



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н. Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н. Э. Баумана)

ФАКУЛЬТЕТ «Информатика, искусственный интеллект и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

ОТЧЕТ

по лабораторной работе № 4
по курсу «Методы вычислений»
на тему: «Метод Ньютона»
Вариант № 15

Студент ИУ7-21М
(Группа)

(Подпись, дата)

Миронов Г. А.
(И. О. Фамилия)

Преподаватель

(Подпись, дата)

Власов П. А.
(И. О. Фамилия)

2024 г.

1 Выполнение индивидуального задания

1.1 Цель работы

Изучение метода Ньютона для решения задачи одномерной оптимизации.

1.2 Постановка задачи

Необходимо:

1. реализовать модифицированный метод Ньютона с конечно-разностной аппроксимацией производных в виде программы на ЭВМ.
2. провести решение задачи

$$\begin{cases} f(x) \rightarrow \min \\ x \in [a, b] \end{cases}$$

для данных индивидуального варианта для лабораторной работы № 1.

3. организовать вывод на экран графика целевой функции, найденной точки минимума $(x^*, f(x^*))$ и последовательности точек $(x_i, f(x_i))$, аппроксимирующих точку искомого минимума (для последовательности точек следует предусмотреть возможность "отключения" вывода ее на экран).
4. провести решение задачи с использованием стандартной функции **fminbnd** пакета MatLAB.

Индивидуальный вариант целевой функции:

$$\sinh\left(\frac{3x^4 - x + \sqrt{17} - 3}{2}\right) + \sin\left(\frac{5^{1/3}x^3 - 5^{1/3}x + 1 - 4 * 5^{1/3}}{-x^3 + x + 2}\right),$$

при $[a, b] = [0, 1]$.

Метод Ньютона

Основная идея метода Ньютона: за очередное приближение корня уравнения принимается точка пересечения касательной к графику функции в точке, отвечающей текущему приближению, с осью OX .

Расчётное соотношение имеет вид:

$$x_{i+1} = x_i - \frac{g(x_i)}{g'(x_i)}.$$

Необходимо реализовать модифицированный метод Ньютона, использующий конечно-разностные аппроксимации вместо первой и второй производных:

$$\begin{cases} f'(x_i) \approx \frac{f(x_i+h)-f(x_i-h)}{2h} \\ f''(x_i) \approx \frac{f(x_i+h)-2f(x_i)+f(x_i-h)}{h^2}, \end{cases}$$

где h — достаточно малая величина.

Условие окончания итераций:

$$\begin{cases} |x_{i+1} - x_i| < \epsilon, \\ |g(x_i)| < \epsilon. \end{cases}$$

1.3 Схема алгоритма

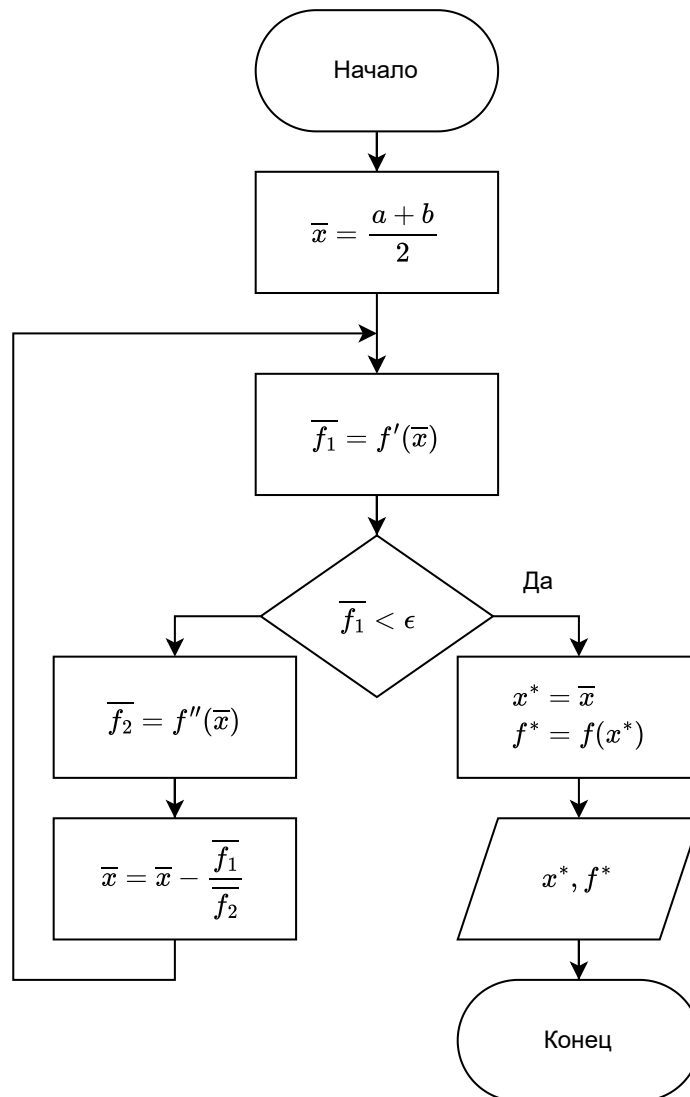


Рисунок 1.1 – Схема алгоритма

1.4 Текст программы

Листинг 1.1 – Файл main.m

```
function lab04()
    clc();
    clf();

    [debugFlg, delayS] = deal(1, 0.5);
    [a, b] = deal(0, 1);
    [eps, h] = deal(1e-2, 1e-3);

    fplot(@f, [a, b]);
```

```

hold on;

% newton_method(a, b, eps, h, debugFlg, delayS);

modified_newton_method(a, b, eps, h, debugFlg, delayS);

if debugFlg
    [x, fval] = fminbnd(@f, a, b);
    fprintf('fminbnd: x=%.10f, f(x)=%.10f\n', x, fval);
end

legend("off");
end

function modified_newton_method(a, b, h, eps, debugFlg, delayS)
    x = (a + b) / 2;

    [f_inc, f_x, f_dec] = deal(f(x + h), f(x), f(x - h));
    [f1, f2] = deal((f_inc - f_dec) / (2 * h), (f_inc - 2 * f_x
        + f_dec) / (h^2));

    i = 1;
    if debugFlg
        print_iter(i, x, f_x, f1, delayS);
    end

    while abs(f1) >= eps
        i = i + 1;

        [f_inc, f_x, f_dec] = deal(f(x + h), f(x), f(x - h));

        f1 = (f_inc - f_dec) / (2 * h);

        if debugFlg
            print_iter(i, x, f_x, f1, delayS);
        end

        x = x - f1 / f2;
    end;

    if debugFlg

```

```

        fprintf('RESULT: %2d iterations: x=%.10f, f(x)=%.10f\n',
            i, x, f_x);
        scatter(x, f_x, 'r', 'filled');
    end
end

function newton_method(a, b, h, eps, debugFlg, delayS)
    x = (a + b) / 2;

    i = 0;

    do
        i = i + 1;

        [f_inc, f_x, f_dec] = deal(f(x + h), f(x), f(x - h));

        [f1, f2] = deal((f_inc - f_dec) / (2 * h), (f_inc - 2 *
            f_x + f_dec) / (h^2));

        if debugFlg
            print_iter(i, x, f_x, f1, delayS);
        end

        x_prev = x;
        x = x_prev - f1 / f2;
    until abs(x - x_prev) < eps;

    if debugFlg
        fprintf('RESULT: %2d iterations: x=%.10f, f(x)=%.10f\n',
            i, x, f_x);
        scatter(x, f_x, 'r', 'filled');
    end
end

function print_iter(i, x, f_x, f1, delayS)
    fprintf(" %2d:\t x = %.10f, f(x) = %.10f, f\'(x) = %.10f\n", i, x, f_x, f1);
    plot(x, f_x, 'xk');
    hold on;
    pause(delayS);
end

```

```

function y = f(x)
    k = power(5,1/3);

    y = sinh((3 * power(x,4) - x + sqrt(17) - 3) / 2) + sin((k *
        power(x, 3) - k * x + 1 - 2 * k) ./ (-power(x,3) + x +
        2));
end

```

1.5 Результаты расчетов для задачи из индивидуального варианта.

Таблица 1.1 – Результаты расчетов

№ п/п	ϵ	N	x^*	$f(x^*)$
1	10^{-2}	2	0.4381262644	-0.5511546082
2	10^{-4}	7	0.4423213847	-0.5511898772
3	10^{-6}	12	0.4423638093	-0.5511898808

1.6 Сводная таблица.

Таблица 1.2 – Сводная таблица, обобщающая вычисления из лабораторных работ №№1–4

ϵ	№ п/п	Метод	N	x^*	$f(x^*)$
10^{-2}	1	поразрядного поиска	19	0.4414062500	-0.5511880697
	2	золотого сечения	11	0.4442719100	-0.5511826696
	3	парабол	2	0.4381262644	-0.5511546082
	4	Ньютона	3	0.4422911664	-0.5511898803
	5	Ньютона модифицированный	6	0.4423048929	-0.5511898806
10^{-4}	1	поразрядного поиска	35	0.4423828125	-0.5511898802
	2	золотого сечения	20	0.4423525313	-0.5511898806
	3	парабол	7	0.4423213847	-0.5511898772
	4	Ньютона	3	0.4423642794	-0.5511898645
	5	Ньютона модифицированный	6	0.4423779335	-0.5511898731
10^{-6}	1	поразрядного поиска	49	0.4423646927	-0.5511898808
	2	золотого сечения	30	0.4423640182	-0.5511898808
	3	парабол	12	0.4423638093	-0.5511898808
	4	Ньютона	3	0.4423642888	-0.5511898646
	5	Ньютона модифицированный	6	0.4423779361	-0.5511898731
	6	функция fminbnd	—	0.4423513587	-0.5511898805