораторная работа № 1

по дисциплине «Компьютерная графика»

**Выполнил**

студент группы 4211

Абдуллаев Л.Х.

Казань

2020

# Задание 1

Цель :

Создать программу, рисующую на экране графики функций синуса и косинуса.

Листинг:

#include "pch.h"

#include <iostream>

#include <GL/glut.h>

#include <math.h>

//задание 1

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(0.9, 0.9, 0.9, 0.0);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

gluOrtho2D(-8.0, 8.0, -2.0, 2.0);

}

void draw() //Функция рисования

{

GLfloat PI = 3.14, x, y, a = 0.5, b = 1;

glClear(GL\_COLOR\_BUFFER\_BIT);

glColor3f(1.0, 0.0, 0.0);

glViewport(10, 10, 600, 400);

glBegin(GL\_LINE\_STRIP);

for (x = -2 \* PI - 2; x <= 2 \* PI + 2; x += PI / 18)

{

y = sin(a\*x);

glVertex2f(x, y);

}

glEnd();

glColor3f(0.5, 0.3, 0.8);

glBegin(GL\_LINE\_STRIP);

for (x = -2 \* PI - 2; x <= 2 \* PI + 2; x += PI / 18)

{

y = cos(b\*x);

glVertex2f(x, y);

}

glEnd();

glColor3f(0.0, 0.0, 1.0);

glBegin(GL\_LINES);

glVertex2f(-8.0, 0.0); glVertex2f(8.0, 0.0);

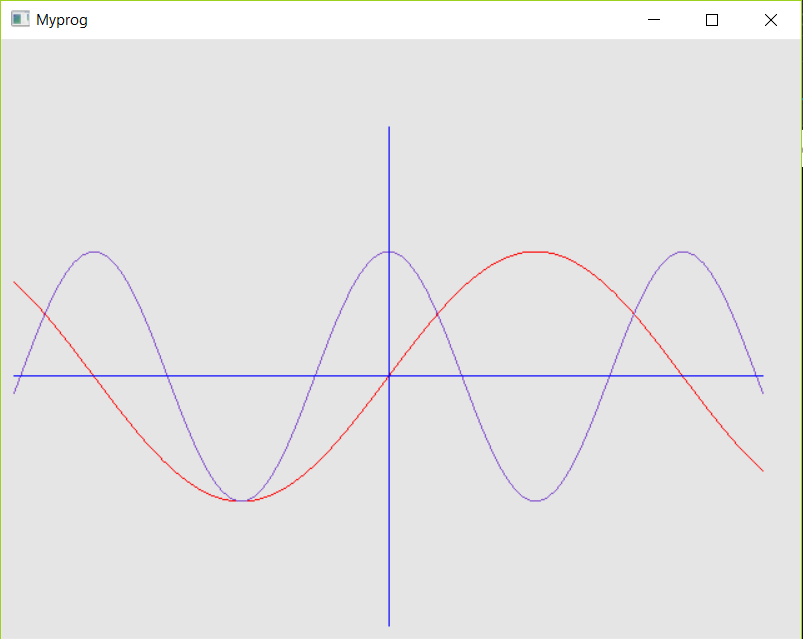
glVertex2f(0.0, 2.0); glVertex2f(0.0, -2.0);

glEnd();

glFlush();

}

Результат работы:



# Задание 2

Цель :

Создать программу, которая покрывает окно мозаикой.

Листинг:

#include "pch.h"

#include <iostream>

#include <GL/glut.h>

#include <math.h>

void init();

void draw();

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

{

glutInit(&argc, argv);

glutInitDisplayMode(GLUT\_SINGLE | GLUT\_RGB);

glutInitWindowSize(660, 480);

glutInitWindowPosition(50, 50);

glutCreateWindow("Myprog");

init();

glutDisplayFunc(draw);

glutMainLoop();

}

void init() //Функция инициализации

{

glClearColor(0.9, 0.9, 0.9, 0.0);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

gluOrtho2D(-100, 100, -100, 100);

}

void draw()

{

GLfloat m = 60;

GLint x = 0, y = 0;

glClear(GL\_COLOR\_BUFFER\_BIT);

for (x = 0; x <= 600; x += m)

for (y = 0; y <= 480; y += m)

{

glViewport(x, y, m, m);

glColor3f(1.0, 0.0, 0.0);

glBegin(GL\_POLYGON);

glVertex2f(-m , m);

glColor3f(1.0, 1.0, 0.0);

glVertex2f(-m,-m);

glColor3f(1.0, 0.0, 0.0);

glVertex2f(m, m);

glColor3f(1.0, 1.0, 0.0);

glVertex2f(m, -m);

glEnd();

}

glFlush();

}

Результат работы:

