|  |  |
| --- | --- |
| 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 №2**

**по курсу**

**«Методы вычислений»**

**Тема**

**«Метод золотого сечения»**

**Вариант 10**

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 золотого сечения**

Метод золотого сечения построен так, чтобы уменьшить число обращений к целевой функции. Для этого пробные точки на каждой итерации выбираются таким образом, чтобы одна из них стала пробной точкой на следующей итерации. Схема алгоритма метода золотого сечения приведена на рисунке 1.

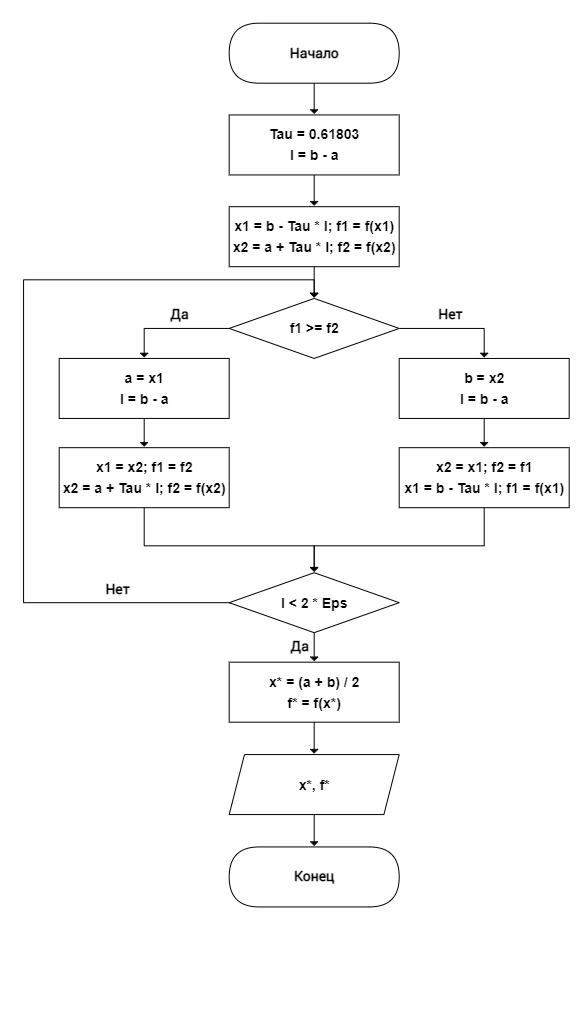


Рисунок 1 – Схема метода золотого сечения.

1. **Текст программы**

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

Листинг 1. Код класса, реализующего метод поразрядного поиска

|  |
| --- |
| classdef BitwiseSearch  properties(Constant)  Eps (1,1) double = 0.0001;  end    methods(Static)  function [X, F, X\_S, F\_S] = Solve(a, b, f)  arguments  a (1,1) double  b (1,1) double  f function\_handle  end    X\_S = [];  F\_S = [];    delta = (b - a) / 4.0;  x0 = a; f0 = f(x0);  x1 = x0; f1 = f0;    while abs(delta) > BitwiseSearch.Eps  X\_S = [X\_S, x0];  F\_S = [F\_S, f0];    x0 = x1; f0 = f1;  x1 = x0 + delta;  f1 = f(x1);    if (f1 < f0)  if (x1 > a && x1 < b)  continue;  end  end    delta = -delta / 4.0;  end    X = x0; F = f0;  end  end  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 | 0.7031 | -0.4652 |
| 2 | 10-4 | 26 | 0.7056 | -0.4653 |
| 3 | 10-6 | 40 | 0.7055 | -0.4653 |