МИНИСТЕРСТВО НАУКИ и ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**Коломенский институт (филиал)**

федерального государственного бюджетного образовательного учреждения высшего образования

**«Московский политехнический университет»**

**Отчет по контрольной работе**

**по дисциплине «Проектная деятельность»**

на тему: «Амплитудно-фазовая характеристика системы автоматического регулирования скорости электродвигателя постоянного тока»

Выполнили студенты ИВТ-21, группы №3: Крупин Максим, Ковалёв Максим, Хоботов Михаил, Готманов Илья

Руководитель /Лобанов А.П./

(подпись)

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата 29.12.2020

**Описание программы**

**Логическая структура программы:**

Программа состоит из одного модуля Unit1, в котором производится расчет P(W) – вещественная частотная характеристика, Q(W) – мнимая частотная характеристика, где W – промежуток от 0 до 50 с-1. По ходу программы W изменяется с шагом 1 с-1. Полученные данные позволяют построить характеристику на комплексной плоскости, которую представляет график Chart1.

Помимо объекта Chart1 на форме есть поля для ввода промежутка W (Edit1, Edit2), Поля для ввода значений, необходимых для решения задачи:

1. Ky – Edit3
2. Ce – Edit4
3. K0 – Edit5
4. Tm – Edit6
5. Tk – Edit7

Объект StringGrid1, который также присутствует на форме, нужен для отображения полученных результатов, полученных в ходе работы программы. На нем отображаются:

1. Номер каждой итерации расчетного цикла.
2. Значение W на данном шаге расчетного цикла.
3. Значение выполнения функции P(W).
4. Значение выполнения функции Q(W).

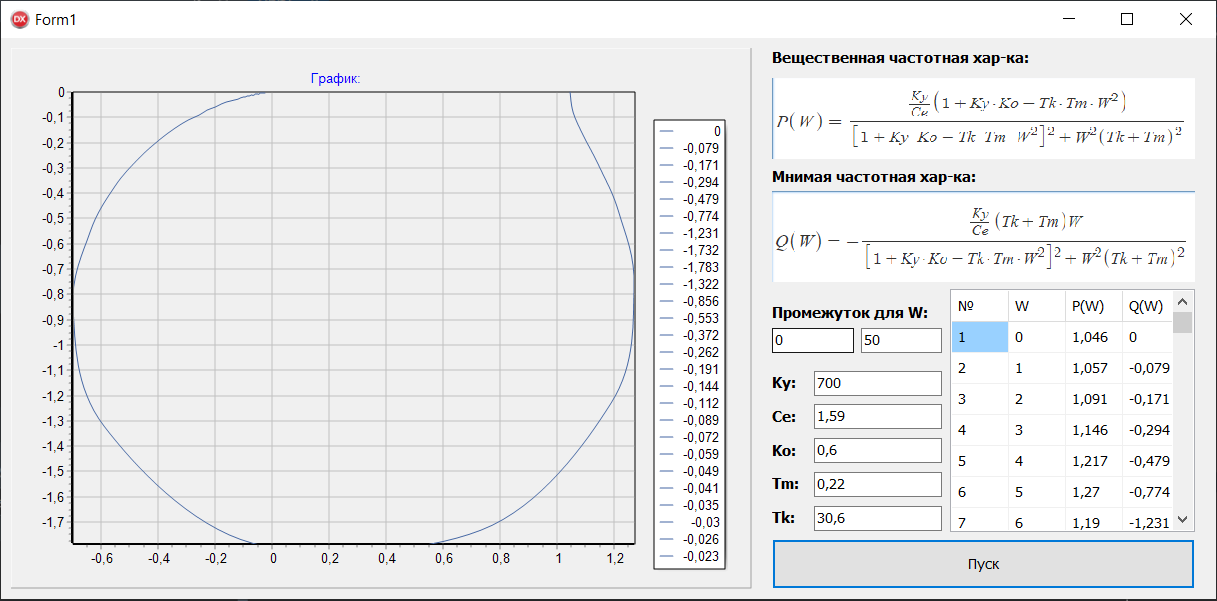
Объекты Image1 и Image2 отвечают за вывод формул для вычисления функций P(W) и Q(W) в виде изображений.

Для выполнения работы программы необходим объект Button1. Это кнопка, которая отвечает за вызов расчетных функций:

function P(PKy, PCe, PKo, PTk, PTm, PW:real):real;

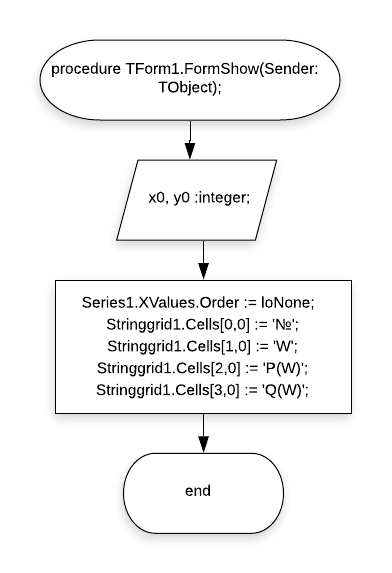
function Q(PKy, PCe, PKo, PTk, PTm, PW:real):real;

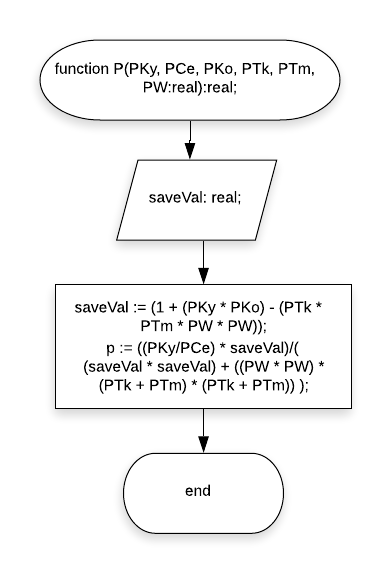
Срабатывает по нажатию на нее (TForm1.Button1Click).

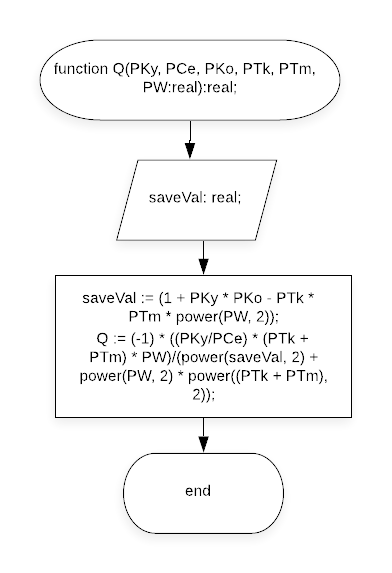


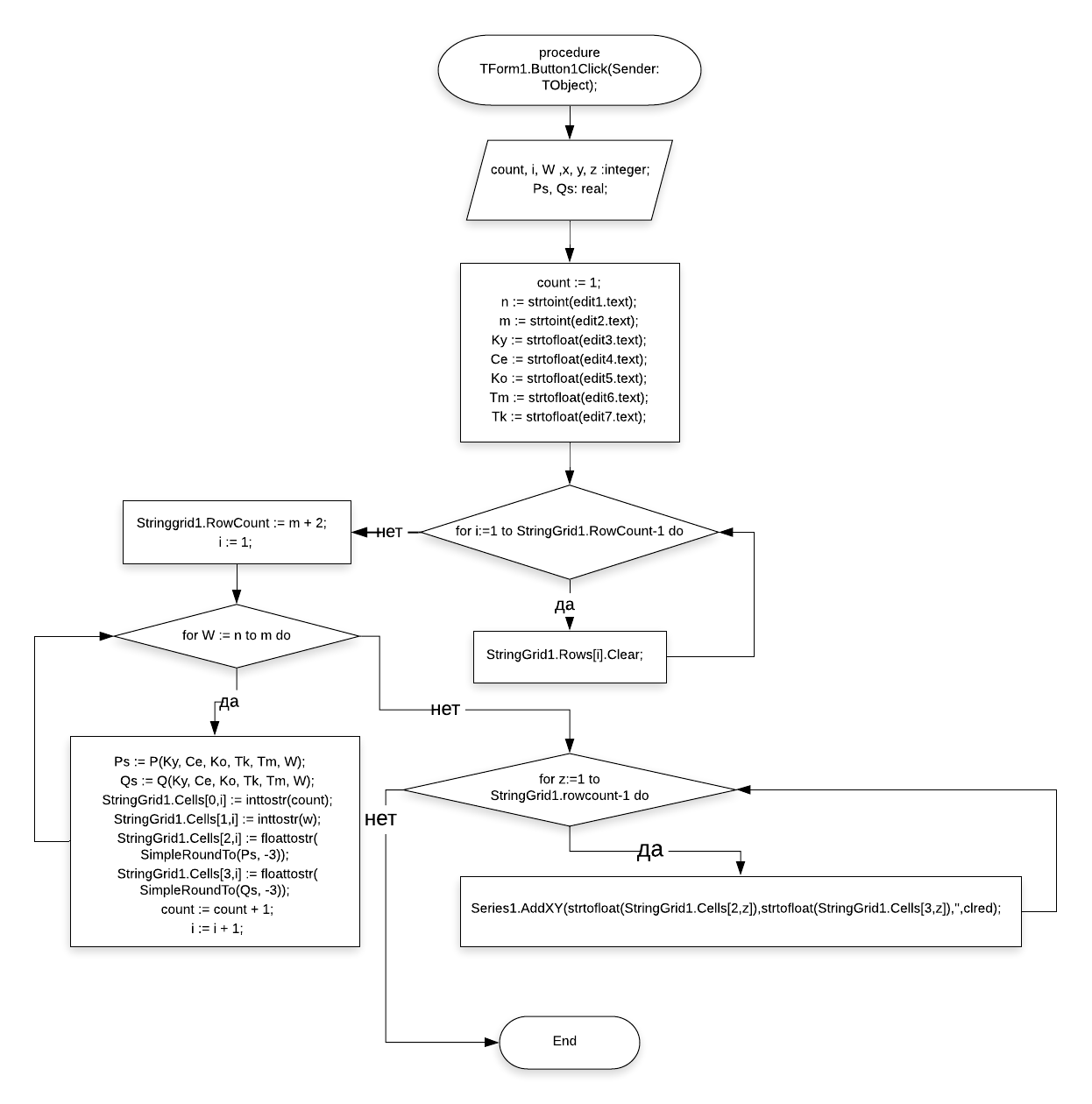
**Блок-схемы:**

1. Оформление таблицы StringGrid1

при запуске программы.

1. Расчет P(W).

3) Расчет Q(W).

4) Заполнение таблицы, расчет, построение графика.

**Описание программы:**

Работа функции TForm1.Button1Click заключается в том, что в ходе цикла с помощью подпрограмм Function P и Function Q, рассчитываются вещественная и мнимая частотные характеристики P(w) и Q(W), изменяя шаг на 1 с-1 в диапазоне 0≤w≤50 с-1 (по условию задачи). Данный промежуток может быть введен пользователем самостоятельно в поля Edit1 и Edit2. Каждая итерация цикла сопровождается занесением полученных значений в таблицу StringGrid1. Все вычисления происходят, основываясь на данных, введённых пользователем в поля Edit3- Edit7.

По окончании расчетов строится график. Для этого используется метод построения по точкам, что отображено в коде, как:

Series1.AddXY(strtofloat(StringGrid1.Cells[2,z]),strtofloat(StringGrid1.Cells[3,z]),'',clred);

Точки для его построения – значения, полученные в ходе работы программы, что отображены в таблице StringGrid1. Построение графика происходит в цикле.

**Код программы:**

unit Unit1;

interface

uses

Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants, System.math, System.Classes, Vcl.Graphics,

Vcl.Controls, Vcl.Forms, Vcl.Dialogs, VclTee.TeeGDIPlus, Vcl.StdCtrls,

Vcl.ExtCtrls, VCLTee.TeEngine, VCLTee.TeeProcs, VCLTee.Chart,

Vcl.Imaging.pngimage, Vcl.Grids, VCLTee.Series, VCLTee.TeeFunci;

type

TForm1 = class(TForm)

Image1: TImage;

Image2: TImage;

Label1: TLabel;

Label2: TLabel;

Edit1: TEdit;

Edit2: TEdit;

Label3: TLabel;

Label4: TLabel;

Label5: TLabel;

Label6: TLabel;

Label7: TLabel;

Label8: TLabel;

Edit3: TEdit;

Edit4: TEdit;

Edit5: TEdit;

Edit6: TEdit;

Edit7: TEdit;

Button1: TButton;

StringGrid1: TStringGrid;

Chart1: TChart;

Series1: TLineSeries;

TeeFunction1: TCustomTeeFunction;

procedure Button1Click(Sender: TObject);

procedure FormShow(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form1: TForm1;

n,m: integer;

Ky, Ce, Ko, Tm, Tk: real;

implementation

{$R \*.dfm}

function P(PKy, PCe, PKo, PTk, PTm, PW:real):real; //P

var

saveVal: real;

begin

saveVal := (1 + (PKy \* PKo) - (PTk \* PTm \* PW \* PW));

p := ((PKy/PCe) \* saveVal)/( (saveVal \* saveVal) + ((PW \* PW) \* (PTk + PTm) \* (PTk + PTm)) );

end;

function Q(PKy, PCe, PKo, PTk, PTm, PW:real):real; //Q

var

saveVal: real;

begin

saveVal := (1 + PKy \* PKo - PTk \* PTm \* power(PW, 2));

Q := (-1) \* ((PKy/PCe) \* (PTk + PTm) \* PW)/(power(saveVal, 2) + power(PW, 2) \* power((PTk + PTm), 2));

end;

procedure TForm1.Button1Click(Sender: TObject);

var

count, i, W ,x, y, z :integer;

Ps, Qs: real;

begin

count := 1;

n := strtoint(edit1.text);

m := strtoint(edit2.text);

Ky := strtofloat(edit3.text);

Ce := strtofloat(edit4.text);

Ko := strtofloat(edit5.text);

Tm := strtofloat(edit6.text);

Tk := strtofloat(edit7.text);

for i:=1 to StringGrid1.RowCount-1 do

StringGrid1.Rows[i].Clear;

Stringgrid1.RowCount := m + 2;

i := 1;

for W := n to m do

Begin

Ps := P(Ky, Ce, Ko, Tk, Tm, W);

Qs := Q(Ky, Ce, Ko, Tk, Tm, W);

StringGrid1.Cells[0,i] := inttostr(count);

StringGrid1.Cells[1,i] := inttostr(w);

StringGrid1.Cells[2,i] := floattostr( SimpleRoundTo(Ps, -3));

StringGrid1.Cells[3,i] := floattostr( SimpleRoundTo(Qs, -3));

count := count + 1;

i := i + 1;

End;

for z:=1 to StringGrid1.rowcount-1 do

Series1.AddXY(strtofloat(StringGrid1.Cells[2,z]),strtofloat(StringGrid1.Cells[3,z]),'',clred);

end;

procedure TForm1.FormShow(Sender: TObject);

var

x0, y0 :integer;

begin

Series1.XValues.Order := loNone;

Stringgrid1.Cells[0,0] := '№';

Stringgrid1.Cells[1,0] := 'W';

Stringgrid1.Cells[2,0] := 'P(W)';

Stringgrid1.Cells[3,0] := 'Q(W)';

end;

end.

**Техническое задание**

1. **Назначение и область применения**

Важную роль при описании динамики линейных систем играют частотные характеристики. Основной частотной характеристикой является амплитудно-фазовая характеристика (АФХ).

Амплитудно-фазовой характеристикой называется конформное отображение мнимой оси плоскости корней характеристического уравнения на комплексную плоскость амплитудно-фазовой характеристики, осуществляемое функцией W(s).

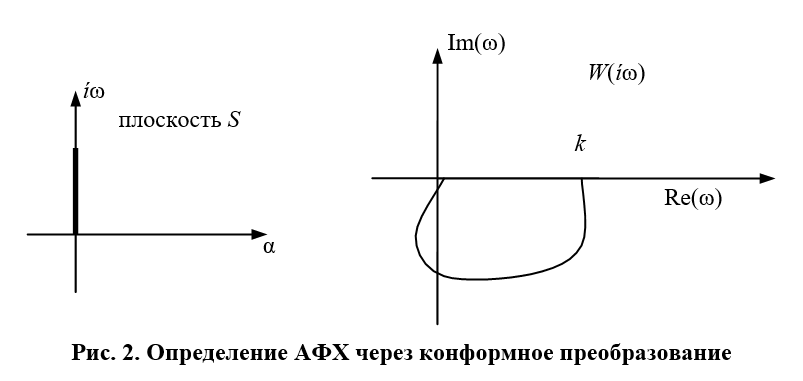
***W* (*i*ω) = Re(ω) + *i* Im(ω)**

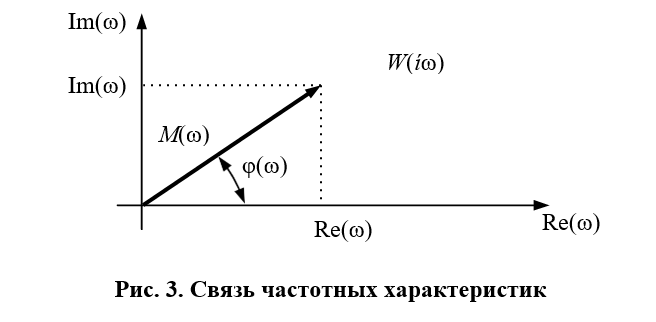
Где *М* (ω)амплитудно-частотная характеристика (АЧХ);

(ω) – Фазово-частотная характеристика (ФЧХ);

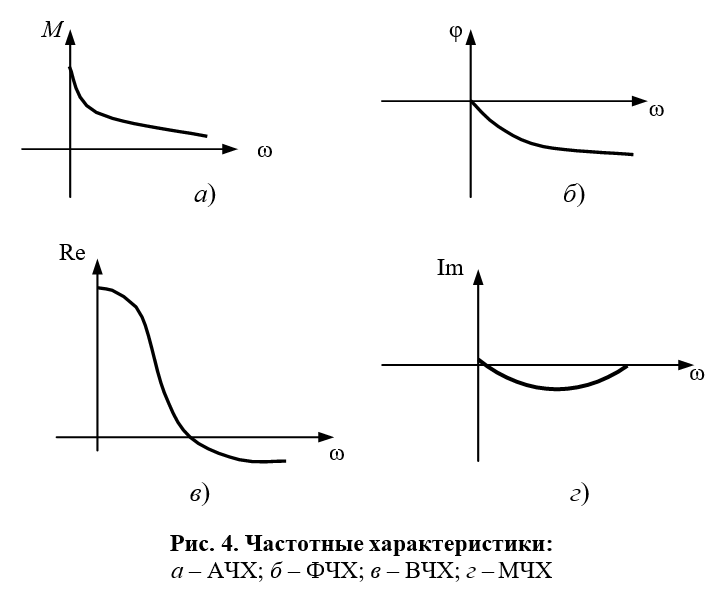
Re(ω) – Вещественная характеристика (ВЧХ);

Im (ω) – Мнимая частотная характеристика (МЧХ);





На практике частотные характеристики получают по передаточной функции. Механизм записи АФХ по передаточной функции сводится к замене в последней комплексного параметра *s* на *ί*ω, т.е. *s* = *ί*ω. Полученное выражение далее преобразуется к показательной форме или алгебраической форме записи амплитудно-фазовой характеристики. Поскольку физически отрицательных частот в природе не существует, все частотные характеристики строятся только для положительных частот. Амплитудно-частотная и вещественно-частотная характеристики являются чётными функциями, а фазово-частотная и мнимая частотная характеристики являются нечётными функциями. В случае необходимости для отрицательных частот, частотные характеристики получают зеркальным отображением относительно действительной оси для чётных характеристик, либо относительно начала координат – для нечётных характеристик. Примеры графиков частотных характеристик представлены на рис. 4.



Степень различия между параметрами входного и выходного сигналов не зависит от амплитуды и фазы входного сигнала, а определяются только динамическими свойствами объекта и частотой колебаний.

Для получения частотных характеристик экспериментальным путём проводится ряд экспериментов, при которых на вход объекта подаётся гармонический сигнал заданной амплитуды и частоты. У полученного на выходе гармонического сигнала измеряется амплитуда и сдвиг фаз. В результате проведённых экспериментов с различными значениями частот частотные характеристики определяются следующим образом.

Амплитудно-частотная характеристика (АЧХ) представляет собой отношение амплитуды выходного сигнала к амплитуде входного сигнала:

*M*   *B* / *A* .

Фазово-частотная характеристика (ФЧХ) – это разность фаз выходного и входного сигналов.

  вых  вх .

Таким образом, амплитудно-фазовая характеристика (АФХ) определяется как комплексная функция, у которой модулем является АЧХ, а фазой – ФЧХ.

1. **Технические и основные требования:**

* Запас устойчивости
* Статическая точность или величина ошибки системы в установившемся режиме работы;
* Динамический режим САУ (переходные процессы);
* Динамическая точность – значение ошибки регулирования при изменяющемся воздействии;

1. **Необходимые стадии разработки:**

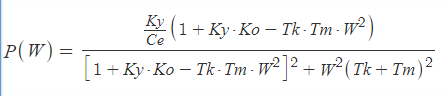
* постановка задачи
* определение и уточнение требований к техническим средствам;
* определение требований к программе;
* определение стадий, этапов и сроков разработки программы и документации на нее;
* согласование и утверждение технического задания.

1. **Постановка задачи**

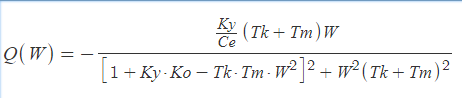
Разработать программу для определения амплитудно-фазовой характеристики системы автоматического регулирования скорости электродвигателя постоянного тока.

Для реализации программы даны следующие данные:

* Вещественная частотная характеристика системы P(W):



* Мнимая частотная характеристика системы Q(W):



Требуется:

1. Вычислить P(W) и Q(W) с шагом w = 1с-1 в диапазоне 0≤w≤50 с-1.
2. Построить по точкам характеристику на комплексной плоскости.

**Руководство программиста**

1. **Назначение и область применение программы**

Программа предназначена для определения амплитудно-фазовой характеристики системы автоматического регулирования скорости электродвигателя постоянного тока, используя предоставленные данные об вещественной частотной характеристики системы P(W), а также мнимой частотной характеристике Q(W).

Необходимые условия для выполнения программы, следующие:

Программа используется на персональном компьютере (ПК) типа IBM PC/AT. Для работы в диалоговом режиме используется экран дисплея, клавиатура и манипулятор типа «мышь». Для поддержки графического режима необходим адаптер SVGA. Входные данные хранятся на гибком и/или жестком дисках. Программа работает под управлением OC Windows 7, 8, 8.1, 10.

1. **Характеристики программы**

Программа выполняется в режиме реального времени, то есть, пока приложение открыто на вычислительной машине оно будет готово к выполнению подпрограмм для получения результатов.

Контроль правильности выполнения программы:

1. Необходимо ввести коэффициенты в поля, располагающиеся на форме. Каждое поле подписано, что позволяет быть уверенным в правильности ввода при наличии корректных данных.
2. При нажатии на кнопку «Пуск», для запуска функции «Button1Click», рассчитываются значения вещественной частотной характеристики P(W) и мнимой частотной характеристики Q(W), изменяя шаг на 1 с-1 в диапазоне 0≤w≤50 с-1.
3. По ходу работы алгоритма, полученные при расчете значения заносятся в таблицу, после чего, основываясь на полученных данных, по точкам строится график – искомая характеристика.
4. **Обращение к программе**

Загрузка и запуск проекта осуществляется следующими способами:

1. При помощи двойного щелчка по значку программы в окне «Проводник» или на рабочем столе;
2. Из главного меню Windows (меню «Пуск»);
3. При помощи выполнения команды «Выполнить»;
4. И другие способы.

Описание выполняемых функций находиться в разделе «Описание программы».

Завершение работы программы может осуществляться через стандартный способ – нажатие на кнопку закрытия программы («крестик на красном фоне»). Также программа может быть завершена с помощью прямой выгрузки из оперативной памяти, для этого надо сделать следующее:

1. Однократным одновременным нажатием клавиш «Ctrl + Alt + Del» на клавиатуре ПК, вызвать окно диспетчера задач;
2. В данном окне, среди запущенных программы выбрать проект;
3. Нажать клавишу «Enter» на клавиатуре ПК или кнопку «Завершить задачу», используя манипулятор мыши.
4. **Входные и выходные данные**

В программе предусмотрен ввод данных пользователем:

* a ≤w≤ b с-1, Где a – Edit1, b – Edit2
* Ky – Edit3
* Ce – Edit4
* K0 – Edit5
* Tm – Edit6
* Tk – Edit7

1. **Сообщения**

В приложении для решения поставленных задач вывод сообщения не предусмотрен.

**Описание языка программирования**

**Delphi** (Де́лфи) — [императивный,](https://ru.wikipedia.org/wiki/%D0%98%D0%BC%D0%BF%D0%B5%D1%80%D0%B0%D1%82%D0%B8%D0%B2%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) [структурированный](https://ru.wikipedia.org/wiki/%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5), [объектно-ориентированный](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5), [высокоуровневый](https://ru.wikipedia.org/wiki/%D0%92%D1%8B%D1%81%D0%BE%D0%BA%D0%BE%D1%83%D1%80%D0%BE%D0%B2%D0%BD%D0%B5%D0%B2%D1%8B%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F) [язык программирования](https://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F) со [строгой](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D0%B8_%D1%81%D0%BB%D0%B0%D0%B1%D0%B0%D1%8F_%D1%82%D0%B8%D0%BF%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F) [статической типизацией](https://ru.wikipedia.org/wiki/%D0%A1%D1%82%D0%B0%D1%82%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D1%82%D0%B8%D0%BF%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F) переменных. Основная область использования — написание прикладного программного обеспечения.

Этот язык программирования является [диалектом](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D0%B0%D0%BB%D0%B5%D0%BA%D1%82) языка [Object Pascal](https://ru.wikipedia.org/wiki/Object_Pascal). Изначально язык Object Pascal относился к несколько другому языку, который был разработан в фирме [Apple](https://ru.wikipedia.org/wiki/Apple) в 1986 году группой [Ларри Теслера](https://ru.wikipedia.org/wiki/%D0%9B%D0%B0%D1%80%D1%80%D0%B8_%D0%A2%D0%B5%D1%81%D0%BB%D0%B5%D1%80)[[2]](https://ru.wikipedia.org/wiki/Delphi_(%D1%8F%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F)#cite_note-2). Однако, начиная с Delphi 7[[3]](https://ru.wikipedia.org/wiki/Delphi_(%D1%8F%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F)#cite_note-3), в официальных документах

компании [Borland](https://ru.wikipedia.org/wiki/Borland) название Delphi стало использоваться для обозначения языка, ранее известного как [Object Pascal](https://ru.wikipedia.org/wiki/Object_Pascal).

Pascal - один из языков программирования, который помогает писать «структурированные» программы в таком виде, в котором структура программы должна непосредственно отражать структуру задачи. Эта особенность языка Pascal, а также его достаточная простота из-за интуитивной понятности его конструкций, позволила языку завоевать прочное место среди языков программирования.

Pascal является традиционным алгоритмическим языком программирования, продолжающим линию Algol-60. Это означает, что программа на языке Pascal представляет собой специально организованную последовательность шагов по преобразованию данных, приводящую к решению некоторой задачи. Это отличает Pascal от так называемых непроцедурных языков типа Prolog, по существу, представляющих собой формализмы для записи начальных условий некоторой задачи и синтезирующих решение посредством встроенных механизмов логического вывода.

Язык Pascal содержит удобные средства для представления данных. Развитая система типов позволяет адекватно описывать данные, подлежащие обработке, и конструировать структуры данных произвольной сложности. Pascal является типизированным языком, что означает фиксацию типов переменных при их описании, а также строгий контроль преобразований типов и контроль доступа к данным в соответствии с их типом (как на этапе компиляции, так и при исполнении программ).

Набор операторов языка Pascal отражает принципы структурного программирования и позволяет записывать достаточно сложные алгоритмы в компактной и элегантной форме. Pascal является процедурным языком с традиционной блочной структурой и статически определенными областями действия имен. Процедурный механизм сочетает в себе простоту реализации и использования и гибкие средства параметризации.

Синтаксис языка достаточно несложен. Программы записываются в свободном формате, что позволяет сделать их наглядными и удобными для изучения.

Паскаль – компилятор, то есть, прежде чем начать исполнение программы, Паскаль полностью прочитывает исходный текст, написанный программистом, и составляет последовательность машинных кодов, выполняющую те действия, которые описал программист в hqundmnl тексте. Эта последовательность сохраняется в файл с расширением “.EXE” и является самостоятельным исполняемым файлом, который может быть запущен сам по себе, уже без участия Паскаля и, даже, на другом компьютере, на котором Паскаль может быть не установлен.