

Московский Авиационный Институт
(Национальный Исследовательский Университет)

Кафедра 806
«Компьютерные науки и прикладная математика»

Курсовая работа
Дисциплина: «Фундаментальная информатика»
I семестр

Группа:	М8О-101Б-22 вар. 26
Студент:	Шляхтуров Александр Викторович
Преподаватель:	Крылов Сергей Сергеевич
Оценка:	
Дата:	20.12.2022

Москва, 2022

1. Задание

Составить программу на языке Си с процедурами решения трансцендентных алгебраических уравнений различными численными методами (итераций, Ньютона и половинного деления — дихотомии). Уравнения оформить как функции параметры, разрешив относительно неизвестной величины в случае необходимости.

Решение

Краткие сведения из численных методов

Рассматривается уравнение вида $F(x)=0$. Предполагается, что функций $F(x)$ достаточно гладкая, монотонная на этом отрезке и существует единственный корень уравнения $x \in [a,b]$. На отрезке $[a, b]$ ищется приближенное решение x с точностью ε , т. е. такое, что $|x - x| < \varepsilon$.

При решении реальных задач, где поведение функции $F(x)$ неизвестно, сначала производят исследование функции (аналитическое, численное или графическое), например, с помощью программ gnuplot, MathLab, MathCAD, Maple. Также выполняют т. н. отделение корней, т. е. разбивают область определения функции на отрезки монотонности, на каждом из которых имеется ровно один корень и выполняются другие условия применимости численных методов (гладкость). Различные численные методы предъявляют разные требования к функции $F(x)$, обладают различной скоростью сходимости и поведением.

В данном задании предлагается изучить и запрограммировать три простейших численных метода решения алгебраических уравнений и провести вычислительные эксперименты по определению корней уравнений на указанных в задании отрезках монотонности и, в качестве дополнительного упражнения, вне их.

Метод Ньютона

Метод Ньютона является частным случаем метода итераций.

Условие сходимости метода: $|F(x) \cdot F''(x)| < (F'(x))^2$ на отрезке $[a, b]$.

Итерационный процесс: $x^{k+1} = x^k - F(x^k)/F'(x^k)$.

Метод половинного деления

Очевидно, что если на отрезке $[a, b]$ существует корень уравнения, то значения функции на концах отрезка имеют разные знаки: $F(a) \cdot F(b) < 0$. Метод заключается в делении отрезка пополам и его сужении в два раза на каждом шаге итерационного процесса в зависимости от знака функции в середине отрезка.

Метод Ньютона

Итерационный численный метод нахождения корня заданной функции, который является частным случаем метода простых итераций. А именно за λ_0 берётся значение производной в каждой новой точке. Тогда итерационный процесс имеет вид $x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$. Условие окончания итераций и начальное значение абсолютно такие же, как и в методе итерации. Условие сходимости: $|f(x) \cdot f''(x)| < (f'(x))^2$

Вариант

Вариант 12	
Уравнение	$e^x - e^{-x} - 2 = 0$
Интервал	$[0; 1]$
Требуемый ответ	0.881

Код программы

```
2. #include <stdio.h>
3. #include <math.h>
4.
5. // Функция, данная в задании
6. long double func ( long double x ) {
7.     return expl(x) - expl(-x) - 2;
8. }
9.
10. // Производная функции, данной в задании
11. long double dif_func (long double x) {
12.     return expl(x) + expl(-x);
13. }
14.
15. // Вторая производная функции, данной в задании
16. long double dif_dif_func (long double x) {
17.     return expl(x) - expl(-x);
18. }
19.
20. // Функция модуль
21. long double my_abs( long double x) {
22.     if ( x < 0 ) {
23.         return -x;
24.     }
25.     return x;
26. }
27.
28. int main() {
29.     //eps - машинное эpsilon (очень маленькое число); a - левая граница
    отрезка; b - правая граница отрезка; x - текущие промежуточное значение
    корня; x1 - предыдущие промежуточное значение корня
30.     long double eps = 1.01, a = 0.01, b = 1.01, x, x1 = 0;
31.
32.     // Вычисление машинного эpsilon
33.     while (1.01 + eps / 2.01 > 1.01) {
34.         eps /= 2.01;
35.     }
36.
37.     printf("Машинное эpsilon для типа double = %.16Le\n", eps);
38.
39.     // Проверка на сходимость
40.     // Для этого проходимся циклом от левой границе к правой, с шагом
    1/1000000, проверяя, выполнение условие сходимости  $|F(x) * F'(x)| < (F'(x))^2$ ,
    если условие не выполняется, мы завершаем программу
41.     for (long long i = 0; i <= 1000000; i++ ) {
42.         x = i / 1000000.01;
43.         if (my_abs(func(x) * dif_dif_func(x)) - dif_func(x) * dif_func(x)
            >= eps) {
```

```

44.         printf("Невозможно вычислить значение методом Ньютона, метод
    не сходится.\n");
45.         return 0;
46.     }
47. }
48.
49. // Изначально, x равен середине отрезка
50. x = (a + b) / 2;
51.
52. // Само вычисление корня
53. while (my_abs(x - x1) >= eps) {
54.     x1 = x;
55.     x = x1 - func(x) / dif_func(x);
56. }
57.
58. // Вывод ответа
59. printf("Приближенное значение корня, полученного при помощи метода
    Ньютона равно: %Le\n", x);
60.}

```

60.1. Результат работы программы

```

alexander@DESKTOP-KNBCFCI:~/kp4$ ./a.out
Машинное эпсилон для типа double = 1.0842021724855044e-19
Приближенное значение корня, полученного при помощи метода Ньютона равно: 8.813736e-01
alexander@DESKTOP-KNBCFCI:~/kp4$ █

```

61. Заключение

Изучая различные методы решения тех или иных задач, можно прийти к выводу, что иногда не все способы решения подходят для точного выполнения определённой задачи. Именно поэтому очень важно и нужно знать различные приемы, правила, с помощью которых можно реализовать решение одной задачи, а также нужно уметь выбрать среди них самый точный и эффективный алгоритм. Именно этим навыкам и обучает данная курсовая работа.

62.Список использованной литературы

1. Брайан Керниган, Деннис Ритчи. Язык программирования Си. Второе издани: Серия книг по программированию от Prentice Hall. 2009. – 292с.:ил.
2. Алгоритм метода половинного деления [Электронный ресурс]// <https://math.semestr.ru/> URL: <https://math.semestr.ru/optim/dichotomy-algorithm.php>
3. Математический энциклопедический словарь. — М.: «Сов. энциклопедия », 1988. — С. 847