

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ВЯТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Институт математики и информационных систем  
Факультет автоматики и вычислительной техники  
Кафедра электронных вычислительных машин

Отчет по лабораторной работе №3  
по дисциплине  
«Программирование»

Выполнил студент гр. ИВТб-1301-05-00	_____ /Макаров С.А./
Руководитель зав. кафедры ЭВМ	_____ /Долженкова М.Л./

Киров 2024

## Цель

Цель лабораторной работы: освоить синтаксис построения процедур и функций, изучить способы передачи данных в подпрограммы, получить навыки организации минимального пользовательского интерфейса.

## Задание

Реализовать программу вычисления площади фигуры, ограниченной кривой  $2 * x^3 - 2 * x^2 + 0 * x + 16$  и осью ОХ (в положительной части по оси ОУ). Вычисление определенного интеграла должно выполняться численно, с применением метода левых прямоугольников. Пределы интегрирования вводятся пользователем. Взаимодействие с пользователем должно осуществляться посредством case-меню. Требуется реализовать возможность оценки погрешности полученного результата. Необходимо использовать процедуры и функции там, где это целесообразно.

## Решение

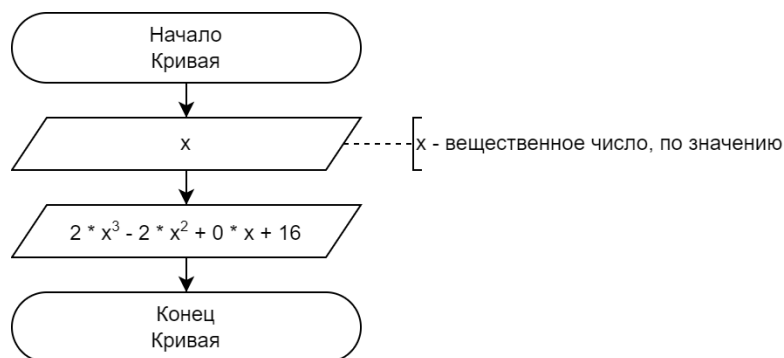


Рисунок 1 – Подпрограмма «Кривая»

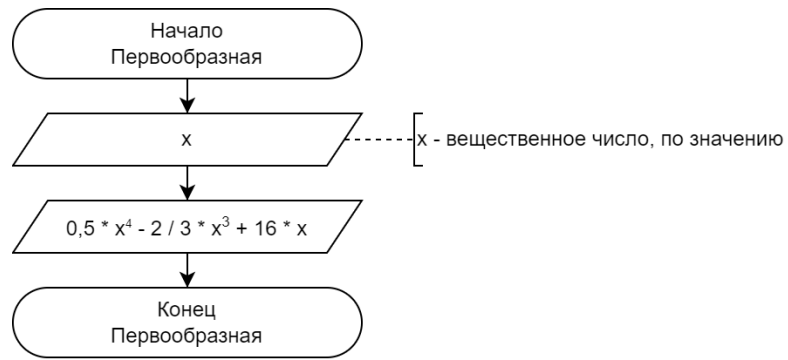


Рисунок 2 – Подпрограмма «Первообразная»

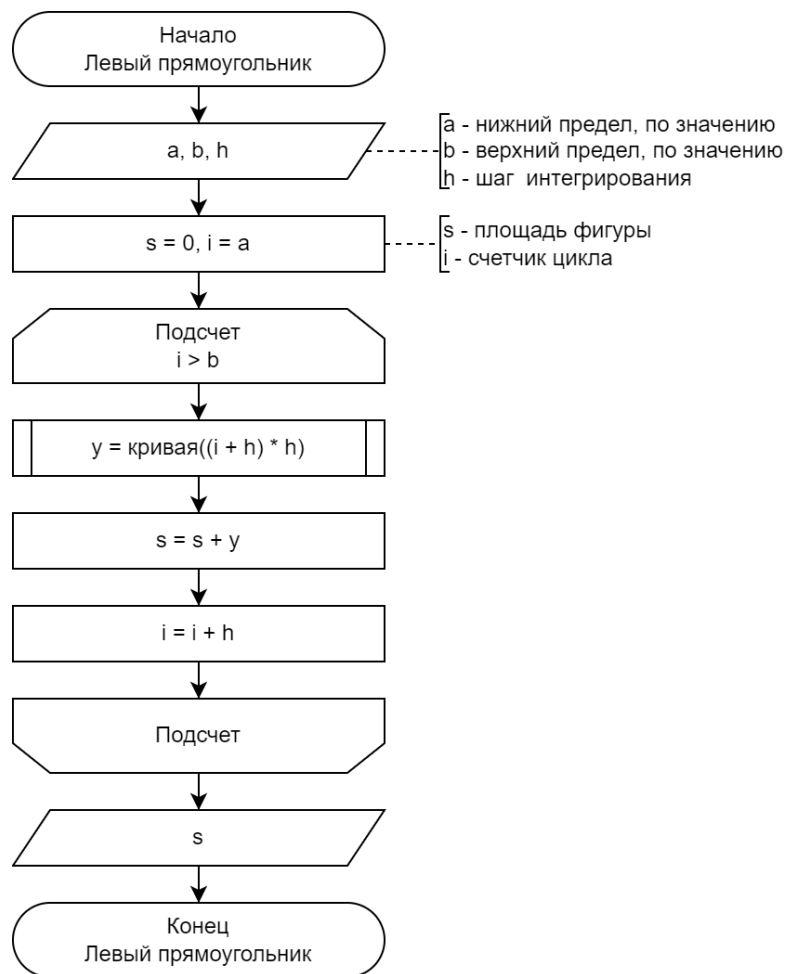


Рисунок 3 – Подпрограмма «Левый прямоугольник»

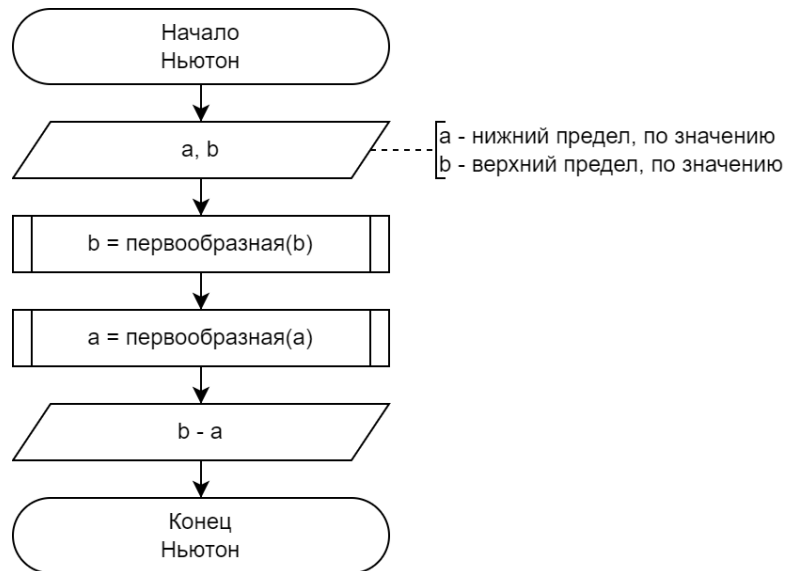


Рисунок 4 – Подпрограмма «Ньютон»

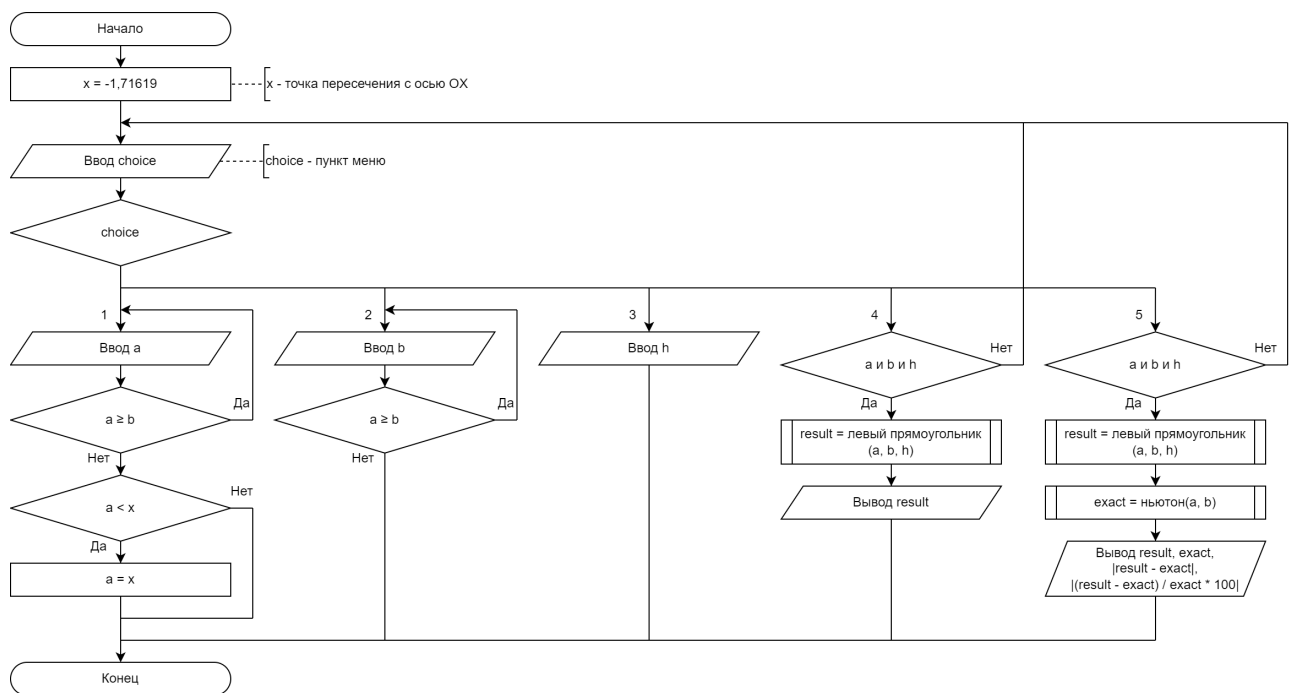


Рисунок 5 – Схема алгоритма программы

```

#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#define X -1.71619

float curve(float x) {
    return 2 * pow(x, 3) - 2 * pow(x, 2) + 0 * x + 16;
}

float antiderivative(float x) {
    return 0.5 * pow(x, 4) - 2.0 / 3.0 * pow(x, 3) + 16 * x;
}

float calc_newton(float a, float b) {
    return antiderivative(b) - antiderivative(a);
}

float left_rect(float a, float b, float h) {
    float s = 0.0;

    for (float i = a; i < b; i = i + h) {
        s += curve(i + h) * h;
    }

    return s;
}

void print_menu() {
    printf("\033[0d\033[2J");
    printf("1. Ввод нижнего предела\n");
    printf("2. Ввод верхнего предела\n");
    printf("3. Ввод шага интегрирования\n");
    printf("4. Расчет интеграла\n");
}

```

```

    printf("5. Расчет погрешности\n");
}

void print_input(int *choice) {
    printf("> ");
    scanf("%d", &*choice);
}

int main() {
    int choice, is_a = 0, is_b = 0, is_h = 0;
    float a, b, h;

    print_menu();
    print_input(&choice);

    while(1)
    {
        switch (choice)
        {
            case 1:
                print_menu();
                printf("Нижний предел: ");
                scanf("%f", &a);

                if (is_b) while (a >= b) {
                    printf("Введите корректный нижний предел: ");
                    scanf("%f", &a);
                }
                if (a < X) a = X;
                is_a = 1;
                print_input(&choice);
                break;

```

```

case 2:
    print_menu();
    printf("Верхний предел: ");
    scanf("%f", &b);

    if (is_a) while (a >= b) {
        printf("Введите корректный верхний предел: ");
        scanf("%f", &b);
    }
    is_b = 1;
    print_input(&choice);
    break;

case 3:
    print_menu();
    printf("Шаг интегрирования: ");
    scanf("%f", &h), is_h = 1;
    print_input(&choice);
    break;

case 4:
    print_menu();
    if (is_a && is_b && is_h) {
        printf("Площадь: %.2f\n", left_rect(a, b, h));
    } else {
        printf("Не введены пределы или шаг интегрирования\n");
    }
    print_input(&choice);
    break;

case 5:
    print_menu();
    if (is_a && is_b && is_h) {

```

```

        printf("Метод левых прямоугольников: %.2f\n",
               left_rect(a, b, h));
        printf("Метод Ньютона-Лейбница:
               %.2f\n", calc_newton(a, b));
        printf("Абсолютная погрешность:      %.2f\n",
               fabs(left_rect(a, b, h) - calc_newton(a, b)));
        printf("Относительная погрешность:   %.2f%%\n",
               fabs((left_rect(a, b, h) - calc_newton(a, b)) /
                   calc_newton(a, b) * 100));
    } else {
        printf("Не введены пределы или шаг интегрирования\n");
    }
    print_input(&choice);
    break;

default:
    print_menu();
    print_input(&choice);
    break;
}
}
return 0;
}

```

## Вывод

В ходе выполнения лабораторной работы удалось освоить синтаксис построения подпрограмм и способы передачи данных в них. Также удалось организовать минимальный пользовательский интерфейс. В результате была реализована программа, которая вычисляет площадь фигуры, ограниченной, а также возможность оценки погрешности полученного результата.