Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

Колледж ВятГУ

**ОТЧЕТ**

**ПО ДОМАШНЕЙ КОНТРОЛЬНОЙ РАБОТЕ №2**

**«Изучение одномерных массивов и строк»**

**ПО ДИСЦИПЛИНЕ «Основы алгоритмизации и программирования»**

Выполнил: студент учебной группы

ИСПк-206-52-00

Федяев Алексей Сергеевич

Преподаватель:

Кузьминых Ангелина Владимировна

Киров

2023

**Цель работы (вариант 11)**: освоить принципы работы в графическом режиме; получить базовые навыки взаимодействия с графическими примитивами.

1. Дополнить программу, реализованную в ходе предыдущей лабораторной работы, режимом визуализации.

2. Предусмотреть возможность вывода кривой, ограничивающей фигуру, на координатную плоскость.

3. Реализовать следующие возможности и элементы: масштабирование графика, подписи на осях, вывод информации о задании.

4. Реализовать не менее двух возможностей из представленных: независимое масштабирование по осям, штриховка вычисляемой площади, визуализация численного расчета интеграла.

**Описание алгоритма 1:**

* Создание функции для определения значения y.
* Вывод case меню.
* Ввести пределы интегрирования и кол-во отрезков.
* Вычисление площади по правилу правых прямоугольников.
* Вычисление погрешности по формуле.
* Вывод системы координат.
* Вывод графика функции.
* Вывод границ и заливка.

**Блок схема задания 1:**



**Код программы:**

**uses** graphABC;

**const**

W = 800; H = 500;

**var** j, n1, x0, y0, x, y, xLeft, yLeft, xRight, yRight, n: integer;

limit\_a, limit\_b, h1, x2, xb, s1, t, E, a, b, dx, fmin, fmax, dy, x1, y1, mx, my, num: real;

exitProgram: boolean;

i: byte;

s: string;

**function** F (x: real): real;

**begin**

result := 2 \* x \*\* 3 - 2 \* x \*\* 2 - 5 \* x + 8;

**end**;

**begin**

exitProgram := false;

**repeat**

writeln('1. Ввод пределов интегрирования');

writeln('2. Ввод количества отрезков');

writeln('3. Вычисление интеграла и погрешности');

writeln('4. Выбор параметров визуализации');

writeln('5. Визуализация графика');

writeln('0. Выход');

writeln('Выберите опцию: '); readln(j);

**case** j **of**

1:

**begin**

writeln('Введите нижний предел интегрирования '); readln(limit\_a);

writeln('Введите верхний предел интегрирования '); readln(limit\_b);

writeln;

**end**;

2:

**begin**

writeln('Введите количество отрезков '); readln(n1);

h1 := (limit\_b - limit\_a) / n1;

s1 := 0;

xb := limit\_a;

**for** i := 1 **to** n1 **do**

**begin**

x2 := xb + i \* h1;

s1 := s1 + f(x2) \* h1;

**end**;

writeln;

**end**;

3:

**begin**

t := 6 \* x2 \*\* 2 - 4 \* x2 - 5;

E := (t / 2) \* (h1 \*\* 2) \* n1;

writeln('Интеграл равен ', s1:12:10);

writeln('Погрешность ', abs(E));

**end**;

4:

**begin**

Writeln('интервал по Х; a и b должны нацело делится на dx');

writeln('a: '); readln(a);

writeln('b: '); readln(b);

writeln('dx: '); readln(dx);

Writeln('интервал по Y; fmin и fmax должны нацело делится на dy');

writeln('fmin: '); readln(fmin);

writeln('fmax: '); readln(fmax);

writeln('dy: '); readln(dy);

writeln;

**end**;

5:

**begin**

SetWindowSize(W, H);

xLeft := 50;

yLeft := 50;

xRight := W - 50;

yRight := H - 50;

mx := (xRight - xLeft) / (b - a);

my := (yRight - yLeft) / (fmax - fmin);

x0 := trunc(abs(a) \* mx) + xLeft;

y0 := yRight - trunc(abs(fmin) \* my);

line(xLeft, y0, xRight + 10, y0);

line(x0, yLeft - 10, x0, yRight);

SetFontSize(12);

SetFontColor(clBlue);

TextOut(xRight + 20, y0 - 15, 'X');

TextOut(x0 - 10, yLeft - 30, 'Y');

SetFontSize(8);

SetFontColor(clRed);

n := round((b - a) / dx) + 1;

**for** i := 1 **to** n **do**

**begin**

num := a + (i - 1) \* dx;

x := xLeft + trunc(mx \* (num - a));

Line(x, y0 - 3, x, y0 + 3);

str(Num:0:1, s);

**if** abs(num) > 10\*\*-15 **then**

TextOut(x - TextWidth(s) **div** 2, y0 + 10, s)

**end**;

n := round((fmax - fmin) / dy) + 1;

**for** i := 1 **to** n **do**

**begin**

num := fMin + (i - 1) \* dy;

y := yRight - trunc(my \* (num - fmin));

Line(x0 - 3, y, x0 + 3, y);

str(num:0:0, s);

**if** abs(num) > 10\*\*-15 **then**

TextOut(x0 + 7, y - TextHeight(s) **div** 2, s)

**end**;

TextOut(x0 - 10, y0 + 10, '0');

x1 := a;

**while** x1 <= b **do**

**begin**

y1 := F(x1);

x := x0 + round(x1 \* mx);

y := y0 - round(y1 \* my);

**if** (y >= yLeft) **and** (y <= yRight) **then** SetPixel(x, y, clGreen);

x1 := x1 + 0.001

**end**;

x := xLeft + trunc(mx \* (limit\_a - a));

y := y0 - round(F(limit\_a) \* my);

Line(x, y0, x, y);

x := xLeft + trunc(mx \* (limit\_b - a));

y := y0 - round(F(limit\_b) \* my);

Line(x, y0, x, y);

FloodFill(x0+5,y0-5,cllime);

FloodFill(x0-5,y0-5,cllime);

SetFontColor(clblack);

**end**;

0:

exitProgram := true;

**end**;

writeln;

**if not** exitProgram **then**

**begin**

write('Нажмите Enter для продолжения...');

readln;

writeln;

ClearWindow;

**end**;

**until** exitProgram;

writeln('Программа завершена.');

**end**.

**Результат выполнения программы 1:**

|  |  |
| --- | --- |
| Входные данные | Выходные данный |
| 1 -1 100 | Интеграл равен 14.6064000000  Погрешность 0.06 |
| -0.8 0.8 50 | Интеграл равен 12.0215552000  Погрешность 0.111616 |
| -0.3 0.3 25 | Интеграл равен 4.7291808000  Погрешность 0.040752 |
| 0 0.5 10 | Интеграл равен 3.2540625000  Погрешность 0.06875 |

**Вывод:**

В ходе выполнения лабораторной работы были освоены принципы работы в графическом режиме, получены базовые навыки взаимодействия с графическими примитивами. Однако в процессе работы выявились и были решены следующие проблемы:

1. Непроверенный ввод пользователя. Если пользователь введет нечисловое значение, это может вызвать ошибку выполнения программы. И ввод неподходящих значений приведёт к ошибкам визуализации.