

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего
образования



НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. Р.Е.АЛЕКСЕЕВА

Институт радиоэлектроники и информационных технологий

Кафедра информатики и систем управления

ОТЧЕТ

По лабораторной работе

по дисциплине

Методы и средства обработки сигналов

РУКОВОДИТЕЛЬ:

(подпись)

_____Авербух М.Л._____
(фамилия, и.,о.)

СТУДЕНТ:

(подпись)

_____Грибанов Д.Д._____
(фамилия, и.,о.)

_____22-ВМз_____
(шифр группы)

Работа защищена «__» _____

С оценкой _____

Нижний Новгород 2024

Задание:

Имеем функцию $y(x) = a_1 * \sin(b_1 * x) + a_2 * \sin(b_2 * x) + a_3 * \sin(b_3 * x)$

Пользователем задаются: $a_1, b_1, a_2, b_2, a_3, b_3, x_0$ (начальное значение), x_k (конечное значение), Δx (шаг). Расчет $y(x)$ по заданным значениям $a_1, b_1, a_2, b_2, a_3, b_3, x_0$ (начальное значение), x_k (конечное значение), Δx (шаг).

Отображение векторов x и y (в виде таблицы). Построение графика $y(x)$ по указанным векторам.

Ход работы:

1) Ввод данных пользователем.

```
// Ввод параметров
cout << "Введите a1, b1, a2, b2, a3, b3 через пробел: ";
cin >> a1 >> b1 >> a2 >> b2 >> a3 >> b3;

cout << "Введите x0 (начало), xk (конец), dx (шаг) через пробел: ";
cin >> x0 >> xk >> dx;
```

2) Проверка корректности заданных пользователем параметров.

```
if (dx <= 0 || x0 > xk) {
    cerr << "Ошибка: некорректные значения диапазона или шага." << endl;
    return 1;
}
```

3) Создание векторов для хранения значений X и Y .

```
vector<double> x_values;
vector<double> y_values;
```

4) Вычисление функций.

```
double y_min = numeric_limits<double>::max();
double y_max = numeric_limits<double>::lowest();
for (double x = x0; x <= xk; x += dx) {
    double y = calculateY(x, a1, b1, a2, b2, a3, b3);
    x_values.push_back(x);
    y_values.push_back(y);

    // Обновление границ для графика
    if (y < y_min) y_min = y;
    if (y > y_max) y_max = y;
}
```

5) Вывод таблицы

```
cout << "      x          y(x)" << endl;
cout << "-----" << endl;
for (size_t i = 0; i < x_values.size(); i++) {
    cout << setw(10) << x_values[i] << setw(15) << y_values[i] << endl;
}
```

6) Построение графика

```
for (size_t i = 0; i < x_values.size() - 1; i++) {
    int x1 = static_cast<int>((x_values[i] - x_min) / (x_max - x_min) * (width - 1));
    int y1 = static_cast<int>((y_max - y_values[i]) / (y_max - y_min) * (height - 1));
    int x2 = static_cast<int>((x_values[i + 1] - x_min) / (x_max - x_min) * (width - 1));
    int y2 = static_cast<int>((y_max - y_values[i + 1]) / (y_max - y_min) * (height - 1));

    int dx = abs(x2 - x1), dy = abs(y2 - y1);
    int sx = x1 < x2 ? 1 : -1, sy = y1 < y2 ? 1 : -1;
    int err = dx - dy;

    while (true) {
        if (x1 >= 0 && x1 < width && y1 >= 0 && y1 < height) {
            int index = (y1 * width + x1) * 3;
            image[index] = 0;
            image[index + 1] = 0;
            image[index + 2] = 0;
        }
        if (x1 == x2 && y1 == y2) break;
        int e2 = 2 * err;
        if (e2 > -dy) { err -= dy; x1 += sx; }
        if (e2 < dx) { err += dx; y1 += sy; }
    }
}
```

Инструменты для реализации:

1) Работа с числами и вычисления

Библиотека: `<cmath>`

Выполнение математических операций (например, синусов).

2) Графические построения

Библиотека: Внешняя библиотека `stb_image_write.h`

Позволяет сохранять результат работы программы в формате графического файла (BMP).

Выбор BMP:

Формат BMP прост для работы: не требует сложного сжатия данных.

Преимущество:

stb_image_write — лёгкая библиотека для создания изображений без необходимости включения сложных графических библиотек, таких как SDL или OpenGL.

3) Работа с массивами данных

Библиотека: <vector>

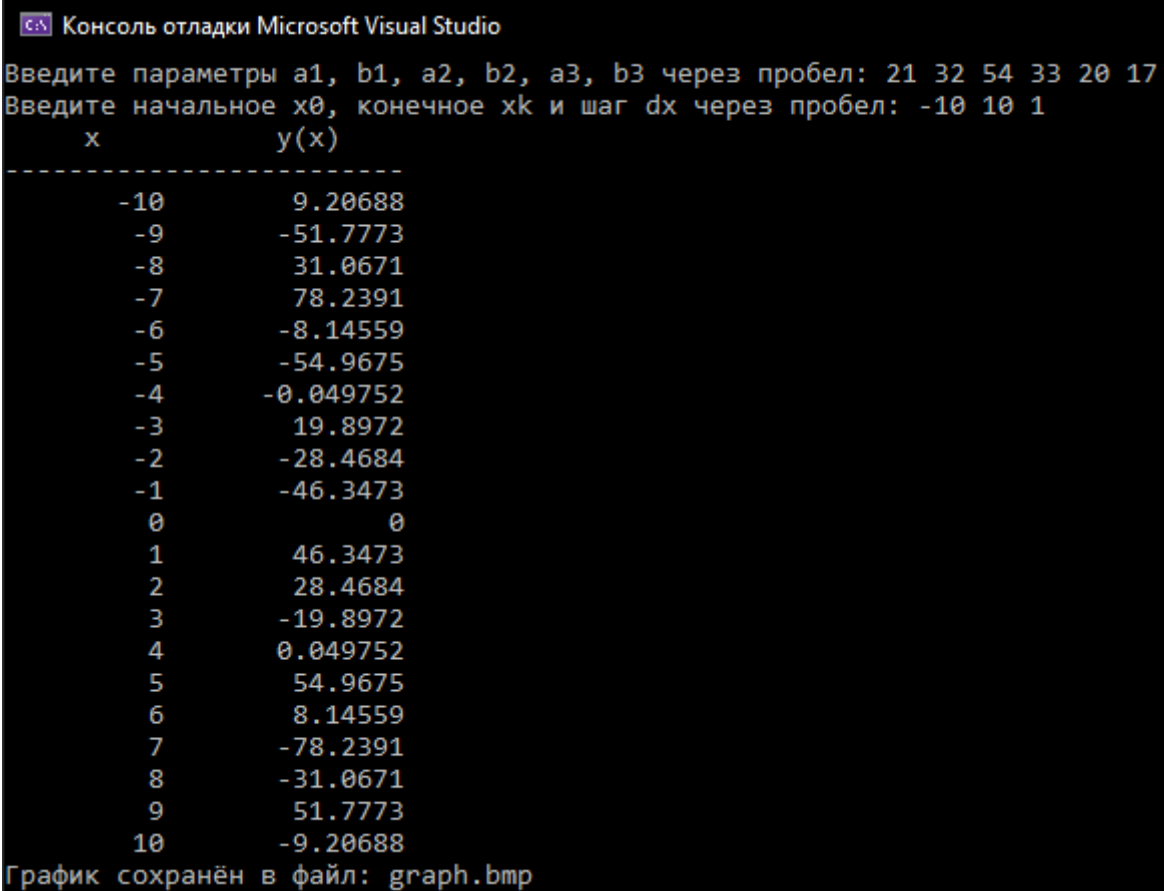
Назначение:

Хранение последовательностей значений для x и соответствующих значений y .

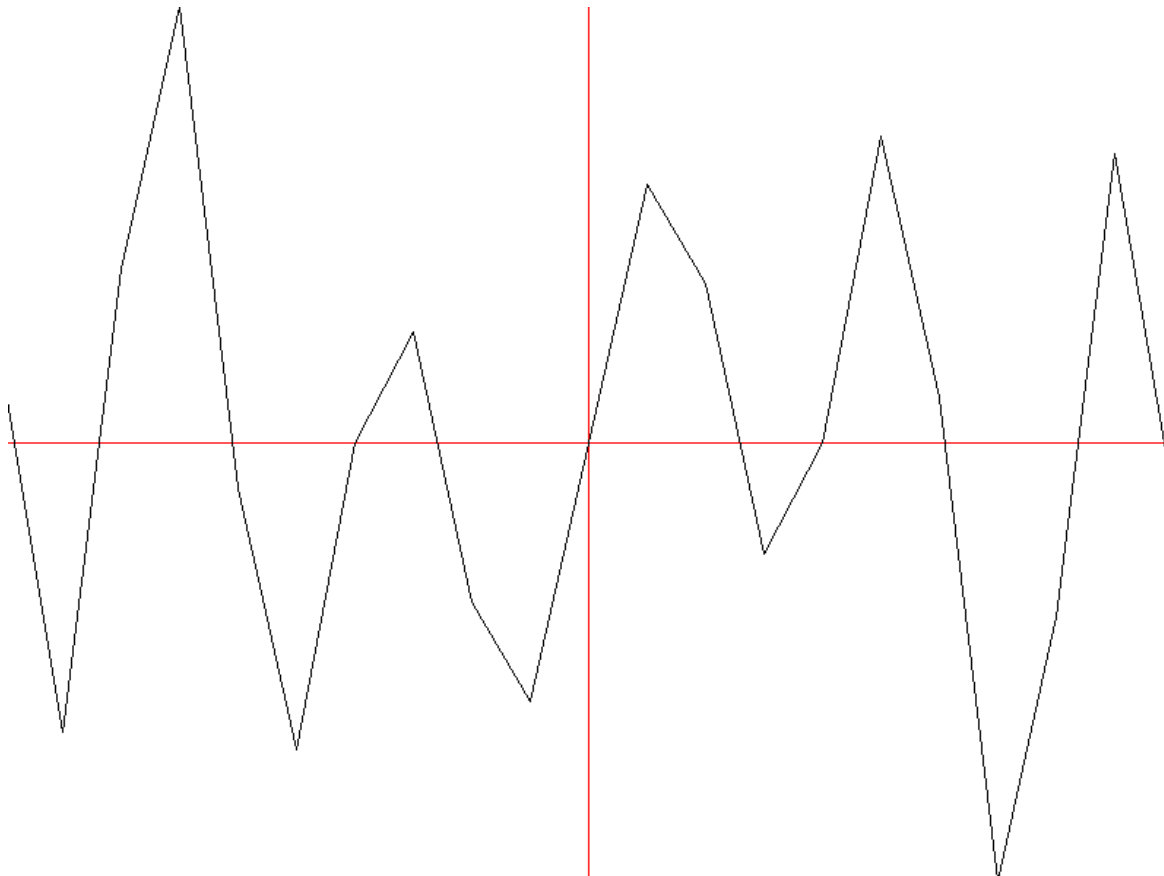
Принцип использования:

Для хранения числовых данных выбран динамический массив (std::vector) вместо статического. Это позволяет работать с неограниченным количеством точек.

Результат работы программы:



```
Консоль отладки Microsoft Visual Studio
Введите параметры a1, b1, a2, b2, a3, b3 через пробел: 21 32 54 33 20 17
Введите начальное x0, конечное xk и шаг dx через пробел: -10 10 1
  x          y(x)
-----
-10         9.20688
-9         -51.7773
-8          31.0671
-7          78.2391
-6         -8.14559
-5         -54.9675
-4        -0.049752
-3          19.8972
-2         -28.4684
-1         -46.3473
 0           0
 1          46.3473
 2          28.4684
 3         -19.8972
 4          0.049752
 5          54.9675
 6           8.14559
 7         -78.2391
 8         -31.0671
 9          51.7773
10         -9.20688
График сохранён в файл: graph.bmp
```



Вывод:

В результате выполнения лабораторной работы была разработана программа для вычислений значений функции вида $y(x) = a_1 * \sin(b_1 * x) + a_2 * \sin(b_2 * x) + a_3 * \sin(b_3 * x)$. Пользователь может задавать параметры функции, а также начальное и конечное значение переменной x и шаг Δx . Программа выводит таблицу значений $y(x)$ для каждого x в указанном интервале и строит график функции.

Это гибкое средство для анализа и визуализации подобных математических выражений, которое может быть использовано для различных задач, требующих вычислений и графического представления данных.