

ЛАБОРАТОРНАЯ РАБОТА №2

ИССЛЕДОВАНИЕ АЛГОРИТМОВ ЦИКЛИЧЕСКОЙ СТРУКТУРЫ

ХОД РАБОТЫ

1. Цель работы:

Исследование алгоритмов циклической структуры Приобретение практических навыков программирования алгоритмов циклической структуры на языке C/C++. Исследование эффективности применения различных видов циклов в задаче табулирования функции.

2. Постановка задачи и вариант задания:

Написать программу табулирования (печати таблицы значений) кусочно-заданной функции $z(x)$ на интервале от $x_{\text{нач}}$ до $x_{\text{кон}}$ с шагом Δx . Таблицу снабдить заголовком и шапкой. Вид функции z представлен на рисунке 2.1. Значения параметров $x_{\text{нач}}$, $x_{\text{кон}}$, a , b и Δx вводятся с клавиатуры. Результаты вычислений выводятся в формате с фиксированной точкой.

$$5) \quad z = \begin{cases} \ln(x) + \sin(x), & \text{если } x \leq a \\ \ln(x) + \cos(x), & \text{если } a < x < b \\ \tan(x), & \text{если } x \geq b \end{cases}$$

Рисунок 2.1 - Вариант задания

3. Краткие теоретические сведения

Для выполнения лабораторной работы необходимо внимательно изучить структуру С-программы, основные типы данных и операторы языка С, управляющие инструкции, позволяющие реализовывать программы линейной, циклической и разветвляющейся структуры, а также ознакомиться со средствами ввода-вывода языков C/C++ и функциями форматирования, описанными в библиотеке `<iomanip>` (для языка C++).

4. Структурная схема алгоритма

Структурная схема алгоритма представлена на рисунке 4.1.

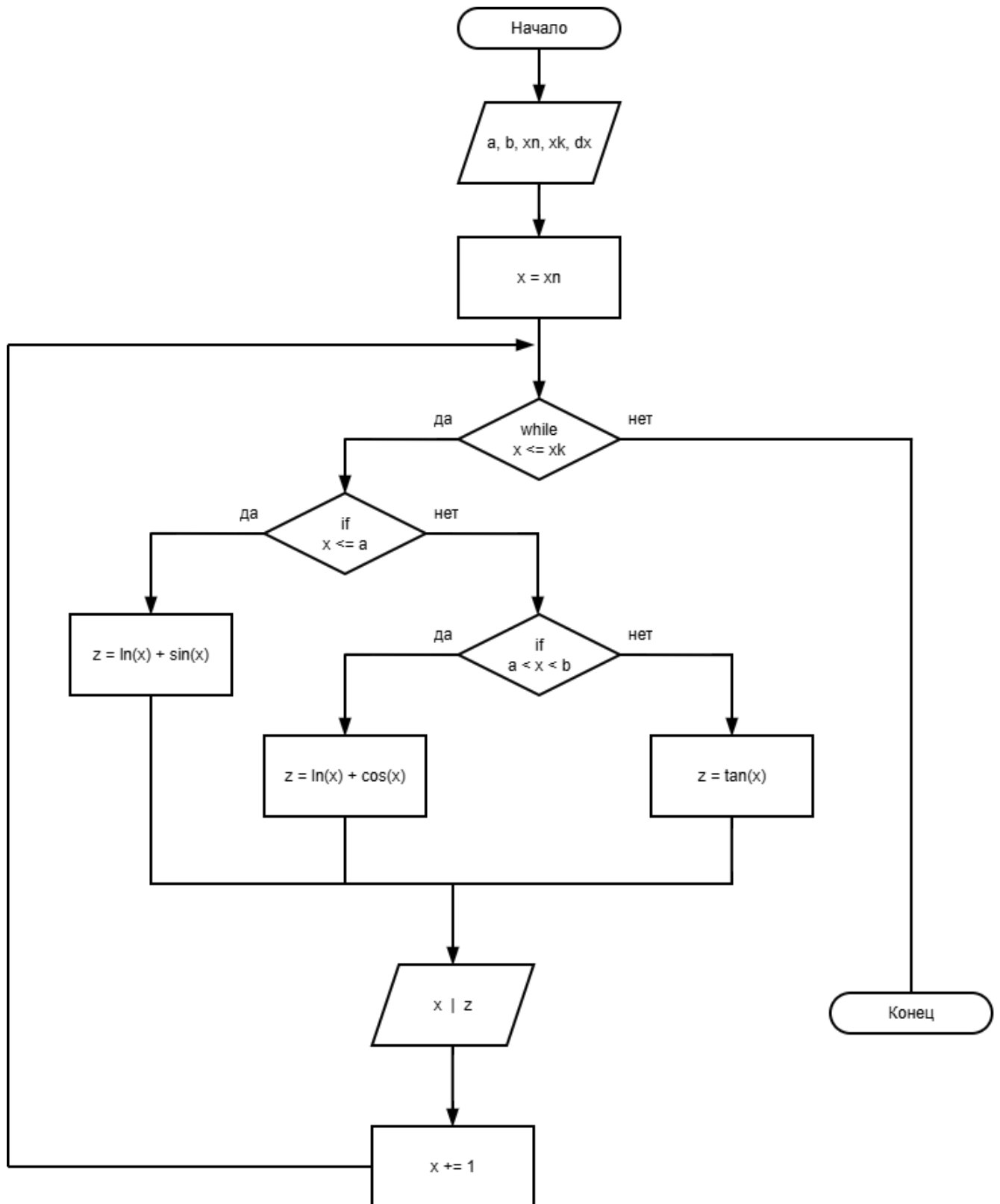


Рисунок 4.1 - блок-схема

5. Текст кода на языке C. Тестирование и отладка программ

```
#include <stdio.h>
#include <math.h>

int main() {
    float a, b, x_start, x_end, dx, x, z;

    // Ввод параметров
    printf("Введите параметр a: ");
    scanf("%f", &a);
    printf("Введите параметр b: ");
    scanf("%f", &b);
    printf("Введите начальное значение x: ");
    scanf("%f", &x);
    printf("Введите конечное значение x: ");
    scanf("%f", &x_end);
    printf("Введите шаг dx: ");
    scanf("%f", &dx);

    if (dx <= 0) {
        printf("Ошибка: шаг dx должен быть положительным!\n");
        return 1;
    }

    printf("\n");
    printf("_____ \n");
    printf("|          x          |          z = f(x)          | \n");
    printf("|_____ |_____ | \n");

    x;
    while (x <= x_end) {
        printf("| %9.4f |", x);
        if (x <= a) {
            z = log(x) + sin(x);
        }
        else if (a < x && x < b) {
            z = log(x) + cos(x);
        }
        else {
            z = tan(x);
        }
        printf(" %-11.4f | \n", z);
        x += dx;
    };

    printf("|_____ |_____ | \n");

    return 0;
}
```

На рисунках 5.1 и 5.2 отображены результаты работы кода согласно трем условиям по заданию для языка C.

Введите параметр a: 1
Введите параметр b: 7
Введите начальное значение x: 2
Введите конечное значение x: 5
Введите шаг dx: 1

x	$z = f(x)$
2.0000	0.2770
3.0000	0.1086
4.0000	0.7327
5.0000	1.8931

Рисунок 5.1 – Первое условие для языка C

Введите параметр a: 5
Введите параметр b: 9
Введите начальное значение x: 1
Введите конечное значение x: 4
Введите шаг dx: 0.5

x	$z = f(x)$
1.0000	0.8415
1.5000	1.4030
2.0000	1.6024
2.5000	1.5148
3.0000	1.2397
3.5000	0.9020
4.0000	0.6295

Рисунок 5.2 – Второе условие для языка C

6. Текст кода на языке C++. Тестирование и отладка программ

```

#include <iostream>
#include <iomanip>
#include <math.h>

using namespace std;

int main()
{
    float a, b, x_start, x_end, dx;

    cout << "Введите параметр a: ";
    cin >> a;
    cout << "Введите параметр b: ";
    cin >> b;
    cout << "Введите начальное значение x: ";
    cin >> x_start;
    cout << "Введите конечное значение x: ";
    cin >> x_end;
    cout << "Введите шаг dx: ";
    cin >> dx;
    float z = 0;
    if (dx <= 0)
    {
        cout << "Ошибка: шаг dx должен быть положительным!" <<
endl;
        return 1;
    }
    cout << fixed << setprecision(4);
    cout << "\n";
    cout << "_____ " << endl;
    cout << " |          x          |          z = f(x)          | " << endl;
    cout << " |_____ |_____ | " << endl;
    for (float x = x_start; x <= x_end; x += dx)
    {
        if (x <= a)
        {
            z = log(x) + sin(x);
        }
        else if (a < x && x < b)
        {
            z = log(x) + cos(x);
        }
        else
        {
            z = tan(x);
        }
        cout << " | " << setw(9) << x << " | " << setw(10) << z
<< " | " << endl;
    }
    cout << " |_____ |_____ | " << endl;
    return 0;
}

```

На рисунках 6.1 и 6.2 отображены результаты работы кода согласно трем условиям по заданию для языка C++.

Введите параметр a: 1
Введите параметр b: 7
Введите начальное значение x: 2
Введите конечное значение x: 5
Введите шаг dx: 1

x	$z = f(x)$
2.0000	0.2770
3.0000	0.1086
4.0000	0.7327
5.0000	1.8931

Рисунок 6.1 – Первое условие для языка C++

Введите параметр a: 5
Введите параметр b: 9
Введите начальное значение x: 1
Введите конечное значение x: 4
Введите шаг dx: 0.5

x	$z = f(x)$
1.0000	0.8415
1.5000	1.4030
2.0000	1.6024
2.5000	1.5148
3.0000	1.2397
3.5000	0.9020
4.0000	0.6295

Рисунок 6.2 – Второе условие для языка C++

ВЫВОД

В ходе лабораторной работы было проведено исследование алгоритмов линейной, разветвляющейся и циклической структур, приобретены практические навыки разработки алгоритмов циклических структур на языках C/C++. Исследованы особенности ввода-вывода значений стандартных типов в языках C/C++ с различной табуляцией (форматированием) с использованием библиотеки `<iomanip>`.