Университет ИТМО

Лабораторная работа №2 «Системы нелинейных уравнений»

по дисциплине: Вычислительная математика

Вариант: 2бв

Выполнил: Неграш Андрей, P3230

Преподаватель: Перл Ольга Вячеславовна

Санкт-Петербург, 2022

Метод хорд

1. Описание метода решения

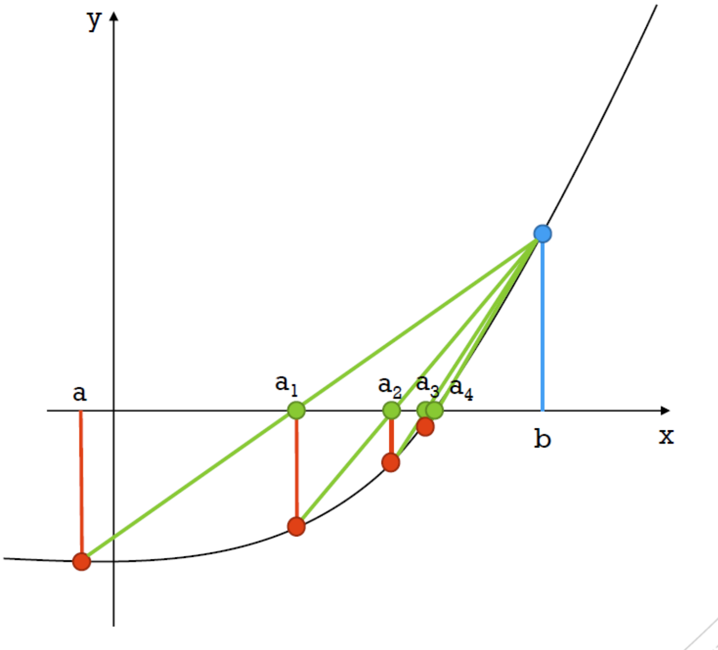
Нелинейная функция *f(x)* на отделённом интервале [a, b] заменяется линейной, в качестве которой берётся хорда (прямая, стягивающая концы нелинейной функции)

Затем вычисляются значения функции на концах отрезка, и строится прямая, соединяющая точки (a, f(a)) и (b, f(b)). При решении нелинейного уравнения методом хорд задаются интервал [a, b], на котором существует только одно решение, и точность ε.

Затем через две точки с координатами (a, f(a)) и (b, f(b)) проводим хорду и определяем точку пересечения этой линии с осью абсцисс, точку c. Если при этом *f(a) × f(c) < 0*, то b = c, иначе a = c.

Поиск решения прекращается при достижении заданной точности |f(c)| < ε.

2. Расчётные формулы



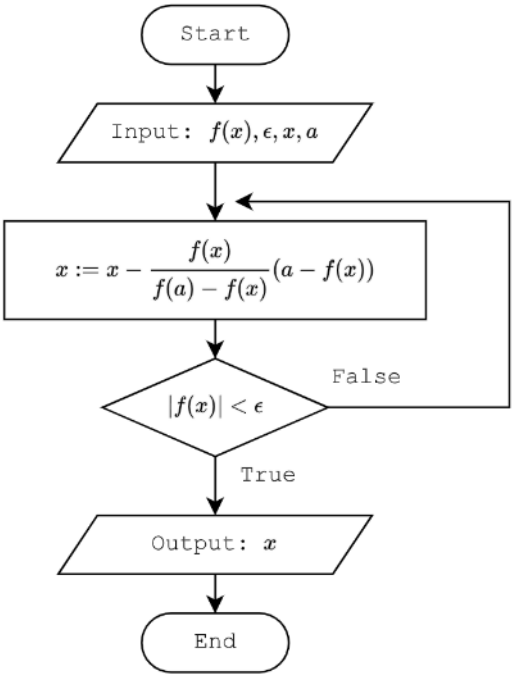
$$x_2 = x_1 - f(x_1)\frac{x_1 - x_0}{f(x_1) - f(x_0)},$$

$$x_3 = x_2 - f(x_2)\frac{x_2 - x_1}{f(x_2) - f(x_1)},$$

$$\vdots$$

$$x_n = x_{n-1} - f(x_{n-1})\frac{x_{n-1} - x_{n-2}}{f(x_{n-1}) - f(x_{n-2})}.$$

3. Блок-схема численного метода



4. Листинг численного метода

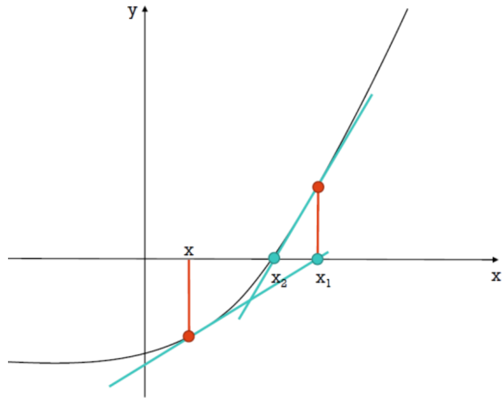
```
public AnswerX calculate(Func function, double precision, double begin, double end,
int iterationCount) {
    double newBegin = begin - (function.calcFunc(begin)/(function.calcFunc(end) -
function.calcFunc(begin)) *
            (end - begin));
    double diff = Math.abs(function.calcFunc(newBegin));

    if ( diff < precision ||iterationCount > 50 ) return new AnswerX(newBegin,
iterationCount);
    return calculate(function, precision, newBegin, end, iterationCount + 1);
}
```

Метод касательных

1. Описание метода решения

Классический метод касательных заключается в том, что если x_n — некоторое приближение к корню x уравнения $f(x) = 0$, то следующее приближение определяется как корень касательной к функции $f(x)$, проведённой в точке x_n. Уравнение касательной к функции $f(x)$ в точке x_n имеет вид:

$$f'(x_j) = \frac{y - f(x_n)}{x - x_n}$$



Тогда при $y = 0$ и $x = x_n$ алгоритм последовательных вычислений сводится к

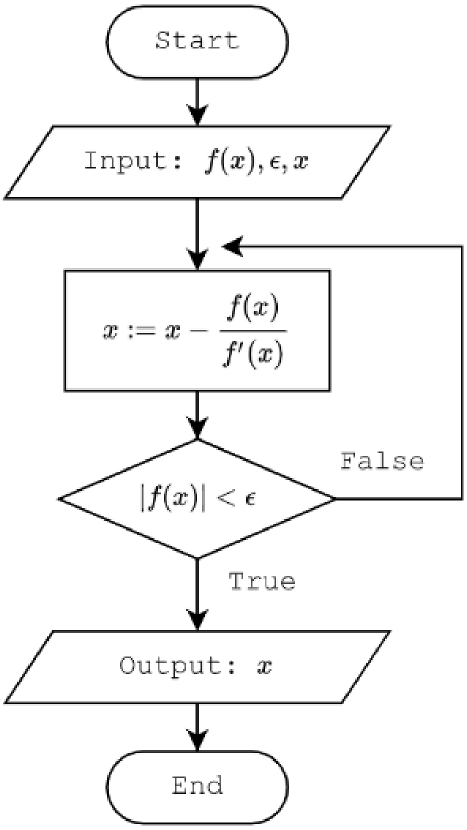
$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Вычисления производятся пока $|x_n - x_{n-1}| < e$, где е – допустимая абсолютная погрешность, заданная для конкретного решения.

Метод касательных применим только если соблюдаются несколько условий:

1) функция *y = f(x)* определена и непрерывна при $x \in [-\infty; +\infty]$
2) *f(a) × f(b) < 0* (функция принимает значения разных знаков на концах отрезка [*a, b*]);
3) производные *f'(x)* и *f''(x)* сохраняют знак на отрезке [*a, b*] (т.е. функция *f(x)* либо возрастает, либо убывает на отрезке [*a, b*], сохраняя при этом направление выпуклости);
4) $f'(x) \neq 0$ при $x \in [a; b]$

2. Блок-схема численного метода



3. Листинг численного метода

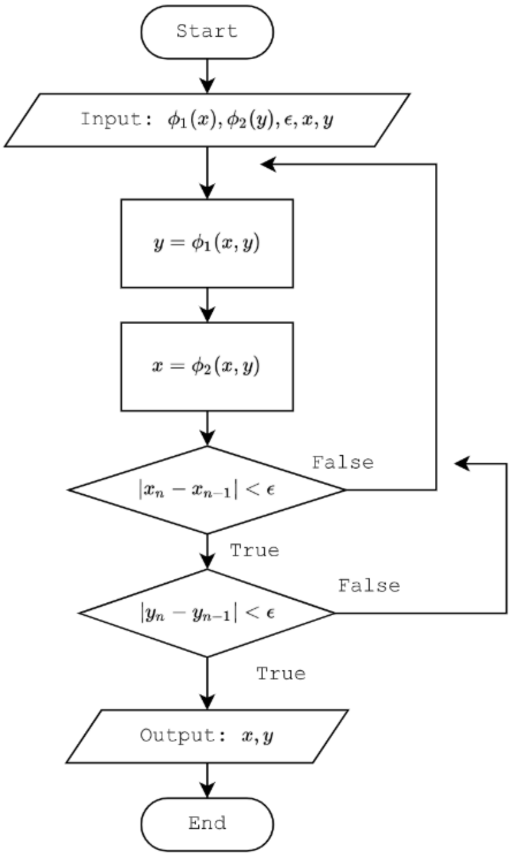
```
public AnswerX calculate(Func function, double precision, double approx, int
iterationCount) {
    double newApprox = approx - ( function.calcFunc(approx)/function.calcDer(approx)
);
    double diff = Math.abs(function.calcFunc(newApprox));

    if ( diff < precision || iterationCount > 50 ) return new AnswerX(newApprox,
iterationCount);
    return calculate(function, precision, newApprox, iterationCount + 1);
}
```

Метод простых итераций

1. Описание метода решения

Для того, чтобы решить СНАУ методом простых итераций, для начала нужно преобразовать каждое уравнение в системе к такому виду: $x_i = \varphi(x_1, x_2, \dots, x_n)$ – то есть выразить из каждого уравнение неизвестную переменную. После этого нужно выбрать вектор начального приближения X^0. Алгоритм для расчёта каждого последующего приближения: $X^{k+1} = F(X^0)$.

2. Блок-схема численного метода



3. Листинг численного метода

```
public AnswerXY calculate(EqSystem system, double precision, double x, double
y, int iterationCount) {
    double newX = system.x1(y);
    double newY = system.y2(x);

    double diffX = Math.abs(newX - x);
    double diffY = Math.abs(newY - y);
```

```
    if ((diffX < precision && diffY < precision) || iterationCount > 50)
        return new AnswerXY(newX, newY, iterationCount);

    return calculate(system, precision, newX, newY, iterationCount + 1);

}
```

Весь код доступен на GitHub: https://github.com/ANegrash/ITMO-all/tree/master/4%20Computational%20math/lab2

Работа программы

Уравнение 1

Лабораторная №2: "Системы нелинейных уравнений"
Вариант: 2бв
Автор: Неграш А.В., P3230

Список доступных команд:
sing - решить нелинейное уравнение
syst - решить систему нелинейных уравнений
q   - выйти из программы
h   - показать список доступных команд

>*sing*

Выберите уравнение:
1: x + cos(x) - 0.67x^3 - 1 = 0
2: x^3 + 2x^2 - 5x - 5 = 0
3: -x^2 + 5 = 0

>*1*

Отлично, вы выбрали уравнение №1
Ещё немного входных данных.
Введите начальное значение интервала для метода хорд:

>*0.5*

Введите конечное значение интервала для метода хорд:

>*1*

Введите точность для метода касательных:

>*0.001*

Введите начальное приближение для метода касательных:

>*1*

Проводим вычисления по методу хорд...
Ответ: x = 0,922642 за 4 итераций

Проводим вычисления по методу касательных...
Ответ: x = 0,922801 за 2 итераций

Разница между ответами, полученными методом хорд и методом касательных:
0,000160

Уравнение 2

Лабораторная №2: "Системы нелинейных уравнений"
Вариант: 2бв
Автор: Неграш А.В., Р3230

Список доступных команд:
sing - решить нелинейное уравнение
syst - решить систему нелинейных уравнений
q   - выйти из программы
h   - показать список доступных команд

>*sing*

Выберите уравнение:
1: x + cos(x) - 0.67x^3 - 1 = 0
2: x^3 + 2x^2 - 5x - 5 = 0
3: -x^2 + 5 = 0

>*2*

Отлично, вы выбрали уравнение №2
Ещё немного входных данных.
Введите начальное значение интервала для метода хорд:

>*1.5*

Введите конечное значение интервала для метода хорд:

>*2*

Введите точность для метода касательных:

>*0.001*

Введите начальное приближение для метода касательных:

>*2*

Проводим вычисления по методу хорд...
Ответ: x = 1,930773 за 3 итераций

Проводим вычисления по методу касательных...
Ответ: x = 1,930805 за 2 итераций

Разница между ответами, полученными методом хорд и методом касательных:
0,000032

Уравнение 3

Лабораторная №2: "Системы нелинейных уравнений"
Вариант: 2бв
Автор: Неграш А.В., P3230

Список доступных команд:
sing - решить нелинейное уравнение
syst - решить систему нелинейных уравнений
q   - выйти из программы
h   - показать список доступных команд

>*sing*

Выберите уравнение:
1: x + cos(x) - 0.67x^3 - 1 = 0
2: x^3 + 2x^2 - 5x - 5 = 0
3: -x^2 + 5 = 0

>*3*

Отлично, вы выбрали уравнение №3
Ещё немного входных данных.
Введите начальное значение интервала для метода хорд:

>*1*

Введите конечное значение интервала для метода хорд:

>*3*

Введите точность для метода касательных:

>*0.001*

Введите начальное приближение для метода касательных:

>*1*

Проводим вычисления по методу хорд...
Ответ: x = 2,235955 за 5 итераций

Проводим вычисления по методу касательных...
Ответ: x = 2,236069 за 4 итераций

Разница между ответами, полученными методом хорд и методом касательных:
0,000114

Система уравнений 1

Лабораторная №2: "Системы нелинейных уравнений"
Вариант: 2бв
Автор: Неграш А.В., P3230

Список доступных команд:
sing - решить нелинейное уравнение
syst - решить систему нелинейных уравнений
q - выйти из программы
h - показать список доступных команд

>*syst*

Выберите систему уравнений:
1: y = x^3;
y = x^2 - 6.

2: y = 0.1x^3;
y = x^2 - 0.5.

>*1*

Отлично, вы выбрали систему уравнений №1
Введите точность:

>*0.01*

Введите начальное приближение для x:

>*1*

Введите начальное приближение для y:

>*1*

Отлично! Проводим вычисления по методу простой итерации...
Решение ситемы: x = -1,534810; y = -3,615462 за 12 итераций

Система уравнений 2

Лабораторная №2: "Системы нелинейных уравнений"
Вариант: 2бв
Автор: Неграш А.В., P3230

Список доступных команд:
sing - решить нелинейное уравнение
syst - решить систему нелинейных уравнений
q   - выйти из программы
h   - показать список доступных команд

>*syst*

Выберите систему уравнений:
1: y = x^3;
   y = x^2 - 6.

2: y = 0.1x^3;
   y = x^2 - 0.5.

>*2*

Отлично, вы выбрали систему уравнений №2
Введите точность:

>*0.01*

Введите начальное приближение для x:

>*1*

Введите начальное приближение для y:

>*1*

Отлично! Проводим вычисления по методу простой итерации...
Решение ситемы: x = 9,947876; y = 98,435811 за 47 итераций

Вывод

Метод касательных хорошо применим для повышения точности корня уравнения, полученного другим методом, однако при этом он имеет значимые минусы: из-за использования касательных при f’(x)=0 метод не сможет найти корень, и существуют условия, при которых он не сходится. Также эффективность метода сильно снижается при недостаточном изначальном приближении корня.

Метод хорд применим для нахождения корня уравнения на определённом интервале. Также, как и для метода касательных, существуют условия, при которых метод не сходится и точно также возникают проблемы при построении хорд для f’(x)=0. Помимо этого методу хорд обычно требуется большее количество итераций, чем метод касательных. Однако есть и плюс – метод хорд не требует использования производной от функции при вычислении.

Плюсы **метода простых итераций** состоят в том, что он очень прост в понимании и реализации. Однако для работы с ним на практике придётся столкнуться с большим количеством итераций для достижения достаточного приближения корня.