МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное бюджетное

образовательное учреждение

высшего образования

ВЯТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФАКУЛЬТЕТ КОМПЬЮТЕРНЫХ И ФИЗИКО-МАТЕМАТИЧЕСКИХ НАУК

КАФЕДРА ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ №1

**по дисциплине**

**“ Методы оптимизации и исследование”**

**Методы поиска экстремума одномерной функции**

Выполнил: студент гр. ФИб-3302-51-00 Ситников Д. А. \_\_\_\_\_\_\_\_\_\_\_

Проверил: доцент кафедры ПМиИ Чупраков П. Г. \_\_\_\_\_\_\_\_\_\_\_\_

Киров 2020

**Цель лабораторной работы.**

Изучить различные способы нахождения экстремумов функции на промежутке, сделать програмную реализацию, применив эти алгоритмы.

**Задание.**

1. Найти минимум произвольной функции f(x) на промежутке [a,b] тремя

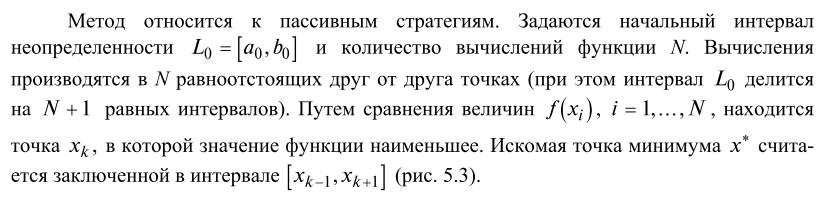
различными методами, сравнить полученные результаты. Предусмотреть

непосредственное задание промежутка [a,b] и эвристический поиск.

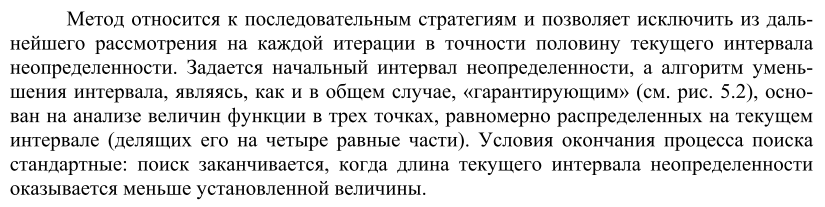
2. Проверить решения с помощью математических пакетов.

**Краткое описание выбранных методов.**

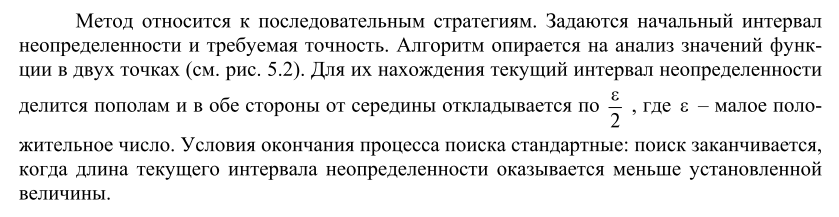
Для поиска экстремумов функции на промежутке будут испольоваться 3 следующих алгоритма:

*Метод равномерного поиска*

*Метод деления пополам*

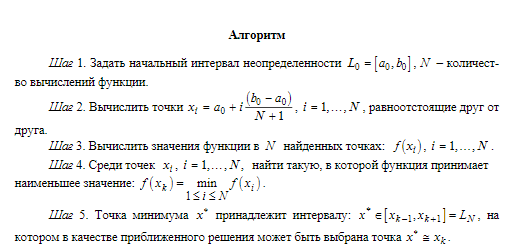
**

*Метод дихтомии*

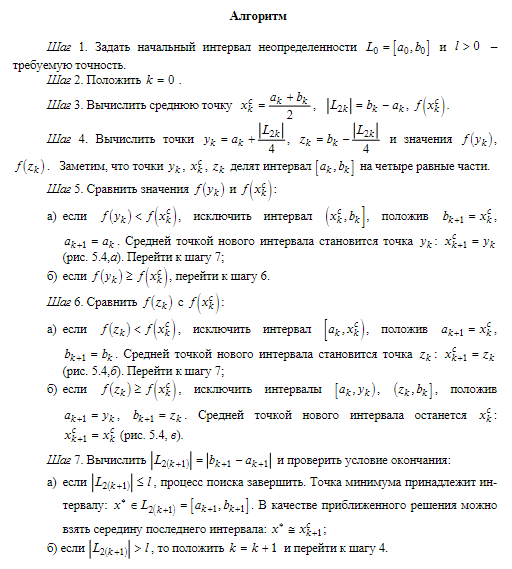


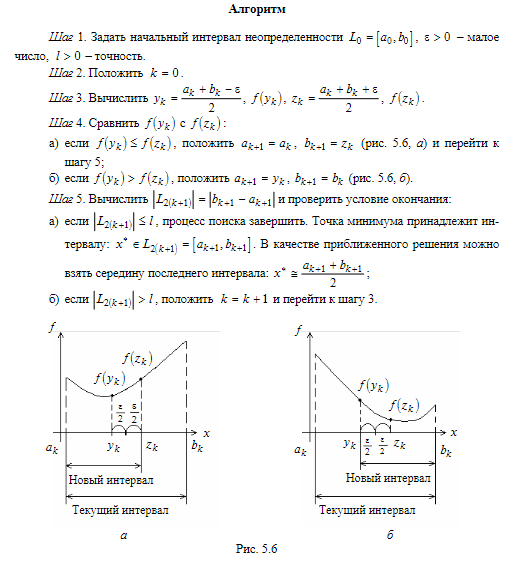
**Схемы алгоритмов**.

Метод равномерного поиска



Метод деления пополам



Метод Дихтомии  


**Листинг программы с комментариями.**

**Метод равномерного поиска**

private static void Ravnomernyi(double a, double b, double n)

{

double dx = (b - a) / n;

double yMin = double.PositiveInfinity;

double xMin = a;

for (double x = a; x <= b; x += dx)

{

double y = f(x);

if (y < yMin)

{

xMin = x;

yMin = y;

}

}

Console.Write("Аргумент функции: ");

Console.WriteLine(xMin);

Console.Write("Значение функции: ");

Console.WriteLine(yMin);

Console.WriteLine();

}

**Метод деления пополам**

private static void deleniepopolam(double a, double b, double n, double eps)

{

int count = 0;

double x, xm, L, x1, x2;

xm = (a + b) / 2;

L = b - a;

x1 = a + L / 4;

x2 = b - L / 4;

while (L >= eps)

{

count++;

if (f(x1) < f(xm))

{

b = xm;

xm = x1;

}

else

{

if (f(x2) < f(xm))

{

a = xm;

xm = x2;

}

if (f(x2) > f(xm))

{

a = x1;

b = x2;

}

}

L = b - a;

x1 = a + L / 4;

x2 = b - L / 4;

x = xm;

}

Console.WriteLine("Аргумент функции: " + xm);

Console.WriteLine("Значение функции: " + f(xm));

Console.WriteLine("Количество итераций: " + count);

Console.WriteLine();

}

**Метод Дихтомии**

private static void dichotomy(double a, double b, double n, double eps)

{

int count = 0;

double L = b - a;

double x\_m;

double x1;

double x2;

do

{

count++;

x\_m = (a + b) / 2;

x1 = a + L / 4;

x2 = b - L / 4;

if (f(x1) < f(x\_m))

{

b = x\_m;

}

else

{

if (f(x2) < f(x\_m))

{

a = x\_m;

}

else

{

a = x1;

b = x2;

}

}

L = b - a;

}

while (L > eps);

Console.WriteLine("Аргумент функции: " + x\_m);

Console.WriteLine("Значение функции: " + f(x\_m));

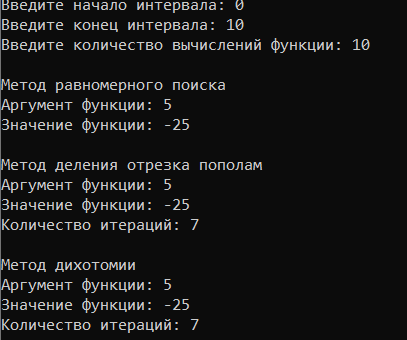
Console.WriteLine("Количество итераций: " + count);

Console.ReadLine();

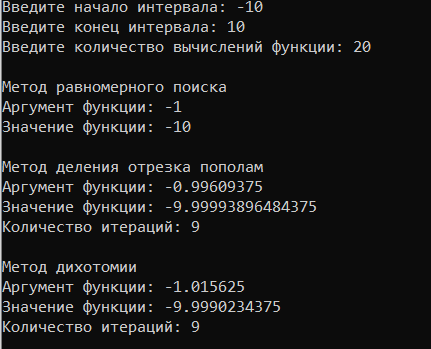
}

**Результаты выполнения программы.**

Тест 1

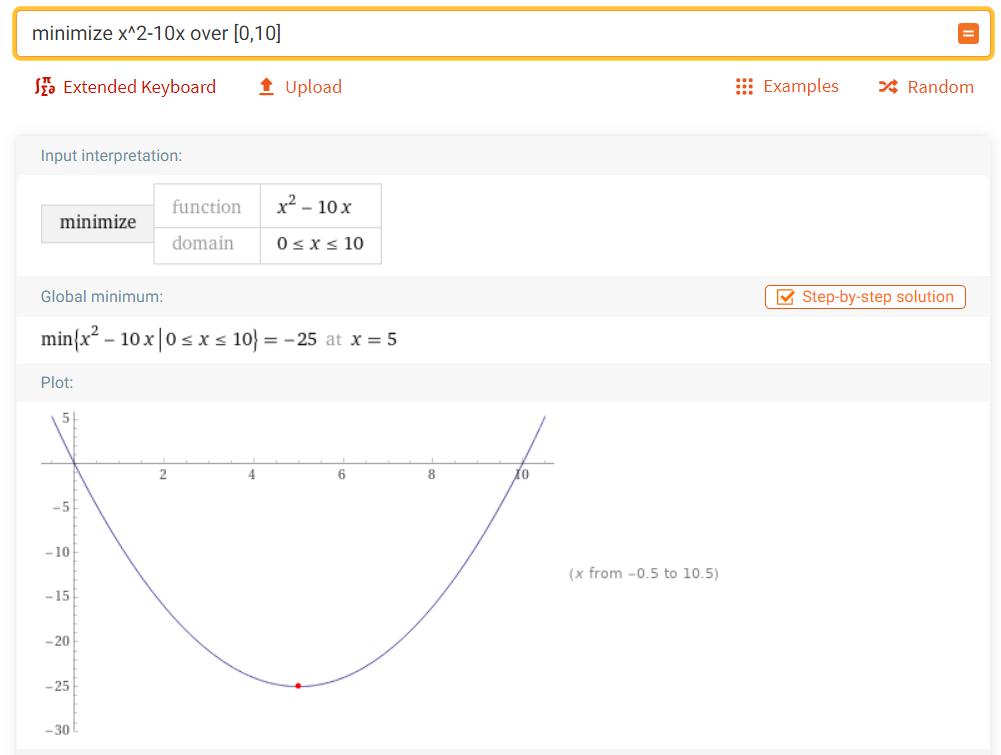


Тест 2

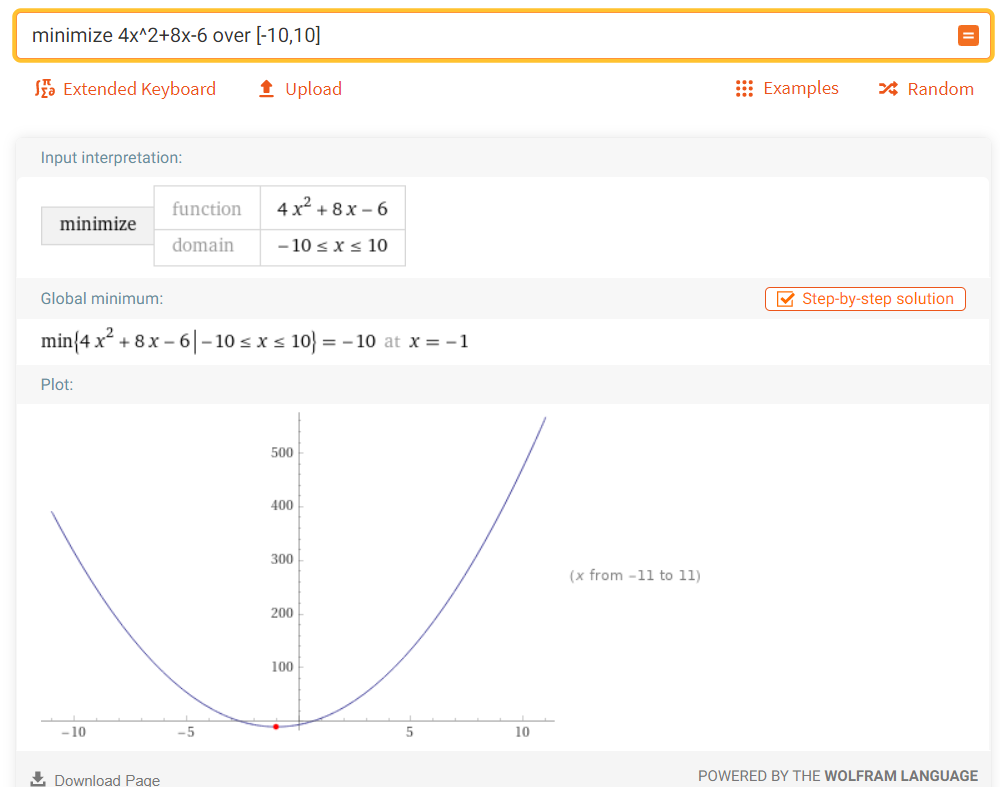


**Проверка вычислений в математическом пакете.**

Тест 1



Тест 2



**Вывод**

В ходе выполнения лабораторной работы, были изучены методы нахождения экстремумов, а так же реализована програмная реализация. Метод деления пополам, и метод дихтомии имеют логарифмическое время выполнения, и являются наиболее быстрыми и эффективными в сравнение с рвномерным поиском который выполняется за линейное время.