Национальный исследовательский ядерный университет МИФИ

Отделение Интеллектуальных кибернетических систем

Направление подготовки Информатика и вычислительная техника

**Лабораторная работа №4**

«Разработка и реализация программного компонента первичной цифровой обработки Фурье-Анализа и вычисления характеристик дискретного сигнала»

**Выполнил**: студент 2 курса группы ИBT1-Б22

Лутфиллаев Абдугаффор Самадович

**Проверил**: доцент кафедры ОИКС

Мышев Алексей Владимирович

Обнинск 2023

**Основные понятия и определения**

**Бит** – базовая единица измерения информации в каналах хранения и передачи, имеющая в них соответственно логические и физические прототипы. Логическим прототипом бита является символическое обозначение его образа, т.е. множество символов базового алфавита, в физической среде информационного канала. Например, для булевой логики – это ноль и единица, символические образы которых определяются символами 0 и 1.

**Бинарное информационное множество** – это множество элементов базового алфавита, образующих логическую структуру в виде сегмента памяти в физической среде канала хранения или поток логически и функционально связанных данных в канале передачи. В каналах передачи и хранения любой информационный объект (файлы и другие логические структуры) рассматриваются как бинарные информационные множества, на которых определяются информационные пространства, задаваемые в виде кортежа <Х,Nm>, где Х - множество символьных цепочек алфавита Nm, который представляет собой множество бинарных цепочек длиной m битов.

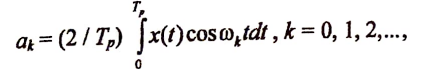
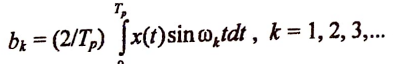
**Информационное пространство** в виде кортежа <Х,Nm> является математической моделью информационного объекта, определенного как бинарное множество. Тогда вероятностно-статистической характеристикой информационного объекта (IO) в <Х,Nm> для заданного алфавита Nm будет таблица информационной насыщенности IO, которая представляет собой дискретное распределение вероятностей букв Nm в IO.

**Определение 1.** Кодовым бинарным словом длины m называется цепочка из m битов.

**Определение 2.** Кодом называется множество кодовых слов.

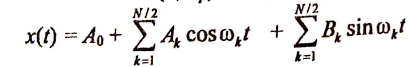
**Определение 3.** Кодовое расстояние E1(Ci,Ci+1) по Хеммингу между кодовыми словами Ci и Ci+1 определяется как количество позиций, в которых значения битов не совпадают.

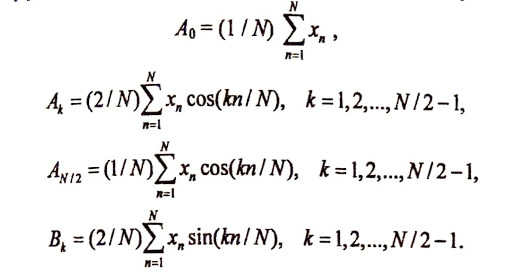
**Ряд Фурье**

Первичная цифровая обработка дискретных сигналов на основе модели разложения их в ряд Фурье включает в себя основное предположение того, что дискретная реализация непрерывного сигнала x(t) обладает периодичностью и период ее равен Tр, а основная частота f1 = 1/Tp. Тогда на основе данных дискретной реализации исходный сигнал x(t) может быть аппроксимирован рядом Фурье на интервале Tp

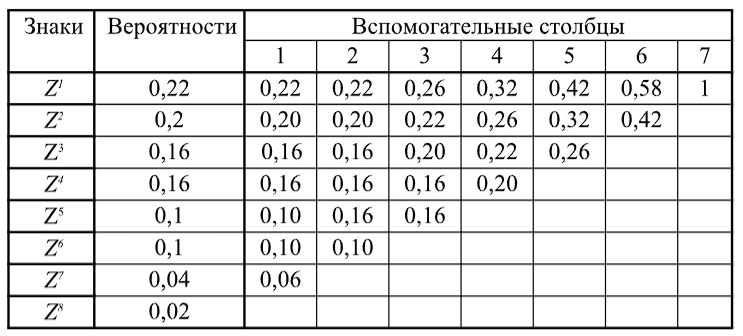
Сделаем следующие предположения: во-первых, реализация x(t) имеет конечную длину Tr = Tp, равную основному периоду; во-вторых, она состоит из четного числа N эквидистантных значений x(t), хотя это и необязательно, с шагом дискретизации h, который выбран таким образом, что частота fc = 1/(2h) достаточно высока; в-третьих, нулевая реализация равна нулю.

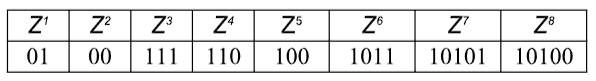
Обозначим преобразованную последовательность в виде

Вычислим по всем N значениям реализации коэффициенты ряда Фурье. Для любой точки t э (0, Tp) этот ряд будет иметь вид

**Коэффициенты Ak и Bk определяются следующими выражениями

**Метод Хаффмена**

Методика разработки алгоритмов и процедур технологий сжатия по этому методу достаточно проста и включает в себя формирование двух элементов: треугольную матрицу вероятностей и построение кодового древа. Треугольная матрица вероятностей имеет размерность (L,L), где L –количество элементов входного алфавита. Элементы столбцов этой матрицы – это вероятности. Первый столбец задается как вероятности элементов входного алфавита, а элементы последующих столбцов формируются по методике Хаффмена. Процесс формирования матрицы поясняется таблицей.

Кодовое древо для формирования элементов выходного алфавита строится на основе матрицы вероятностей. Из точки (первый элемент последнего столбца), соответствующий вероятности 1, направляем две ветви, причем ветви с большей вероятностью присваиваем – 1, а с меньшей – 0. Такое последовательное ветвление продолжаем до тех пор, пока не дойдем до вероятности каждого элемента входного алфавита. На рис. 3.1 показана иллюстрация этого процесса. После построения кодового древа, двигаясь сверху вниз записываем, (определяем) элементы выходного алфавита. Словарь будет иметь следующий вид, в котором символы Z 1 ,Z 2 ,Z 3 ,Z 4 ,Z 5 ,Z 6 ,Z 7 ,Z 8 первой строки обозначают буквы входного алфавита и соответствующие им бинарные цепочки фиксированной длины, а бинарные цепочки переменной длины во второй строке – это буквы выходного алфавита

**Цель работы**

Разработка и реализация компьютерных технологий алгоритмов первичной обработки дискретного сигнала: первичная оценка спектра и основные характеристики, возможное создание приложения визуализации результатов, освоение навыков спектрального анализа по визуальной картинке сложных и простых сигналов

**Этапы выполнения работы**

1. Разработка моделей программного компонента согласно заданию расчета коэффициентов ряда Фурье по алгоритму прямого преобразования Фурье.

2. Реализация программного интерфейса ввода исходных данных и вывода результатов через окно монитора и ввод-вывод на внешние устройства.

3. Разработка программного приложения графической визуализации результатов.

4. Отладка программных компонент.

5. Тестирование и контрольные примеры.

6. Расчеты и анализ. Выводы.

7. Прием-сдача программного компонента в диалоговом режиме.

**Входные данные**

|  |  |
| --- | --- |
| N | Количество точек |
| h | Шаг дискретизации |
| t0 | Начальная точка |

**Листинг программы:**

#include "Lab4Cmake.h"

#include <muParser.h>

#include <iostream>

#include <cmath>

#include <vector>

using namespace mu;

using namespace std;

#define M\_PI 3.1415926535

int main()

{

    setlocale(LC\_ALL, "russian");

    try

    {

        double value;

        Parser p;

        string funk;

        int N;

        double h;

        cout << "Введите функцию: ";

        getline(cin, funk);

        p.DefineVar("x", &value);

        p.DefineConst("pi", M\_PI);

        p.SetExpr(funk);

        cout << "Введите N: ";

        cin >> N;

        cout << "Введите h: ";

        cin >> h;

        cout << "Введите t0: ";

        double t0;

        cin >> t0;

        double xs = 0;

        cout << "-------------------------------------------" << endl;

        cout << "Разложение в ряд Фурье для функции: " << funk << endl;

        cout << "N = " << N << ", h = " << h << endl;

        cout << "-------------------------------------------" << endl;

        for (int k = 1; k <= N / 2 - 1; k++)

        {

            double s1 = 0, s2 = 0;

            xs = 0;

            int j = 1;

            for (int n = 1; n <= N; n++)

            {

                value = t0 + j \* h;

                double x = p.Eval();

                s1 += cos(2.0 \* M\_PI \* static\_cast<double>(k) \* static\_cast<double>(n) / static\_cast<double>(N)) \* x;

                s2 += sin(2.0 \* M\_PI \* static\_cast<double>(k) \* static\_cast<double>(n) / static\_cast<double>(N)) \* x;

                xs += t0 + x;

                j += 1;

            }

            double w = static\_cast<double>(k) / (static\_cast<double>(N) \* h);

            cout << "\nt = " << t0 + k \* h << "\t\t" << "A\_" << k << " = " << 2.0 / static\_cast<double>(N) \* s1

                << "\t\t" << "B\_" << k << " = " << 2.0 / static\_cast<double>(N) \* s2

                << "\t\t" << "W\_" << k << " = " << w << endl;;

        }

        cout << "A\_0 = " << 1.0 / static\_cast<double>(N) \* xs << endl;

        cout << "-------------------------------------------" << endl;

        int asd;

        cin >> asd;

    }

    catch (Parser::exception\_type& e)

    {

        std::cerr << e.GetMsg() << std::endl;

    }

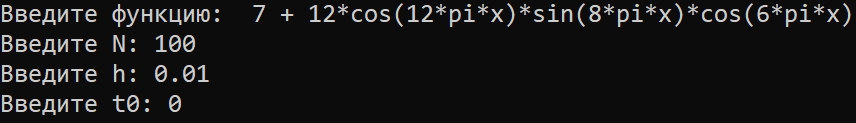
    system("pause");

    return 0;

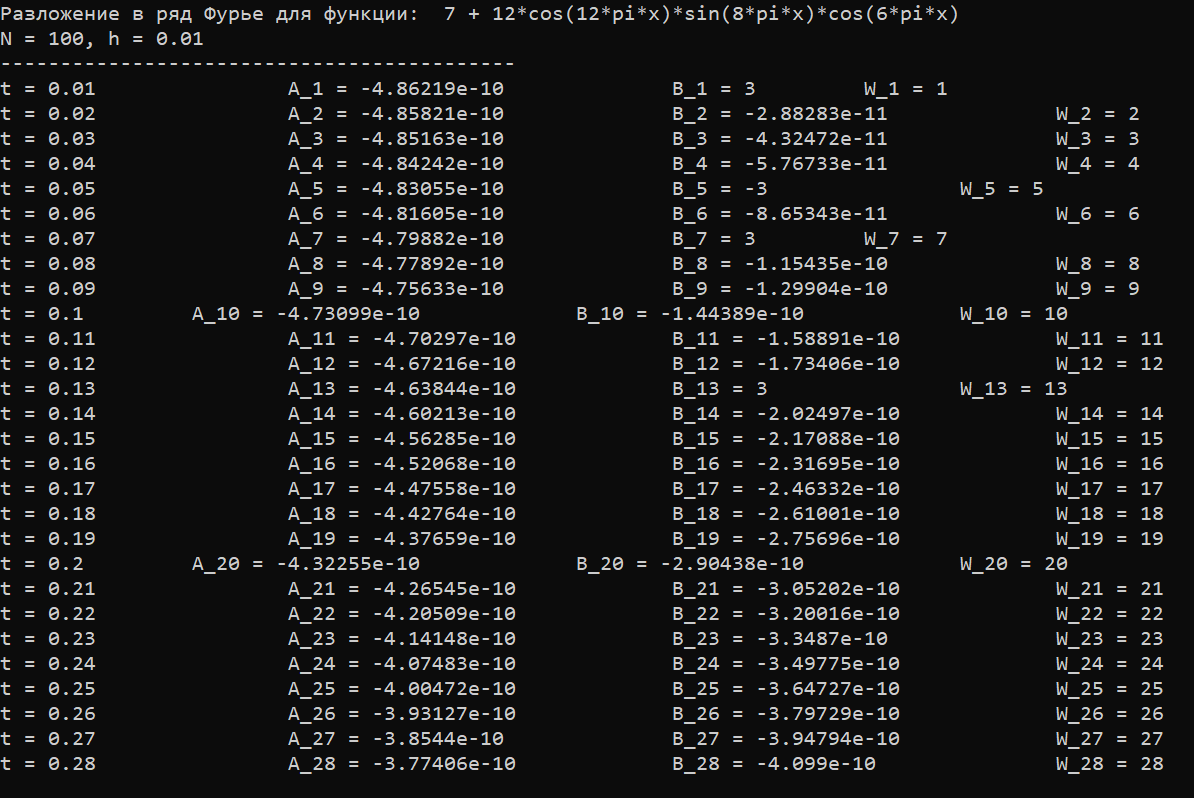
}

**Тестирование**

Начальные данные:



Работа программы:



**Вывод**:

Было разработано и реализовано компьютерная технология алгоритмов первичной обработки дискретного сигнала: первичная оценка спектра и основные характеристики. Было создано приложения визуализации результатов. Был освоен навык спектрального анализа по визуальной картинке сложных и простых сигналов.