

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский Авиационный Институт»  
(Национальный Исследовательский Университет)

Институт: №8 «Компьютерные науки и прикладная математика»  
Кафедра: 806 «Вычислительная математика и программирование»

Курсовая работа  
по курсу «Фундаментальная  
информатика» I семестр  
Задание 4  
«Процедуры и функции в качестве параметров»

Группа	М8О-109Б-22
Студент	Серякова А.А.
Преподаватель	Сысоев М.А.
Оценка	
Дата	29.12.2022

## Постановка задачи

Составить программу на Си с процедурами решения трансцендентных алгебраических уравнений различными численными методами (итераций, Ньютона и половинного деления — дихотомии). Нелинейные уравнения оформить как параметры-функции, разрешив относительно неизвестной величины в случае необходимости. Применить каждую процедуру к решению двух уравнений, заданных двумя строками таблицы, начиная с варианта с заданным номером. Если метод неприменим, дать математическое обоснование и графическую иллюстрацию, например, с использованием `gnuplot`.

Вариант 15, 16:

Функция:

15	$0,4 + \operatorname{arctg} \sqrt{x} - x = 0$	[1, 2]	итераций	1.2388
16	$3 \sin \sqrt{x} + 0.35x - 3.8 = 0$	[2, 3]	итераций	2.2985

### Что требуется в задании?

В задании требуется реализовать 3 метода нахождения корней уравнения вида  $F(x) = 0$  и применить их для двух соответствующих функций. Причем корень уравнения находится не точно, а с погрешностью, равной  $\varepsilon$ , и лежит на отрезке  $[a, b]$ . Реализовать нужно такие методы как: метод Дихотомии, метод итераций, метод Ньютона (частный случай метода итераций).

#### 2. Метод итераций.

Идея метода заключается в замене исходного уравнения  $F(x) = 0$  уравнением вида  $x = f(x)$ .

Достаточное условие сходимости метода:  $|f'(x)| < 1, x \in [a, b]$ . Это условие необходимо проверить перед началом решения задачи, так как функция  $f(x)$  может быть выбрана неоднозначно, причем в случае неверного выбора указанной функции метод расходится.

Начальное приближение корня:  $x^{(0)} = (a + b) / 2$  (середина исходного отрезка).

Итерационный процесс:  $x^{(k+1)} = f(x^{(k)})$ .

Условие окончания:  $|x^{(k)} - x^{(k-1)}| < \varepsilon$ .

Приближенное значение корня:  $x^* \approx x^{(k_{\text{конечное}})}$ .

### Вариант 15

15	$0,4 + \operatorname{arctg} \sqrt{x} - x = 0$	[1, 2]	итераций	1.2388
----	---	--------	----------	--------

Преобразуем уравнение  $F(x) = 0$  к виду  $f(x) = x$  и находим производную (это функции  $D$  и  $f$ ).

```
#include <stdio.h>
```

```

#include <math.h>
#include <stdbool.h>
double (*someFunc) (double);
typedef double (*Function) (double);
double IterationMethod(Function func,double a,double b,double eps) {
    double x = (a + b) / 2.0;
    while(fabs(func(x) - x) > eps)
        x = func(x);
    return x;
}
bool checkIterationMethod(Function derivative,Function func,double a,double
b,double eps) {
    double x = (a + b) / 2.0;
    while(fabs(func(x) - x) > eps) {
        if(fabs(derivative(x)) >= 1.0)
            return false;
        x = func(x);
    }
    return true;
}
double f(double x) {
    return 0.4+atan(sqrtl(x));
}
double D(double x) {
    return 1.0/((x+1.0)*2.0*sqrtl(x));
}
int main() {
    double eps = 0.001;
    printf("Iteration Method result: ");
    if(checkIterationMethod(&D, &f, 2.0, 3.0, eps)) {
        printf("%.4f", IterationMethod( &f, 2.0, 3.0, eps));
    }
    else {
        printf("cannot be applied\n");
    }
}

```

/tmp/R4Ghx6qmmq.o

Iteration Method result: 1.2401

16	$3 \sin \sqrt{x} + 0.35x - 3.8 = 0$	[2, 3]	итераций	2.2985
----	-------------------------------------	--------	----------	--------

```
#include <stdio.h>
#include <math.h>
#include <stdbool.h>
double (*someFunc) (double);
typedef double (*Function) (double);
double IterationMethod(Function func,double a,double b,double eps) {
    double x = (a + b) / 2.0;
    while(fabs(func(x) - x) > eps)
        x = func(x);
    return x;
}
bool checkIterationMethod(Function derivative,Function func,double a,double
b,double eps) {
    double x = (a + b) / 2.0;
    while(fabs(func(x) - x) > eps) {
        if(fabs(derivative(x)) >= 1.0)
            return false;
        x = func(x);
    }
    return true;
}
double f(double x) {
    return 3.8/0.35-3.0*sin(sqrt(x))/0.35;
}
double D(double x) {
    return -30.0*cos(sqrt(x))/(7*sqrt(x));
}
int main() {
    double eps = 0.001;
    printf("Iteration Method result: ");
    if(checkIterationMethod(&D, &f, 2.0, 3.0, eps)) {
        printf("%.4f", IterationMethod( &f, 2.0, 3.0, eps));
    }
    else {
        printf("cannot be applied\n");
    }
}
```

/tmp/R4Ghx6qmmq.o

Iteration Method result: 2.2982

Вывод

Я написала программы используя метод итераций. Все работает хорошо и все

понятно. Так же я ознакомилась и с другими методами.