**Министерство науки и высшего образования Российской Федерации**

федеральное государственное автономное образовательное учреждение

высшего образования

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**

**ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Инженерная школа природных ресурсов

Направление подготовки Химическая технология

Отделение химической инженерии

**ИТЕРАЦИОННЫЕ МЕТОДЫ РЕШЕНИЯ НЕЛИНЕЙНЫХ УРАВНЕНИЙ**

**Лабораторная работа по дисциплине «Углубленный курс информатики»**

Выполнил студент гр. 2Д93 К.С. Гейден

(Подпись)

\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ 2020 г.

Отчет принят:

Преподаватель

доцент ОХИ ИШПР, к.т.н. В.А. Чузлов

(Подпись)

\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ 2020 г.

Томск 2020 г.

**Цель работы:** научиться использовать итерационные методы (половинного деления, простых итераций и метод Ньютона) для решения нелинейных уравнений.

**Теоретическая часть**

Метод деления отрезка пополам

Пусть дана функция f(x) = 0, непрерывная на интервале [a, b].

•f(a) \* f(b) < 0 – условие наличия корня на данном интервале.

•Поделим отрезок [a, b] пополам точкой x1 c координатой   
x1 = (a + b) / 2 и вычислим значение функции f(x1).

Возможны два случая:

1. f(a) \* f(x1) > 0 → корень находится на отрезке [x1, b], исключаем отрезок [a, x1];

2. f(a) \* f(x1) < 0 → корень находится на отрезке [a, x1], исключаем отрезок [x1, b].

•Продолжаем деление пополам до тех пор, пока длина оставшегося интервала   
[a, b] не станет меньше некоторой заданной малой величины e , т.е.

(b - a) < e , и тогда любое значение аргумента из отрезка [a, b] можно считать корнем с погрешностью e. Обычно принимают в качестве корня середину отрезка.

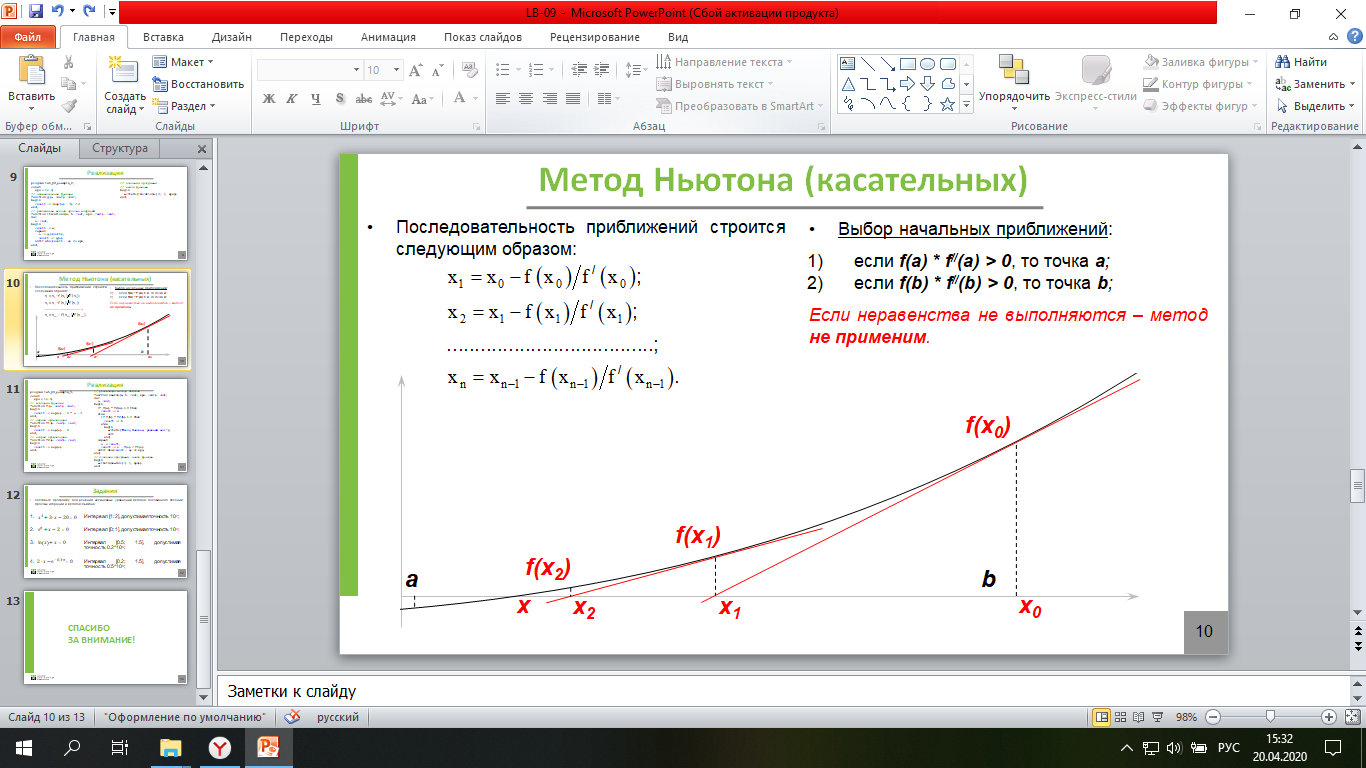
Метод простых итераций

•Заменить исходное уравнение f(x) = 0, на эквивалентное g(x) = x и строить итерации по правилу xn+1 = g(xn).

•Условие сходимости: |g’ (x)|<1

Метод Ньютона (касательных)

•Последовательность приближений строится следующим образом:



Выбор начальных приближений

1)если f(a) \* f//(a) > 0, то точка a;

2)если f(b) \* f//(b) > 0, то точка b;

Если неравенства не выполняются – метод не применим.

**Практическая часть**

**Задание 1**

**Исходные данные**: x4+3∙x-20=0

Интервал [1;2], допустимая точность 10-2

**Задание:** решить уравнение методом

1. половинного деления
2. простых итераций
3. методом Ньютона

**Программная реализация**

1)Метод половинного деления:

Program lab91;

const

eps = 1e-2;

function f(x: real): real;

begin

result := exp(4\*ln(x))+3\*x-20;

end;

function dihotomy(a, b: real; eps: real): real;

var

x: real;

begin

repeat

x := (a + b) / 2;

if f(a) \* f(x) > 0 then

a := x

else

b := x

until (abs(a - b) <= eps) or (f(x) = 0);

result := x

end;

begin

writeln('x = ',dihotomy(1, 2, eps))

end.

**Ответ**

x = 1.9453125

2)Метод итераций:

program LB9\_1;

const

eps = 1e-2;

function g(x: real): real;

begin

result := x-(exp(4\*ln(x))+3\*x-20)/35;

end;

function iterations(a, b: real; eps: real): real;

var

x: real;

begin

result := a;

repeat

x := g(result);

result := g(x)

until abs(result - x) <= eps;

end;

begin

writeln('x = ',iterations(1, 2, eps))

End.

**Ответ**

x = 1.94045649555801

3) Метод Ньютона:

program L9\_1;

const

eps = 1e-2;

function f(x: real): real;

begin

result := exp(4\*ln(x))+3\*x-20

end;

function f1(x: real): real;

begin

result := 4\*exp(3\*ln(x))+3

end;

function f2(x: real): real;

begin

result := 12\*exp(2\*ln(x))

end;

function newton(a, b: real; eps: real): real;

var

x: real;

begin

if f(a) \* f2(a) > 0 then

result := a

else

if f(b) \* f2(b) > 0 then

result := b

else

begin

writeln('Метод Ньютона, решений нет!');

exit

end;

repeat

x := result;

result := x - f(x) / f1(x)

until abs(result - x) <= eps;

end;

begin

writeln('x = ',newton(1, 2, eps))

End.

**Ответ**

x = 1.94047935224908

**Задание 2**

**Исходные данные**: ex+x-2=0

Интервал [0;1], допустимая точность 10-3

**Задание:** решить уравнение методом

1. половинного деления
2. простых итераций
3. методом Ньютона

**Программная реализация**

1)Метод половинного деления:

program LB9\_2;

const

eps = 1e-3;

function f(x: real): real;

begin

result := exp(x)+x-2;

end;

function dihotomy(a, b: real; eps: real): real;

var

x: real;

begin

repeat

x := (a + b) / 2;

if f(a) \* f(x) > 0 then

a := x

else

b := x

until (abs(a - b) <= eps) or (f(x) = 0);

result := x

end;

begin

writeln('x=',dihotomy(0, 1, eps))

end.

**Ответ**

x=0.4423828125

2)Метод итераций:

program lab92;

const

eps = 1e-3;

function g(x: real): real;

begin

result := x-(exp(x)+x-2)/4;

end;

function iterations(a, b: real; eps: real): real;

var

x: real;

begin

result := a;

repeat

x := g(result);

result := g(x)

until abs(result - x) <= eps;

end;

begin

writeln('x = ',iterations(0, 1, eps))

End.

**Ответ**

x = 0.442675206680943

3)Метод Ньютона:

program L9\_2;

const

eps = 1e-3;

function f(x: real): real;

begin

result := exp(x)+x-2

end;

function f1(x: real): real;

begin

result := exp(x)+1

end;

function f2(x: real): real;

begin

result := exp(x)

end;

function newton(a, b: real; eps: real): real;

var

x: real;

begin

if f(a) \* f2(a) > 0 then

result := a

else

if f(b) \* f2(b) > 0 then

result := b

else

begin

writeln('Метод Ньютона, решений нет!');

exit

end;

repeat

x := result;

result := x - f(x) / f1(x)

until abs(result - x) <= eps;

end;

begin

writeln('x = ',newton(0, 1, eps))

End.

**Ответ**

x = 0.442854401004033

**Задание 3**

**Исходные данные**: ln⁡(x)+х=0

Интервал [0,5;1,5], допустимая точность 2\*10-5

**Задание:** решить уравнение методом

1. половинного деления
2. простых итераций
3. методом Ньютона

**Программная реализация**

1)Метод половинного деления:

program lab93;

const

eps = 2e-5;

function f(x: real): real;

begin

result := ln(x)+x;

end;

function dihotomy(a, b: real; eps: real): real;

var

x: real;

begin

repeat

x := (a + b) / 2;

if f(a) \* f(x) > 0 then

a := x

else

b := x

until (abs(a - b) <= eps) or (f(x) = 0);

result := x

end;

begin

writeln('x = ',dihotomy(0.5, 1.5, eps))

End.

**Ответ**

x = 0.567153930664063

2)Метод интераций:

program LB9\_3;

const

eps = 2e-5;

function g(x: real): real;

begin

result := x-(ln(x)+x)/3;

end;

function iterations(a, b: real; eps: real): real;

var

x: real;

begin

result := a;

repeat

x := g(result);

result := g(x)

until abs(result - x) <= eps;

end;

begin

writeln('x = ',iterations(0.5, 1.5, eps))

End.

**Ответ**

x = 0.567141959719474

3) Метод Ньютона:

program L9\_3;

const

eps = 2e-5;

function f(x: real): real;

begin

result := ln(x)+x

end;

function f1(x: real): real;

begin

result := 1/x+1

end;

function f2(x: real): real;

begin

result := -1/sqr(x)

end;

function newton(a, b: real; eps: real): real;

var

x: real;

begin

if f(a) \* f2(a) > 0 then

result := a

else

if f(b) \* f2(b) > 0 then

result := b

else

begin

writeln('Метод Ньютона, решений нет!');

exit

end;

repeat

x := result;

result := x - f(x) / f1(x)

until abs(result - x) <= eps;

end;

begin

writeln('x = ',newton(0.5, 1.5, eps))

End.

**Ответ**

x = 0.567143290399369

**Задание 4**

**Исходные данные**: 2\*x-e-0,1∙x=0

Интервал [0,2;1,5], допустимая точность 5\*10-5

**Задание:** решить уравнение методом

1. половинного деления
2. простых итераций
3. методом Ньютона

**Программная реализация**

1)Метод половинного деления:

program lab94;

const

eps = 5e-5;

function f(x: real): real;

begin

result := 2\*x-exp(-0.1\*x);

end;

function dihotomy(a, b: real; eps: real): real;

var

x: real;

begin

repeat

x := (a + b) / 2;

if f(a) \* f(x) > 0 then

a := x

else

b := x

until (abs(a - b) <= eps) or (f(x) = 0);

result := x

end;

begin

writeln('x = ',dihotomy(0.2, 1.5, eps))

End.

**Ответ**

x = 0.476718139648438

2)Метод итераций:

program LB9\_1;

const

eps = 5e-5;

function g(x: real): real;

begin

result := x-(2\*x-exp(-0.1\*x))/2;

end;

function iterations(a, b: real; eps: real): real;

var

x: real;

begin

result := a;

repeat

x := g(result);

result := g(x)

until abs(result - x) <= eps;

end;

begin

writeln('x = ',iterations(0.2, 1.5, eps))

End.

**Ответ**

x = 0.476721637710739

3)Метод Ньютона:

program lab9ex3;

const

eps = 5e-5;

function f(x: real): real;

begin

result := 2\*x-exp(-0.1\*x)

end;

function f1(x: real): real;

begin

result := 2+exp(-0.1\*x)/10

end;

function f2(x: real): real;

begin

result := -exp(-0.1\*x)/100

end;

function newton(a, b: real; eps: real): real;

var

x: real;

begin

if f(a) \* f2(a) > 0 then

result := a

else

if f(b) \* f2(b) > 0 then

result := b

else

begin

writeln('Метод Ньютона, решений нет!');

exit

end;

repeat

x := result;

result := x - f(x) / f1(x)

until abs(result - x) <= eps;

end;

begin

writeln('x = ',newton(0.2, 1.5, eps))

End.

**Ответ**

x = 0.476723086001294

**Выводы**

В ходе лабораторной работы были изучены итерационные методы решения (половинного деления, простых итераций и метод Ньютона) для решения нелинейных уравнений.