МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

# «КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ» (ФГБОУ ВО «КубГУ»)

**Факультет компьютерных технологий и прикладной математики Кафедра прикладной математики**

# ОТЧЕТ

**по дисциплине**

# «Методы оптимизации»

Работу выполнил М. А. Пузырёв

Работу принял преподаватель Е. С. Троценко

Краснодар 2024

# Постановка задачи

Требуется найти безусловный минимум функции 𝑓(𝑥) = x2 + 2 одной переменной методом дихотомии на промежутке [−3; 7], с заданными

𝜀 = 0,2 и 𝑙 = 0,5 , т.е. такую точку 𝑥∗ ∈ 𝑅, что 𝑓(𝑥∗) = min 𝑓(𝑥), x *R*.

# Стратегия поиска

Метод относится к последовательным стратегиям. Задаётся начальный интервал неопределённости и требуемая точность. Алгоритм опирается на анализ значений функции в двух точках. Для их нахождения текущий интервал неопределённости делится пополам и в обе стороны от середины откладывается

по , где 𝜀 – малое положительное число. Условия окончания процесса поиска

стандартные: поиск заканчивается, когда длина текущего интервала неопределённости оказывается меньше установленной величины.

# Алгоритм

Шаг. 1 Задать начальный интервал неопределённости 𝐿0 = [𝑎0, 𝑏0], 𝜀 > 0 − малое число, 𝑙 > 0 − точность.

Шаг. 2 Положить 𝑘 = 0.

Шаг. 3 Вычислить yk = , zk = , 𝑓(yk), 𝑓(zk).

Шаг. 4 Сравнить 𝑓(𝑦𝑘) c 𝑓(𝑧𝑘):

а) если 𝑓(𝑦𝑘) ≤ 𝑓(𝑧𝑘), положить 𝑎𝑘+1 = 𝑎𝑘 , 𝑏𝑘+1 = 𝑧𝑘 и перейти к шагу

5;

б) если 𝑓(𝑦𝑘) > 𝑓(𝑧𝑘), положить 𝑎𝑘+1 = 𝑦𝑘 , 𝑏𝑘+1 = 𝑏𝑘.

Шаг. 5 Вычислить |𝐿2(𝑘+1)| = |𝑏𝑘+1 − 𝑎𝑘+1| и проверить условие

окончания:

а) если |𝐿2(𝑘+1)| ≤ 𝑙 , процесс поиска завершается и 𝑥∗ ∈ 𝐿2(𝑘+1) = [𝑎𝑘+1, 𝑏𝑘+1] . В качестве приближенного решения можно взять середину

последнего интервала: 𝑥∗ = ;

б) если |𝐿2(𝑘+1)| > 𝑙 , положить 𝑘 = 𝑘 + 1 и перейти к шагу 3.

# Код программы

# Реализация кода представлена на языке C++:

#include <iostream>

#include <cmath>

#include <math.h>

using namespace std;

double F(double x) {

return pow(x, 2) + 2;

}

int main() {

setlocale(LC\_ALL, "rus");

int it = 0;

double a = -3.0;

double b = 7.0;

double eps = 0.2;

double l = 0.5;

double x, y;

do {

x = (a + b - eps) / 2;

y = (a + b + eps) / 2;

if (F(x) <= F(y))

b = y;

else

a = x;

it++;

} while (fabs(b - a) > l);

double min = (a + b) / 2;

cout << "Искомый минимум функции: " << min << endl;

cout << "Значение в минимуме: " << F(min) << endl;

cout << "Сходимость заданной функции: " << 1 / pow(2, (it / 2)) << endl;

return 0;

}

# Сходимость

Для метода дихотомии характеристика относительного уменьшения

начального интервала неопределённости находится по формуле 𝑅(𝑁) = 1

𝑁

22

, где

𝑁 − количество вычислений функции.

Для функции 𝑓(𝑥) = 𝑥2 + 2\*x при заданном малом числе и точности

количество вычислений функции 𝑁 = 6. Тогда сходимость будет равна 0,125.

# Вывод

Для решения задачи по поиску безусловного минимума функции

𝑓(𝑥) = 𝑥2 + 2\*x

методом дихотомии была написана программа на языке

C++. Результатом работы программы является вычисленное минимальное значение функции, равное 2.00451 в точке минимума, равной -0.0671875, при этом сходимость равна 0,125 (рис.1). При изменении параметров точности и малого числа, 𝑙 и 𝜀 соответственно, точка минимума, значение функции в ней и сходимость меняются. При увеличении 𝜀 либо при уменьшении 𝑙 число

итераций, необходимых для нахождения безусловного минимума становится больше, следовательно сходимость уменьшается.

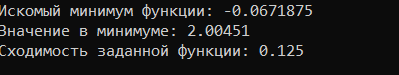


Рисунок 1 – Результат работы программы