

Московский государственный технический университет  
им. Н.Э. Баумана

---

Лабораторная работа № 1 по курсу  
«Методы вычислений»  
**«Метод поразрядного поиска»**

Работу выполнил  
студент группы ИУ7-23М  
Кадыров Руслан

Москва, 2018г.

## Цель работы

Изучение метода поразрядного поиска для решения задачи одномерной минимизации.

## Постановка задачи

1. Реализовать метод поразрядного поиска в виде программы на ЭВМ;
2. Провести решение задачи

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

для индивидуального варианта;

3. Организовать вывод на экран графика целевой функции, найденной точки минимума  $(x^*, f(x^*))$  и последовательности точек  $(x_i, f(x_i))$ , приближающих точку искомого минимума.

Исходные данные задачи: вариант №6

Целевая функция

$$\operatorname{ch}\left(\frac{3x^3 + 2x^2 - 4x + 5}{3}\right) + \operatorname{th}\left(\frac{x^3 - 3\sqrt{2}x - 2}{2x + \sqrt{2}}\right) - 2.5$$

на отрезке  $[0,1]$ ;

## Описание метода поразрядного поиска

Метод поразрядного поиска – это усовершенствованный метод перебора с точки зрения уменьшения количества вычислений целевой функции. В рамках данного метода необходимо сначала найти грубое приближение  $x^*$  с достаточно большим шагом, а затем уточнить это значение с более мелким шагом, рассматривая лишь точки из некоторой окрестности найденного приближения. Приведем алгоритм метода поразрядного поиска.

**Шаг 1.** Выбрать начальный шаг  $\Delta = (b - a)/4$ , положить  $x_0 = a$ , вычислить  $f(x_0)$ .

**Шаг 2.** Положить  $x_1 = x_0 + \Delta$ , вычислить  $f(x_1)$ .

**Шаг 3.** Сравнить  $f(x_0)$  и  $f(x_1)$ . Если  $f(x_0) > f(x_1)$ , то переходим к шагу 4, иначе – к шагу 5.

**Шаг 4.** Положить  $x_0 = x_1$  и  $f(x_0) = f(x_1)$ , проверить условия  $a < x_0 < b$ . Если они выполнены, перейти к шагу 2, иначе – к шагу 5.

**Шаг 5.** Проверка на окончание поиска. Если  $|\Delta| \leq \varepsilon$ , то вычисления завершить. Положить  $x^* = x_0$ ,  $f^* = f(x^*)$ , иначе перейти к шагу 6.

**Шаг 6.** Изменение направления и шага поиска. Положить  $x_0 = x_1$ ,  $x_1 = x_0 + \Delta$ ,  $\Delta = -\Delta/4$ . Перейти к шагу 2.

## Текст программы

Листинг скрипта MetodyVich\_Lab\_1.m

```
eps = 0.01;  
a = 0;  
b = 1;  
[xRes, fRes, xi, fi, iterCount] = BitwiseSearch(a,b, eps);
```

```

%Получение данных для построения графика целевой функции
xArr = zeros(1,iterCount);
fArr = zeros(1,iterCount);
a = -4;
step = 1;
while(step < 600)
    xArr(step) = step*0.01 + a;
    fArr(step) = Func(step*0.01 + a);
    step = step + 1;
end

%Получение данных для построения точек, приближающихся к минимуму
xiArr = zeros(1,iterCount);
fiArr = zeros(1,iterCount);

step = 1;
for i=1:50
    if(xi(i)~=0)
        xiArr(step) = xi(i);
        fiArr(step) = fi(i);
        step = step + 1;
    end
end

fprintf('Количество вычислений целевой функции: %1.d.\n',iterCount);
fprintf('x* = %1.10f.\n',xRes);
fprintf('f(x*) = %1.10f.\n',fRes);

plot(xArr, fArr,xRes, fRes, 'ro', xiArr, fiArr, 'b*');
grid on;
title('График целевой функции f(x)');
xlabel('x');
ylabel('Значение целевой функции f(x)');
ylim([-2 15])

%Вычисление значения целевой функции в точке x
function X = Func(x)
firstPart = cosh((3*(x^3) + 2*(x^2) - 4*x + 5)/3);
secondPart = tanh((x^3 - 3*sqrt(2)*x -2)/(2*x + sqrt(2)));
X = firstPart + secondPart - 2.5;
%X = cosh((3*(x^3)+2*(x^2)-4*x+5)/3)+tanh((x^3-3*sqrt(2)*x-2)/(2*x+sqrt(2)))-2.5;
end

%Метод поразрядного поиска
function [xResult, fResult, xI, fI, iterationCount] = BitwiseSearch(a,
b, eps)
% a - начало отрезка, b - конец отрезка, eps - эпсилон, точность
поиска
% xResult - оптимальный x*, %fResult - значение целевой функции в x*
% xI-последовательность xi, приближающих точку искомого минимума
% fI-последовательность fi, приближающих точку искомого минимума
% iterationCount - число вычислений значения целевой функции

xi = zeros(1,50);
fi = zeros(1,50);

```

```

delta = (b - a)/4;
x0 = a;
f0 = Func(x0);
iter = 0;
while(true)
    iter = iter + 1;
    x1 = x0 + delta;
    f1 = Func(x1);
    xi(iter) = x1;
    fi(iter) = f1;

    if(f0 > f1)
        x0 = x1;
        f0 = f1;
        if(~(a < x0 && x0 < b))
            if(abs(delta) <= eps)
                xResult = x0;
                fResult = f0;
                break;
            else
                x0 = x1;
                f0 = f1;
                delta = - (delta/4);
            end
        end
    else
        if(abs(delta) <= eps)
            xResult = x0;
            fResult = f0;
            break;
        else
            x0 = x1;
            f0 = f1;
            delta = - (delta/4);
        end
    end
end
iterationCount = iter;
xI = xi;
fI = fi;
end

```

### Результаты расчета задачи индивидуального варианта

№ п/п	$\varepsilon$	$N$	$x^*$	$f(x^*)$
1	$10^{-2}$	18	0.4804687500	-1.4738794316
2	$10^{-4}$	35	0.4824218750	-1.4738932843
3	$10^{-6}$	49	0.4824180603	-1.4738932844