Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

Национальный исследовательский ядерный университет «МИФИ»

Обнинский институт атомной энергетики–

Филиал федерального государственного автономного образовательного учреждения высшего образования

«Национальный исследовательский ядерный университет «МИФИ»

(ИАТЭ НИЯУ МИФИ)

Отделение Интеллектуальных кибернетических систем

Направление подготовки: 01.03.02 Прикладная математика и информатика, профиль – Прикладная информатика

УДК 51.37

Утверждаю

Директор ИАТЭ НИЯУ МИФИ

\_\_\_\_\_\_\_\_\_\_\_\_\_Т.А. Осипова

«\_\_\_» \_\_\_\_\_\_\_\_\_2021 г.

ЛАБОРАТОРНАЯ РАБОТА № 4

по теме:

Разработка и реализация программного компонента первичной цифровой обработки,

Фурье-анализа и вычисления характеристик дискретного сигнала

Студент гр. М-Б20 \_\_\_\_\_\_\_\_\_\_\_\_\_\_ С.М. Заграевская

Преподаватель,

к.ф.-м.н., доцент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.В. Мышев

Обнинск 2021

СОДЕРЖАНИЕ

I. Введение 3

1.1. Постановка цели 3

1.2. Постановка задачи и этапы выполнения работы 3

1.3. Основные понятия и определения 3

II. Основная часть 6

2.1. Листинг программы 6

2.2. Тестирование 9

III. Заключение 10

**Введение**

**1.1. Постановка цели**

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.

**1.2. Постановка задач и этапы выполнения работы**

1. Разработка модели программного компонента согласно заданию расчета коэффициентов ряда Фурье по алгоритму прямого преобразования Фурье.
2. Разработка программного интерфейса ввода исходных данных и вывода результатов, как через окна монитора, так и ввод-вывод на внешние устройства.
3. Разработать программное приложение графической визуализации результатов.
4. Отладка программных компонентов.
5. Тестирование и контрольные примеры.
6. Расчеты и анализ. Визуализация.
7. Сдача программного компонента в диалоговом режиме.

**1.3.** **Основные понятия и определения**

**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 *Tp*, а основная частота *f*1*=1/Tp****.*** Тогда на основе данных дискретной реализации исходный сигнал *x*(*t*)может быть аппроксимирован рядом Фурье на интервале *Tp*

*x*(*t*) *= a0/2 + *(*akcosωkt + bksinωkt*),(1.1)

*ak =* (*2/Tp*) *x*(*t*)*cos ωkt dt , k =* 0,1,2*,* (1.2)

*bk =*  (*2/Tp*) *x*(*t*)*sinωkt dt , k =* 1,2,3*,*

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, будем считать, что нулевая реализация равна нулю, и обозначим преобразованную последовательность в виде:

*xn = x*(*nh***) *,*** *n =* 1,2, … , *N .* (1.3)

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

*x*(*t*) *= A0 + Akcos ωkt + Bksin ωkt .* (1.4)

В точках *t = nt , n=*1,2,…,*N* ***,*** где*Tp = Nh* ***,*** а *ωk = k/Tp = k/*(*Nh*) ряд можно записать в следующем виде:

*xn = x*(*nh*) *=*   *A0 + Akcos ωkt + Bksin ωkt ,* (1.5)

или

*xn = x*(*nh*) *= A0 + Akcos*(*kn/N*)  *+ Bksin*(*kn/N*) *.* (1.6)

Коэффициенты *Ak* и *Bk*определяются следующими выражениями:

*A*0 *=* (*1/N*) *xn ,*

*Ak  =* (2*/N*) * xncos*(*kn/N*) *, k =* 1,2*,…, N/*2-1 *,* (1.7)

*AN/*2 *=* (1*/N*) *xncos*(*kn/N) , k =* 1,2,…, *N/*2-1*,*

*Bk  =* (2*/N*) *xn sin*(*kn/N) , k =* 1,2,…, *N/*2-1 ***.***

*Замечание.* Для удобства проведения расчетов и избежания возникающих «неурядиц», когда значения аргумента *t* функции *sin*(*t*) измеряются в радианах, лучше сделать переход в ряде Фурье от функции *sin*(*t*) к функции*sin*(*2πt*). Тогда гармоники базисных функций будут обращаться в ноль в рациональных и целых точках отсчетов.

**Алгоритм расчета коэффициентов *Ak*** и ***B k***

Алгоритм расчета коэффициентов ***Ak*** и ***B k*** включает в себя выполнение следующих шагов:

- определение величины ***α*** ***= (kn/N)*** при фиксированных ***k*** и ***n****;*

- вычисление *cos****α*** и*sin****α****;*

- вычисление ***xn****cos****α*** и ***xn****sin****α****;*

- вычисление суммы для каждого из этих выражений при ***n=*1,2,…*,N****;*

- приращение аргумента ***k*** на единицу и повторение всех пере-численных действий.

Такой способ требует выполнения около ***N*2** операций умножения и сложения действительных чисел.

# Основная часть

# 2.1. Листинг программы

#include <iostream>

#include <iomanip>

#include <cmath>

using namespace std;

const double PI = 3.1415926;

struct Data {

double\* x;

int N;

};

double X(double t) {

return 2 \* PI \* sin(t);

}

double A0(Data& data) {

double res = 0;

for (int i = 1; i <= data.N; i++)

res += data.x[i];

return res / data.N;

}

double A(int k, Data& data) {

double res = 0;

for (int i = 1; i <= data.N; i++)

res += data.x[i] \* cos(2 \* PI \* k \* i / data.N);

return 2 \* res / data.N;

}

double B(int k, Data& data) {

double res = 0;

for (int i = 1; i <= data.N; i++)

res += data.x[i] \* sin(2 \* PI \* k \* i / data.N);

return 2 \* res / data.N;

}

double X(int n, Data& data) {

double res = A0(data);

for (int k = 1; k < data.N / 2; k++) {

res += A(k, data) \* cos(2 \* PI \* k \* n / data.N) + B(k, data) \* sin(2 \* PI \* k \* n / data.N);

}

res += A(data.N / 2, data) / 2;

return res;

}

int main()

{

int N;

double t0, dt;

cout << "t0 dt N"<< endl;

cin>> t0 >>dt>> N;

double\* x = new double[N + 1];

for (int i = 0; i <= N; i++)

x[i] = X(t0 + dt \* i);

x[0] = x[N] = (x[0] + x[N]) / 2;

double w0, dw;

w0 = dw = 1 / (dt \* N);

Data data;

data.x = x;

data.N = N;

int shift = 20;

cout << setw(shift) << "k" << setw(shift) << "Wk" << setw(shift) <<"Ak" << setw(shift) << "Bk" << endl;

cout << setw(shift) << 0 << setw(shift) << w0 << setw(shift) << A0(data) << endl;

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

cout << setw(shift) << k << setw(shift) << w0 + k \* dw << setw(shift) << A(k, data) << setw(shift) << B(k, data) << endl;

}

cout << setw(shift) << N / 2 << setw(shift) << " " << setw(shift) << A(N / 2, data) / 2 << endl << endl << endl;

cout << setw(shift) << "i" << setw(shift) << "Xi(f)" <<setw(shift) << "Xi(s)" << setw(shift) << "delta" << endl;

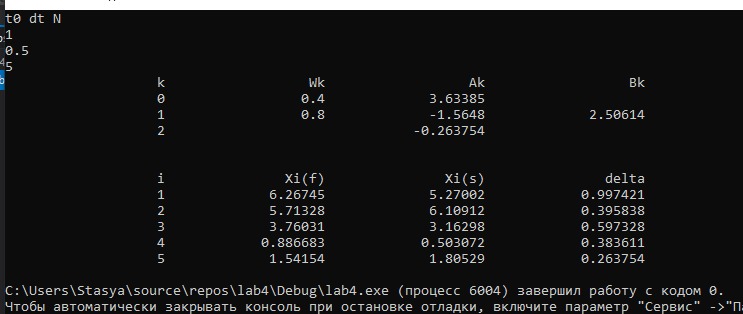
for (int i = 1; i <= N; i++)

cout << setw(shift) << i << setw(shift) << x[i] << setw(shift) << X(i, data) << setw(shift) << abs(x[i] - X(i, data)) << endl;

}

# 2.2. Тестирование

В целях тестирования программы был проведен 3 теста, результаты которого представлены ниже.



Программа отработала верно.

**ЗАКЛЮЧЕНИЕ**

Результатом данной лабораторно работы явилось создание программы на языке программирования С++, реализующей следующие функции:

• расчет коэффициентов ряда Фурье произвольного дискретного

сигнала;

• расчет основных характеристик сигнала;

• диалоговый режим взаимодействия;

• графическая визуализацию результатов.

Через решение поставленной задачи были изучены и апробированы методы первичной обработки цифровых сигналов, приобретены навыки основ цифрового спектрального анализа дискретных сигналов, навыки разработки программных компонентов первичной обработки дискретных сигналов, навыки анализа результатов на основе технологии компьютерной визуализации.