НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

**Кафедра системного програмування і спеціалізованих комп’ютерних систем**

**Лабораторна робота №5**

з дисципліни

**«Програмування»**

Тема: **«Побудова графіків функцій»**

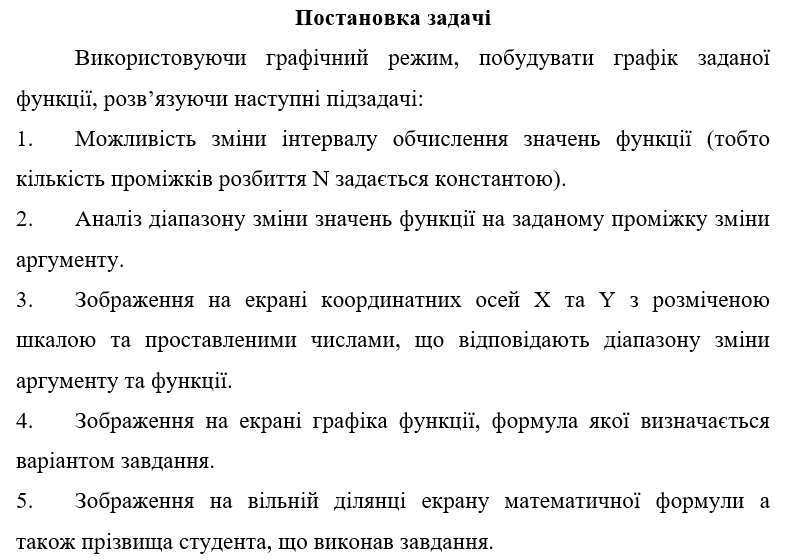
Виконав: студент І курсу

ФПМ групи КВ-61

Бідяк М.А.

Перевірила: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Київ – 2017



**Завдання варіанту №2**



**Текст програми:**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <SFML/Graphics.hpp>

#include <stdlib.h>

#include <math.h>

#include<iostream>

#include <string.h>

using namespace sf;

int main()

{

int a = 2;

ContextSettings settings;

settings.antialiasingLevel = 8;

RenderWindow window(VideoMode(1280, 720), "lab-5", Style::Default, settings);

const int xmax = 10, xmin = -10, ymax = 10, ymin = -10, xbeg = 290, xend = 990, ybeg = 10, yend = 710;

float mx, my;

int i;

const int n = 1;

int p = 300, x0 = 640, y0 = 360;

mx = (xend - xbeg) / (xmax - xmin);

my = (yend - ybeg) / (ymax - ymin);

double fi, dfi = 0.0001;

const float k = round(20 / dfi);

RectangleShape rectangle(sf::Vector2f(1000, 1));

rectangle.setPosition(x0 - 500, y0);

rectangle.setFillColor(Color::Blue);

RectangleShape rectangle1(sf::Vector2f(1, 700));

rectangle1.setPosition(x0, y0 - 350);

rectangle1.setFillColor(Color::Blue);

RectangleShape rectangleArr1\_1(sf::Vector2f(1, 30));

rectangleArr1\_1.setPosition(x0, 10);

rectangleArr1\_1.rotate(20);

rectangleArr1\_1.setFillColor(Color::Blue);

RectangleShape rectangleArr1\_2(sf::Vector2f(1, 30));

rectangleArr1\_2.setPosition(x0, 10);

rectangleArr1\_2.rotate(-20);

rectangleArr1\_2.setFillColor(Color::Blue);

RectangleShape rectangleArr2\_1(sf::Vector2f(1, 30));

rectangleArr2\_1.setPosition(1140, y0);

rectangleArr2\_1.rotate(110);

rectangleArr2\_1.setFillColor(Color::Blue);

RectangleShape rectangleArr2\_2(sf::Vector2f(1, 30));

rectangleArr2\_2.setPosition(1140, y0);

rectangleArr2\_2.rotate(70);

rectangleArr2\_2.setFillColor(Color::Blue);

Font font;

font.loadFromFile("AdonisC\_Bold\_Italic.otf");

Text text, text1, textMadeBy,formula;

text.setFont(font);

text.setCharacterSize(14);

text.setFillColor(Color::Yellow);

text.setPosition(x0 + mx, y0);

text1.setFont(font);

text1.setCharacterSize(14);

text1.setFillColor(Color::Yellow);

text1.setPosition(mx, y0 + my);

textMadeBy.setFont(font);

textMadeBy.setCharacterSize(30);

textMadeBy.setFillColor(Color::Yellow);

textMadeBy.setPosition(950, 150);

textMadeBy.setString("Bidyak M.A");

formula.setFont(font);

formula.setCharacterSize(30);

formula.setFillColor(Color::Yellow);

formula.setPosition(750, 180);

formula.setString("y = (4 \* a\*a\*a) / (x1\*x1 + 4 \* a\*a)");

const int num = k\*n;

double \*x, \*y, \*r;

x = (double\*)calloc(num, sizeof(double));

y = (double\*)calloc(num, sizeof(double));

r = (double\*)calloc(num, sizeof(double));

double x1 = -10; i = 0;

while (x1<10)

{

x[i] = x1;

y[i] = (4 \* a\*a\*a) / (x1\*x1 + 4 \* a\*a);

i++;

x1 += dfi;

}

VertexArray pointmap(sf::Points, num);

for (register int i = 0; i <num; i++) {

pointmap[i].position = sf::Vector2f((x0 + round(x[i] \* mx)), (y0 + round(y[i] \* my)));

pointmap[i].color = sf::Color::Cyan;

}

while (window.isOpen())

{

Event event;

while (window.pollEvent(event))

{

if (event.type == Event::Closed)

{

window.close();

}

}

window.clear();

window.draw(pointmap);

window.draw(rectangle);

window.draw(rectangle1);

window.draw(rectangleArr1\_1);

window.draw(rectangleArr1\_2);

window.draw(rectangleArr2\_1);

window.draw(rectangleArr2\_2);

for (int j = xmin; j <= xmax; j++)

{

char buff1[5];

\_itoa(j, buff1, 10);

text.setString(buff1);

text.setPosition(x0 + j\*mx, y0);

window.draw(text);

}

for (int j = ymin; j <= ymax; j++)

{

char buff1[5];

\_itoa(-j, buff1, 10);

text1.setString(buff1);

text1.setPosition(x0, y0 + j\*my);

window.draw(text1);

}

window.draw(textMadeBy);

window.draw(formula);

window.display();

}

free(x);

free(y);

free(r);

return 0;

}

**Тестування програми:**

