|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **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 и системы управления\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

КАФЕДРА \_\_\_\_\_\_\_Программное обеспечение ЭВМ и информационные технологии\_\_\_\_\_\_\_\_

**ОТЧЕТ**

**по Лабораторной работе №1**

**по курсу**

**«Методы вычислений»**

**Тема**

**«Метод поразрядного поиска»**

**Вариант 9**

Студент \_\_\_ИУ7-21М\_\_\_\_  **\_\_\_**Карпухин А.С.\_\_\_

(Группа) (И.О.Фамилия)

Преподаватель  **\_\_\_\_**Власов П.А.**\_\_\_\_\_\_**

(И.О.Фамилия)

1. *г.*
2. **Постановка задачи**

Общий вид задачи оптимизации:

где , .

Если в данной задаче , то соответствующая задача называется задачей одномерной оптимизации.

Задача одномерной минимизации имеет вид:

1. **Исходные данные варианта**

Целевая функция *f*:

a = 0, b = 1.

1. **Метод поразрядного поиска**

Метод поразрядного поиска является усовершенствованием метода перебора с целью минимизации количества обращений к целевой функции. В основе метода лежит идея грубого определения локализации искомой точки минимума с ее последующим уточнением.

Схема алгоритма поразрядного поиска приведена ниже на рисунке 1.

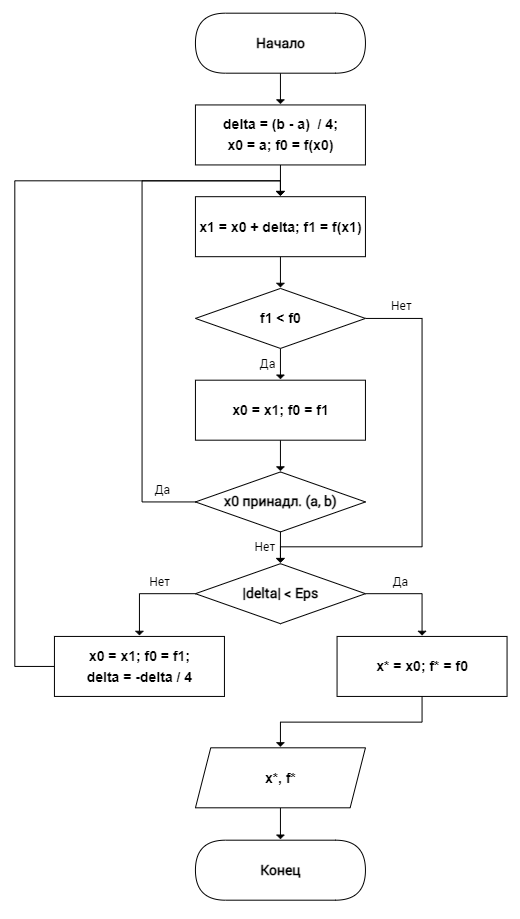


Рисунок 1 – Схема алгоритма поразрядного поиска.

1. **Текст программы**

Код разработанной программной реализации метода поразрядного поиска приведен в листинге 1.

Листинг 1. Код программы, реализующей метод поразрядного поиска

|  |
| --- |
| function lab1()  f = @(x) (tan((x.^4.0 + 2.0.\*x.^2.0 - 2.0.\*x + 2.0^(1.0/2.0) + 1.0) ./ 8.0) + ...  sin((4.0.\*x.^3.0 - 7.0.\*x - 9.0) ./ (20.0.\*x + 28.0)));    a = 0; b = 1;  X = a:0.01:b;  Y = f(X);  eps = [0.01, 0.0001, 0.000001];    fprintf("\nРезультаты вычисления точки минимума\n");  fprintf("для различных значений точности:\n\n");  fprintf(" # | Eps | N | x\* | f(x\*)\n");  fprintf("---|----------|----|---------|--------\n");    figure('Units', 'normalized', 'OuterPosition', [0 0 1 1]);  title('Метод поразрядного поиска');  tiledlayout(2, 2);    for i = 1:length(eps)  % Вычисление точки минимума и минимума функции  [X0, F0, X\_S, F\_S] = BitwiseSearch(0, 1, f, eps(i));    % Вывод строки таблицы результатов вычислений  fprintf("%2i | %5f | %2i | %5.5f | %5.5f\n", ...  i, eps(i), length(X\_S), X0, F0);    % Вывод графика для данной точности  ax = nexttile;  % Целевая функция  plot(ax, X, Y, '-b','LineWidth',1.5);  hold on;  % Последовательность приближений  plot(ax, X\_S, F\_S, '\*g','LineWidth', 2);  % Точка минимума  plot(ax, X0, F0, '\*r','LineWidth', 4);  title(ax, sprintf("Точность Eps = %2.0e", eps(i)));  legend('Целевая функция', 'Последовательность приближений', ...  'Точка минимума');  end    end    % Метод поразрядного поиска  function [X, F, X\_S, F\_S] = BitwiseSearch(a, b, f, eps)  arguments  a double % Левая граница отрезка  b double % Левая граница отрезка  f function\_handle % Целевая функция  eps double % Точность  end    delta = (b - a) / 4.0; % Шаг поиска    x0 = a; % Начальное приближение точки минимума  f0 = f(x0); % Начальное приближение минимума функции    X\_S = [x0]; % Массив всех приближений точки минимума  F\_S = [f0]; % Массив всех приближений минимума функции    x1 = x0;  f1 = f0;    while abs(delta) > eps  X\_S = [X\_S, x1];  F\_S = [F\_S, f1];    x0 = x1;  f0 = f1;    x1 = x0 + delta;  f1 = f(x1);    if (f1 < f0 && x1 > a && x1 < b)  continue;  end    delta = -delta / 4.0;  end    X = x1;  F = f1;  end |

1. **Результаты расчётов**

Результаты расчетов для задачи одномерной минимизации в соответствии с индивидуальным вариантом для различных значений *ε* приведены ниже в таблице 1.

Таблица 1. Результаты расчетов для задачи индивидуального варианта.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *№* | *ε* | *N* | *x\** | *f(x\*)* |
| 1 | 10-2 | 16 | 0.40625 | -0.06508 |
| 2 | 10-4 | 34 | 0.38354 | -0.06533 |
| 3 | 10-6 | 49 | 0.38379 | -0.06533 |