Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования

<<Вятский государственный университет>> Колледж ВятГУ

**ОТЧЕТ**

**ПО ДОМАШНЕЙ КОНТРОЛЬНОЙ РАБОТЕ №5**

**<<Построить график функции в паскаль>>**

**ПО ДИСЦИПЛИНЕ <<Основы алгоритмизации и программирования>>**

Выполнил: студент учебной группы

ИСПк-101-51-00

Шертосбитов Михаил Романович

Преподаватель: Сергеева Елизавета Григорьевна

Киров

2024

Цель работы: Получить базовые навыки работы с одномерными массивами освоить принципы работы со строками как с частным случаем одномерных массивов

Задание

Программа AreaUnderCurveVisualization предназначена для вычисления площади под кривой, заданной функцией, и визуализации этого процесса. Она использует метод левых прямоугольников для численного интегрирования и графическую библиотеку GraphABC для отображения графиков и результатов.

program AreaUnderCurveVisualization;

uses

GraphABC;

type

TFunction = function(x: Double): Double;

function f(x: Double): Double;

begin

Result := x \* x \* x + x \* x + x + 1; // Определение функции

end;

function LeftRectangles(a, b: Double; n: Integer; func: TFunction): Double;

var

h, x: Double;

i: Integer;

begin

h := (b - a) / n;

Result := 0;

for i := 0 to n - 1 do

begin

x := a + i \* h;

Result := Result + func(x);

end;

Result := Result \* h; // Умножаем на ширину отрезка

end;

procedure DrawAxes;

begin

Line(0, WindowHeight div 2, WindowWidth, WindowHeight div 2); // Ось X

Line(WindowWidth div 2, 0, WindowWidth div 2, WindowHeight); // Ось Y

end;

procedure DrawFunction(func: TFunction);

var

xScreen, yScreen: Integer;

xReal: Real;

begin

for xScreen := 0 to WindowWidth do

begin

xReal := (xScreen - WindowWidth div 2) / 20; // Масштабирование по оси X

yScreen := WindowHeight div 2 - Round(func(xReal) \* 20); // Масштабирование по оси Y

if (yScreen >= 0) and (yScreen <= WindowHeight) then

SetPixel(xScreen, yScreen, clWhite);

end;

end;

procedure ShadingArea(a, b: Real; n: Integer; func: TFunction);

var

xScreen, yScreen: Integer;

xReal, h: Real;

begin

h := (b - a) / n;

for xScreen := Round(a \* 20 + WindowWidth div 2) to Round(b \* 20 + WindowWidth div 2) do

begin

xReal := (xScreen - WindowWidth div 2) / 20;

yScreen := WindowHeight div 2 - Round(func(xReal) \* 20);

Line(xScreen, WindowHeight div 2, xScreen, yScreen);

end;

end;

procedure VisualizeIntegralCalculation(a, b: Real; n: Integer; func: TFunction);

var

i, xScreen: Integer;

xReal, h: Real;

begin

h := (b - a) / n;

SetBrushColor(clRed);

for i := 0 to n - 1 do

begin

xReal := a + i \* h;

xScreen := Round((xReal - a) \* 20) + WindowWidth div 2;

Rectangle(xScreen, WindowHeight div 2, xScreen + Round(h \* 20), WindowHeight div 2 - Round(func(xReal) \* 20));

end;

SetBrushColor(clBlack);

end;

procedure CalculateArea;

var

a, b: Double;

n: Integer;

begin

Write('Введите нижний предел интегрирования (a): ');

ReadLn(a);

Write('Введите верхний предел интегрирования (b): ');

ReadLn(b);

Write('Введите количество разбиений (n): ');

ReadLn(n);

var area := LeftRectangles(a, b, n, f);

WriteLn('Площадь фигуры: ', area:0:6);

SetWindowWidth(800);

SetWindowHeight(600);

ClearWindow(clBlack);

DrawAxes;

DrawFunction(f);

ShadingArea(a, b, n, f);

VisualizeIntegralCalculation(a, b, n, f);

end;

procedure DisplayMenu;

begin

WriteLn('1. Вычислить площадь фигуры');

WriteLn('2. Выход');

Write('Выберите действие: ');

end;

var

choice: Integer;

begin

repeat

DisplayMenu;

ReadLn(choice);

case choice of

1: CalculateArea;

2: WriteLn('Выход из программы...');

else

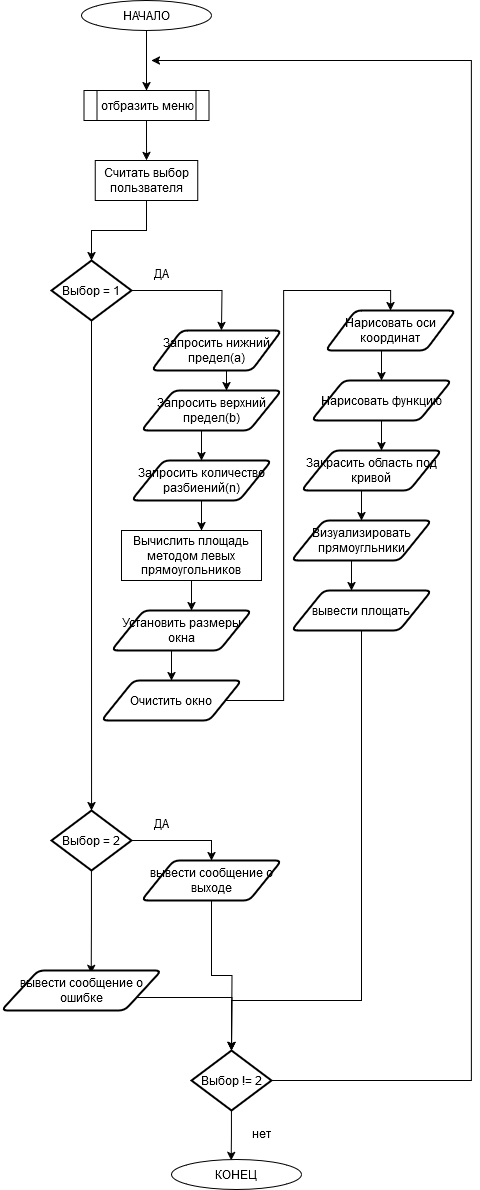
WriteLn('Неверный выбор. Попробуйте снова.');

end;

WriteLn;

until choice = 2;

end.



Вывод

В ходе выполнения лабораторного задания была разработана программа, предназначенная для вычисления площади под кривой, заданной математической функцией, и визуализации этого процесса с использованием графической библиотеки GraphABC. Программа реализует метод численного интегрирования — метод левых прямоугольников, что позволяет оценить площадь под кривой на заданном интервале.

Лабораторное задание позволило углубить понимание численного интегрирования и его применения в вычислениях. Реализация программы на Pascal ABC с использованием графической библиотеки GraphABC продемонстрировала возможности визуализации математических функций и их интегралов.

Программа может быть дополнительно улучшена, например, путем добавления других методов численного интегрирования, таких как метод трапеций или Симпсона, а также улучшения графической части для более детализированной визуализации.