**МИНОБРНАУКИ РОССИИ**

Федеральное государственное бюджетное

образовательное учреждение высшего образования

**«Ухтинский государственный технический университет»**

**(УГТУ)**

Кафедра вычислительной техники, информационных систем и технологий

**ОТЧЕТ ПО УЧЕБНОЙ ПРАКТИКЕ**

(ознакомительная практика)

Шифр 191407 Группа ИСТ-2-19 Курс 1

Морданов Егор Владимирович

Проверил:

ст. преподаватель кафедры ВТИСиТ С. С. Сочко

Ухта

2020

Содержание

[ВВЕДЕНИЕ 3](#_Toc45628620)

1. [Условие задачи](#_Toc40557997) 3
2. [Постановка задачи](#_Toc40557998) 3
3. [Сбор и анализ информации](#_Toc40557999) 3
4. [Разработка программы](#_Toc40557999) 5

[ЗАКЛЮЧЕНИЕ 7](#_Toc45628621)

[СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ](#_Toc45628621) 8

[ПРИЛОЖЕНИЕ](#_Toc45628621) 9

# ВВЕДЕНИЕ

Место прохождения практики: кафедра ВТИСиТ УГТУ.

Цели и задачи прохождения практики: рассмотреть метод поиска минимума функции одной переменной, методом половинного деления. Научиться основным понятием и терминам, самостоятельно определять минимум функции, чётко понимать каждое действие в алгоритме и понимать для чего эти действия используется, в конечном итоге реализовать программу вычисление минимума методом половинного деления в среде программирования Delphi.

1. Задание

Изучить тему «Численные методы поиска минимум функции одной переменной. Метод половинного деления». Написать реализацию метода в среде разработки Delphi.

1. Постановка задачи

Для выполнения поставленной задачи необходимо выполнить следующие задачи:

* Разобрать метод половинного деления.
* Разобрать поиск минимума функции с помощью данного метода.
* Создание презентации для углубления знаний и демонстрации данного метода другим обучающимся.
* Реализация данного метода в среде Delphi и демонстрация его руководителю практики.

1. Сбор и анализ информации

**Описание основных понятий и терминов**.

Нелинейным уравнением - называется уравнения вида ɕ(x) = 0, где ɕ(x) некоторая нелинейная функция. Корнем уравнения н f(x)=0 называется такое значение c, что f(c)= 0. Теорема нелинейных уравнений: если функция f(x)

1. непрерывна на отрезке [a;b] вместе со своими производными первого и второго порядков;
2. значения функции на концах отрезка разных знаков: f(a)\*f(b) < 0;
3. первая и вторая производные сохраняют определенный знак на всем отрезке (то есть функция f(x) является монотонной на рассматриваемом отрезке),

то уравнение f(x)= 0 имеет единственное решение на отрезке [a;b].

Минимумом функции - называют точку на функции, в которой значение функции меньше, чем в соседних точках

Метод половинного деления один из методов решения нелинейных уравнений и основан на последовательном сужении интервала, содержащего единственный корень уравнения F(x)=0 до того времени, пока не будет достигнута заданная точность ɛ.

Пусть задан отрезок [а,b], содержащий один корень уравнения. Предварительно необходимо определить области локализации корней данного уравнения. Если на отрезке [а,b] содержится более одного корня, то метод не работает.

Метод половинного деления так же называется методом дихотомии.

**Описание работы алгоритма.**

Находится середина начального отрезка [a;b] – точка c = (a+b)/2.

Затем выбирается - ∂ точность к которой мы будем стремиться при вычислениях.

После находим x1 и x2:

Далее находим функции F(x1) и F(x2), если F(x1) > F(x2), то x ∈ [x1;b], и левая граница отрезка a переносится в точку x1, в противном случае иначе. Повторяем данные действия пока не достигнем условия:

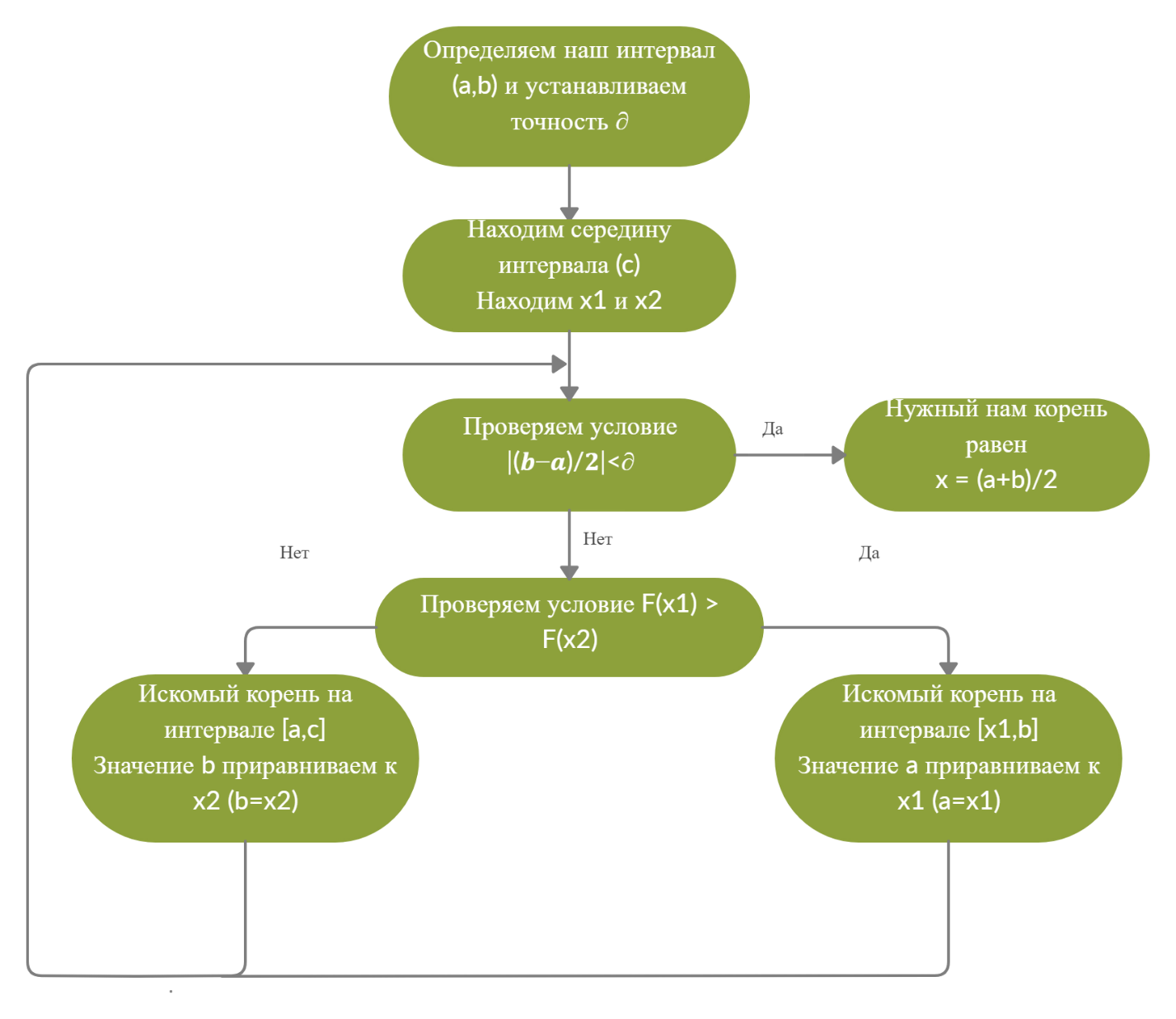
****

Рисунок 1 – Блок-схема алгоритма

1. Разработка программы

Для реализации данного метода, я сначала написал его в среде Pascal после того, как проверив вычисления и сравнив ответы программы, я проверил ещё раз алгоритм и после приступил к реализации данного метода в среде Delphi. Интерфейс программы состоит из надписи с нашей функцией в программе, также окна, где мы можем менять значения: левосторонний интервал a, правосторонний интервал b, точность e. Имеет кнопку «Найти» для запуска расчётов программы, а также два окна с выводом значения минимума и значения функции минимума и таблица с промежуточными итогами расчётов [1].

Список основных процедур и функций для работы программы

* Функция для округления до 10 знаков после запятой, используется для более удобного предоставления информации в таблице:

function roundto(x:real):real;

* Функция для которое содержит выражение, которое мы используем для расчётов в программе:

function f(x:real):real;

* Процедура, используемая для построения таблицы промежуточных расчётов в программе:

procedure table(var i:integer);

* Функция, используемая для нахождения минимума функции, используя выражение в программе:

function rootfind():real;

# Заключение

Был рассмотрен метод поиска минимума функции одной переменной, методом половинного деления. Были изучены основные понятия и термины данного метода. Был проведён анализ действий в алгоритме. Была написана программа в среде Delphi, которая имеет промежуточные итоги для удобства пользователя.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Метод половинного деления. - Текст: электронный // ЯКласс : [сайт]. –

URL : [https://cutt.ly/hpCztJ6](https://vk.com/away.php?utf=1&to=https%3A%2F%2Fcutt.ly%2FhpCztJ6). ‒ Дата обращения : 07.07.2020

1. Метод половинного деления (метод дихотомии). - Текст: электронный // Студфайл : [сайт]. – URL : [https://cutt.ly/xpCbHex](https://vk.com/away.php?utf=1&to=https%3A%2F%2Fcutt.ly%2FxpCbHex). ‒ Дата обращения : 07.07.2020
2. Метод половинного деления. Нахождение минимума. - Текст: электронный // Киберфорум : [сайт]. – URL : [https://cutt.ly/NpCb6xd](https://vk.com/away.php?utf=1&to=https%3A%2F%2Fcutt.ly%2FNpCb6xd). ‒ Дата обращения : 10.07.2020
3. Описание метода половинного деления. - Текст: электронный // Пикод : [сайт]. – URL : [http://pycode.ru/files/hw1.pdf](https://vk.com/away.php?to=http%3A%2F%2Fpycode.ru%2Ffiles%2Fhw1.pdf&cc_key=). ‒ Дата обращения : 07.07.2020

ПРИЛОЖЕНИЕ

Листинг 1 – Основная программа

|  |
| --- |
| **unit** Unit1;  **interface**  **uses**  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  Dialogs, StdCtrls, Buttons, Grids;  **type**  TForm1 = **class**(TForm)  edt1: TEdit;  edt2: TEdit;  edt3: TEdit;  btn1: TBitBtn;  a: TLabel;  lbl1: TLabel;  Label1: TLabel;  edt4: TEdit;  Label2: TLabel;  edt5: TEdit;  lbl2: TLabel;  lbl3: TLabel;  strngrd1: TStringGrid;  lbl4: TLabel;  **procedure** btn1Click(Sender: TObject);  **private**  { Private declarations }  **public**  { Public declarations }  **end**;  **var**  Form1: TForm1;  **implementation**  {$R \*.dfm}  **procedure** TForm1.btn1Click(Sender: TObject);  **var** a,b,x1,x2,e:real; i:integer;  **function** roundto(x:real):real;  **begin**  roundto:=round(x\*10000000000)/10000000000;  **end**;  **procedure** table(**var** i:integer);  **begin**  strngrd1.Cells[0,i]:=IntToStr(i);  strngrd1.Cells[1,i]:=FloatToStr(roundto(a));  strngrd1.Cells[2,i]:=FloatToStr(roundto(b));  strngrd1.Cells[3,i]:=FloatToStr(roundto(x1));  strngrd1.Cells[4,i]:=FloatToStr(roundto(x2));  strngrd1.Cells[5,i]:=FloatToStr(roundto(f(x1)));  strngrd1.Cells[6,i]:=FloatToStr(roundto(f(x2)));  **if** (Abs((b-a)/2) < e) **then**  strngrd1.Cells[7,i]:='Да'  **else** |

Продолжение листинга 1

|  |
| --- |
| strngrd1.Cells[7,i]:='Нет';  strngrd1.Cells[8,i]:=FloatToStr(roundto(Abs((b-a)/2)));  strngrd1.RowCount:=i+1;  **end**;  **function** rootfind():real;  **begin**  e:=strtofloat(edt3.text);  x1:=((a+b)/2)-e/2;  x2:=((a+b)/2)+e/2;  table(i);  **while** (abs((b-a)/2) > e) **do begin**  inc(i);  x1:=((a+b)/2)-e/2;  x2:=((a+b)/2)+e/2;  **if** f(x1) > f(x2) **then**  a:=x1  **else**  b:=x2;  table(i);  **end**;  rootfind:=(a+b)/2;  **end**;  **begin**  a:=strtofloat(edt1.text);  b:=strtofloat(edt2.text);  i:=0;  edt4.text:='';  edt5.text:='';  **if** a > b **then**  edt4.text:='Введите корректный интервал a<b'  **else if** (edt3.text[2]='.' ) **then**  edt4.text:='Точность нужно вводить через запятую'  **else if** (strtofloat(edt3.text) > 1) **then**  edt4.text:='Введите корректную точность e < 1'  **else begin**  edt4.text:=floattostr(rootfind());  edt5.text:=floattostr(f(rootfind()));  **end**;  **end**;  **end**. |

Протокол выполнения программы

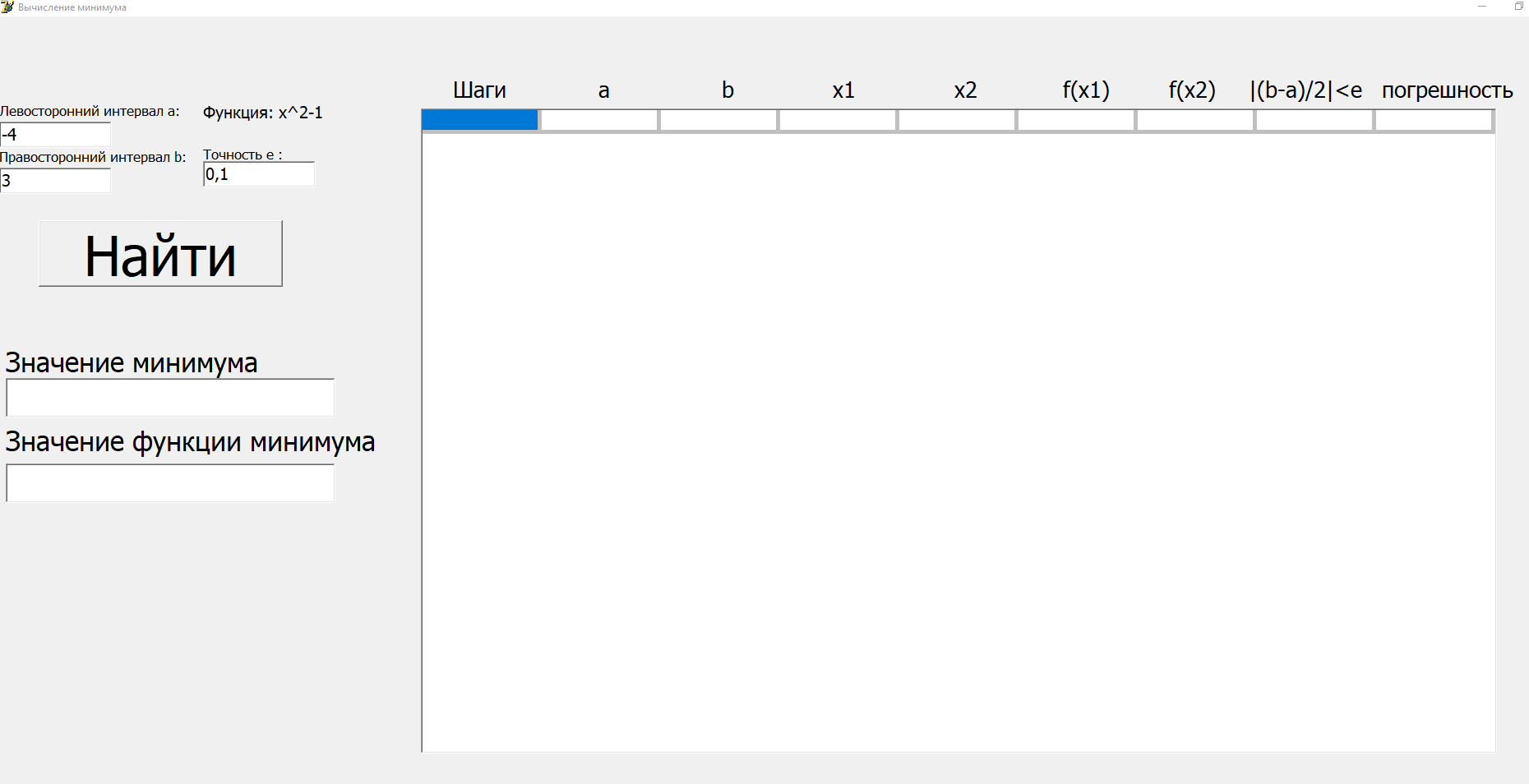


Рисунок 2 – Меню программы

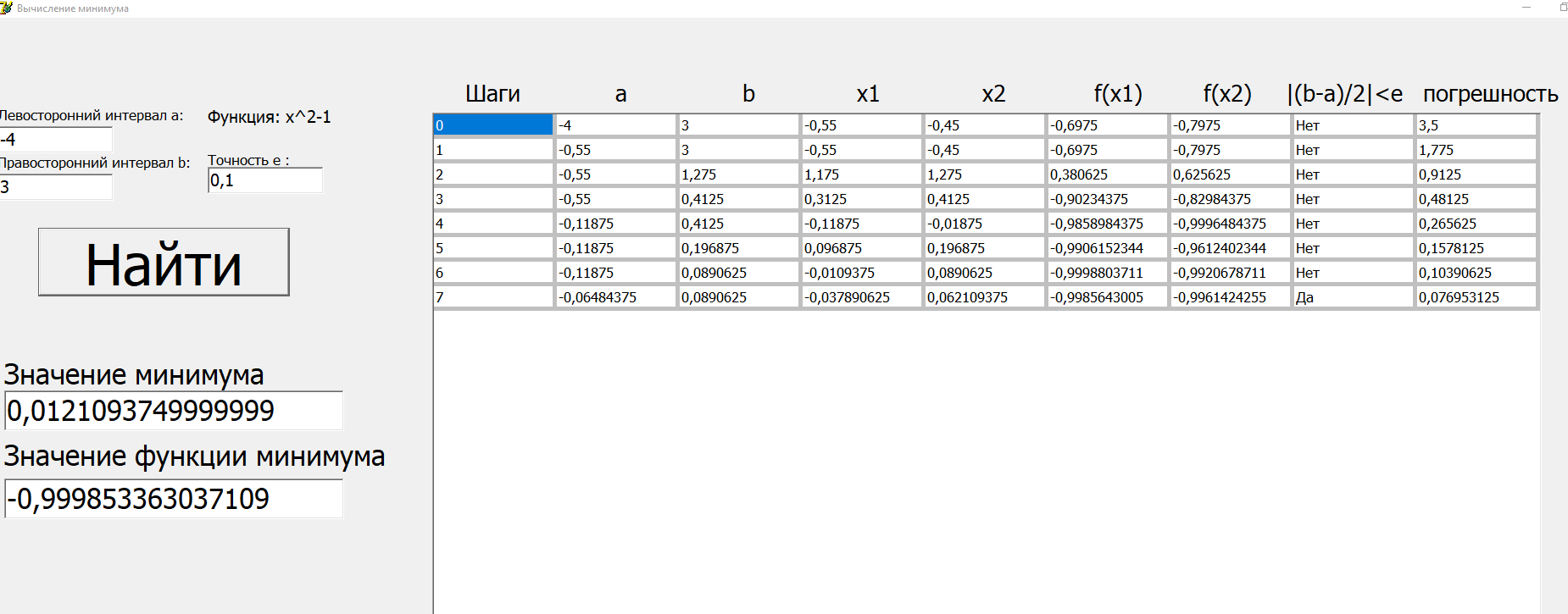


Рисунок 3 – Выполнение программы