

Лабораторная работа 1

Численное решение алгебраических и трансцендентных уравнений

Цель работы: изучить методы решения нелинейных уравнений, алгоритм каждого метода, формулы для вычисления и условия выбора начальных приближений, написать программу на языке программирования для реализации данных методов.

К методам решения нелинейного уравнения относят метод половинного деления, метод простых итераций, метод касательных и метод хорд. Рассмотрим алгоритм каждого из перечисленных методов.

Алгоритм метода половинного деления

Ввод a, b, ε, n

Начало цикла

$$x = \frac{a + b}{2}$$

$$F1 = f(a)$$

$$F2 = f(x)$$

Если $F1 * F2 > 0$, то $a = x$ иначе $b = x$

$$n = n + 1$$

Конец цикла, условие $|b - a| > \varepsilon$

Печать $x, f(x), n$

Ход решения:

1. Построить график функции.
2. Найти точку пересечения функции с осью X , т.е. найти корень уравнения.

3. Определить отрезок $[a, b]$ содержащий корень уравнения.

Теорема: Если функция определена, непрерывна, строго монотонна и меняет знак на границах отрезка $[a, b]$ то в этом отрезке есть корень уравнения, причем единственный.

4. Написать программу, реализующий описанный выше алгоритм, и ввести начальные данные согласно первым пунктам.

5. Вывести на экран результат программы.

Алгоритм метода простых итераций

Ввод x, ε, n

Начало цикла

$$x1 = \varphi(x)$$

$$c = |x1 - x|$$

$$x = x1$$

$$n = n + 1$$

Конец цикла, условие $c > \varepsilon$

Печать $x, f(x), n$

Ход решения:

1. Найти $\varphi(x)$ уравнения, т.е. выразить x .
2. Проверить условие сходимости для корректного ввода начального приближения (значение x): $|\varphi'(x)| < 1$.
3. Написать программу, реализующий описанный выше алгоритм, и ввести начальные данные.
4. Вывести на экран результат программы.

Алгоритм метода касательных (Ньютона)

Ввод x, ε, n, h

Начало цикла

$$pr = \frac{f(x)(x + h) - f(x+h)}{h}$$

$$x1 = x - \frac{f(x)}{pr}$$

$$c = |x1 - x|$$

$$x = x1$$

$$n = n + 1$$

Конец цикла, условие $c > \varepsilon$

Печать $x, f(x), n$

Ход решения:

1. Найти $f(x_0) * f''(x_0) > 0$ уравнения.
2. Проверить условие сходимости для корректного ввода начального приближения (значение x): $f(x_0) * f''(x_0) > 0$.
3. Найти $f'(x)$ уравнения, т.е. найти первую производную функции.
4. Написать программу, реализующий описанный выше алгоритм, и ввести начальные данные.
5. Вывести на экран результат программы.

Алгоритм метода хорд

Ввод x, p, ε, n

Начало цикла

$$x1 = x - \frac{f(x) * (x - p)}{f(x) - f(p)}$$

$$c = |x1 - x|$$

$$x = x1$$

$$n = n + 1$$

Конец цикла, условие $c > \varepsilon$

Печать $x, f(x), n$

Ход решения:

1. Найти $f(x) * f''(x) < 0$ уравнения.

2. Проверить условие сходимости для корректного ввода начального приближения (подвижной границы) $f(x) * f''(x) < 0$.

3. Написать программу, реализующий описанный выше алгоритм, и ввести начальные данные.

4. Вывести на экран результат программы.

Замечание. Значение $f(x)$ на выходе программы должно быть максимально близким к 0, результаты каждого метода должны быть приближенно равными между собой.

ВАРИАНТЫ ЗАДАНИЙ

№ варианта	Уравнение	Точность
1	2	3
1	$x^3 + 4x - 6 = 0$	$\varepsilon = 0.0001$
2	$x^3 + 3x + 1 = 0$	$\varepsilon = 0.0001$
3	$3x - \cos x - 1 = 0$	$\varepsilon = 0.0001$
4	$3x^4 + 4x^3 - 12x^2 - 5 = 0$	$\varepsilon = 0.0001$
5	$x^3 + 4x - 6 = 0$	$\varepsilon = 0.0001$
6	$3x - e^x = 0$	$\varepsilon = 0.0001$
7	$x^3 - 2x^2 - 4x + 7 = 0$	$\varepsilon = 0.0001$
8	$x^3 + 3x - 1 = 0$	$\varepsilon = 0.0001$

9	$2x^4 - 8x^3 + 8x^2 - 1 = 0$	$\varepsilon = 0.0001$
10	$3x^4 + 8x^3 + 6x^2 - 10 = 0$	$\varepsilon = 0.0001$

Входные данные, после запуска программы – отрезок, содержащий корень (или начальное приближение) и точность вычисления.

Выходные данные – найденный корень уравнения и проверка (подстановка найденного значения в исходное уравнение).

Реализация алгоритмов на языке C#

Ниже представлен программный код реализации *метода половинного деления*. В основной части программы задаются значения переменных, необходимых для реализации алгоритма, описывается сам алгоритм в виде цикла с предусловием WHILE, на каждом шаге которого вызывается метод *func* для вычисления функции при значениях a или x , и организовывается вывод значения x , количество итераций n и значение функции $F(x)$.

Static double func (double x) //метод для нахождения функции

```
{
Return Math. Pow (x, 3) + 4 * x - 6;
}
static void Main (string[] args);
{
Console. Write Line («Метод половинного деления»);
double a = 1, b = 2, n = 0, e = 0.0001, x = 0;
while (Math. Abs (b - a) > e);
{
x = (a + b) / 2;
if (func(a) * func (x) > 0) a = x;
else b = x;
n++;
}
x = (a + b) / 2;
Console. Write Line ("x= " + x + "\n= " + n + "\nF (x)= " + func (x)).
```

Для реализации, например, *метода хорд*, необходимо использовать цикл с постусловием, в котором также производится многократный вызов метода *func*.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Корень уравнения.
2. Основные этапы нахождения решения.
3. Метод половинного деления. Геометрическая интерпретация.

Причина выбора отрезка $[a, b]$.

4. Метод простых итераций. Геометрическая иллюстрация и условие сходимости.

5. Метод касательных. Геометрическое представление и условие сходимости.

6. Метод хорд. Геометрическая интерпретация и условие сходимости.