**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»**

Выполнила студентка гр. 2Д93 А.И. Чустеева

(Подпись)

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

Отчет принят:

Преподаватель

доцент ОХИ ИШПР, к.т.н. В.А. Чузлов

(Подпись)

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

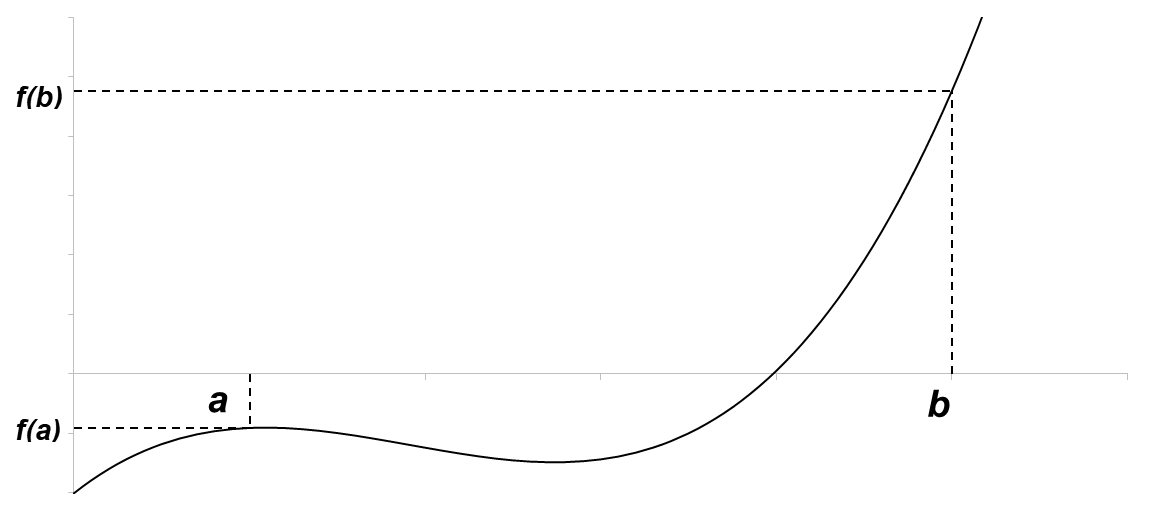
Томск 2020 г.

**Цель работы:** научиться решать нелинейные уравнения методом половинного деления, методом простых итераций и методом Ньютона.

**Теоритическая часть.**

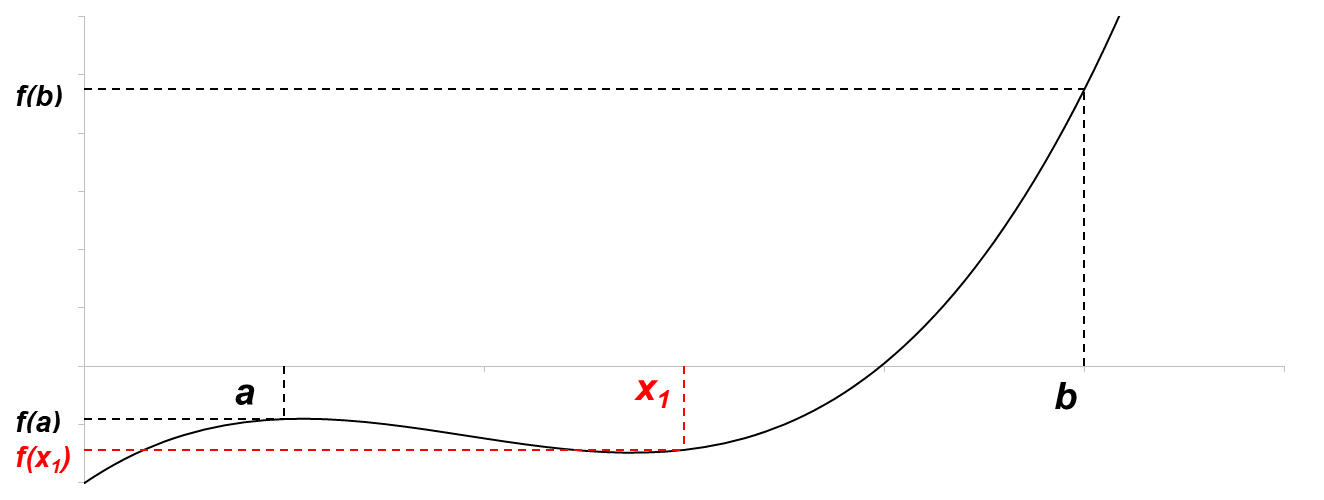
**Метод деления отрезка пополам**

* Пусть дана функция ***f(x) = 0***, непрерывная на интервале ***[a, b]***.
* ***f(a) \* f(b) < 0*** – условие наличия корня на данном интервале.



**Метод деления отрезка пополам**

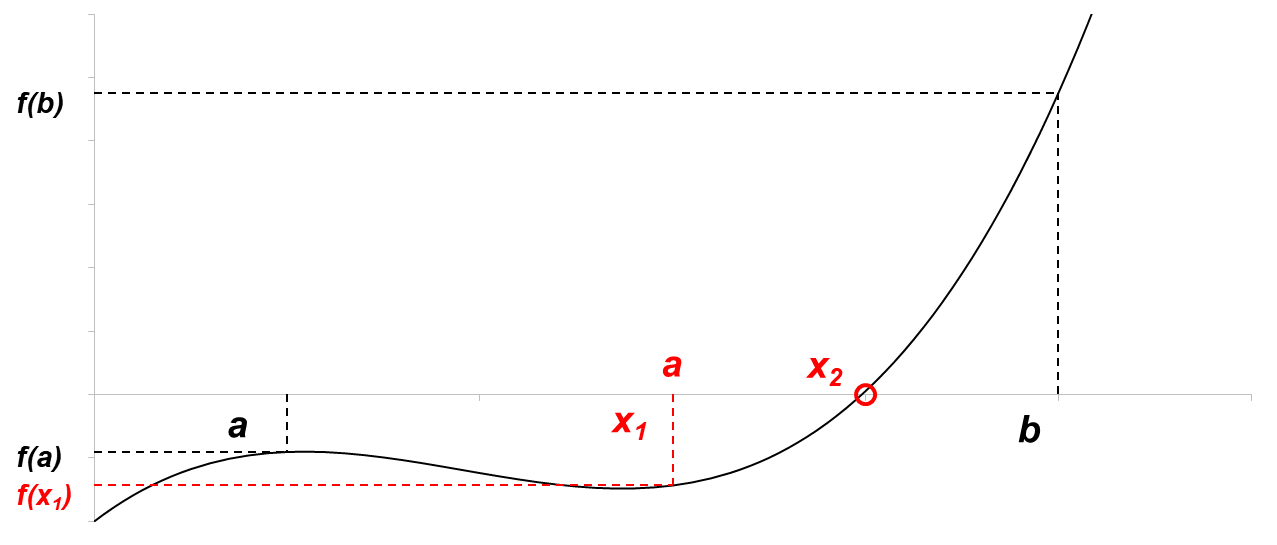
* Поделим отрезок ***[a, b]*** пополам точкой ***x1*** c координатой   
  ***x1 = (a + b) / 2*** и вычислим значение функции ***f(x1)***.



*Возможны два случая:*

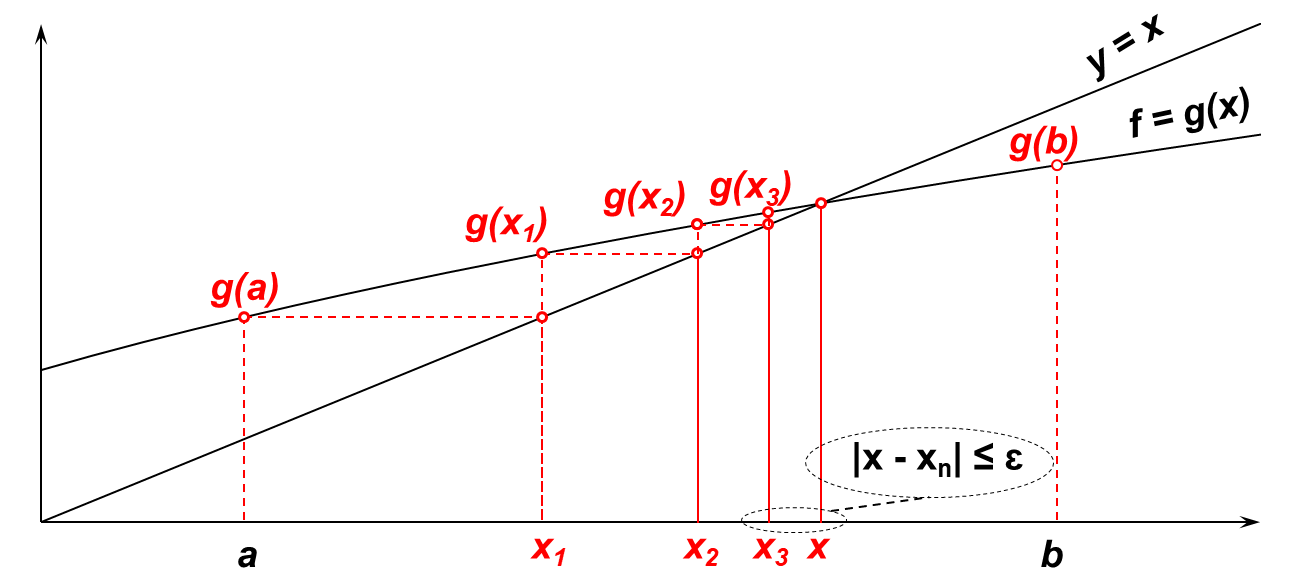
1. ***f(a) \* f(x1) > 0***→ корень находится на отрезке ***[x1, b]***,исключаем отрезок ***[a, x1];***
2. ***f(a) \* f(x1) < 0*** → корень находится на отрезке ***[a, x1]***,исключаем отрезок ***[x1, b].***

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**

* Заменим исходное уравнение ***f(x) = 0***, на эквивалентное ***g(x) = x*** и будем строить итерации по правилу ***xn+1 = g(xn)***.
* Условие сходимости:



**Метод Ньютона (касательных)**

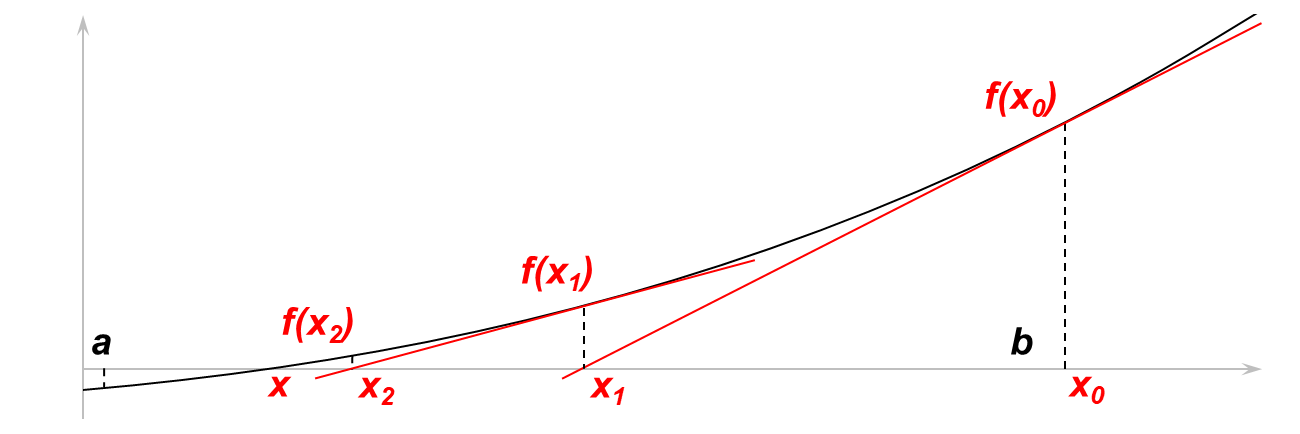
* Последовательность приближений строится следующим образом:



Выбор начальных приближений:

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

***Если неравенства не выполняются – метод не применим***



**Практическая часть**

**Задание 1.**

**Задание:**

Составить программу для решения нелинейного уравнения:

1. Методом половинного деления;

2. Методом простых итераций;

3. Методом Ньютона.

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

**Программная реализация:**

1. **Program** L9\_Z1;

**const**

eps = 1e-2;

**function** f(x:real):real;

**begin**

result:=sqr(sqr(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(dihotomy(1,2, eps));

**end**.

**b) Program** L9\_ Z1;

**const**

eps = 1e-2;

**function** f(x:real): real;

**begin**

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

**end**;

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

**var**

x: real;

**begin**

result:= a;

**repeat**

x := f(result);

result := f(x);

**until** abs(result - x)<=eps;

**end**;

**begin**

writeln(iterations(1, 2, eps));

**end**.

c) **Program** L9\_Z1;

**const**

eps = 1e-2;

**function** f(x:real): real;

**begin**

result:=sqr(sqr(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(newton(1, 2, eps))

**end**.

**Ответ:**

1. 1.9453125;
2. 1.9404754;
3. 1.9404793.

**Задание 2.**

**Задание:**

Составить программу для решения нелинейного уравнения:

1. Методом половинного деления;

2. Методом простых итераций;

3. Методом Ньютона.

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

**Программная реализация:**

1. **Program** L9\_Z2;

**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(dihotomy(0,1, eps));

**end**.

1. **Program** L9\_Z2;

**const**

eps = 1e-3;

**function** g(x: real): real;

**begin**

result := (exp(x)\*x-exp(x)+2)/ (exp(x)+1);

**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(iterations(0, 1, eps))

**end**.

1. **Program** L9\_Z2;

**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(newton(0, 1, eps))

**end**.

**Ответ:**

1. 0.4423828125;
2. 0.4428544010;
3. 0.4428544010.

**Задание 3.**

**Задание:**

Составить программу для решения нелинейного уравнения:

1. Методом половинного деления;

2. Методом простых итераций;

3. Методом Ньютона.

Интервал [0.5; 1.5], допустимая точность 0.2\*10-4;

**Программная реализация**

1. **Program** L9\_Z3;

**const**

eps=0.2e-4;

**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(dihotomy(0.5, 1.5, eps));

**end**.

1. **Program** L9\_Z3;

**const**

eps = 0.2e-4;

**function** g(x:real):real;

**begin**

result:=(1+x-x\*(ln(x)+x))/(x+1);

**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(iterations(0.5, 1.5, eps));

**end**.

1. **Program** L9\_Z3;

**const**

eps = 0.2e-4;

**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(newton(0.5, 1.5, eps))

**end**.

**Ответ:**

1. 0.56715393066;
2. 0.56715393066;
3. 0.56714329039.

**Задание 4.**

**Задание:**

Составить программу для решения нелинейного уравнения:

1. Методом половинного деления;

2. Методом простых итераций;

3. Методом Ньютона.

Интервал [0.2; 1.5], допустимая точность 0.5\*10-4;

**Программная реализация:**

1. **Program** L9\_Z4;

**const**

eps=0.5e-4;

**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(dihotomy(0.2,1.5, eps));

**end**.

1. **Program** L9\_Z4;

**const**

eps=0.5e-4;

**function** g(x:real):real;

**begin**

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

**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(iterations(0.2, 1.5, eps))

**end**.

1. **Program** L9\_Z4;

**const**

eps=0.5e-4;

**function** f(x:real):real;

**begin**

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

**end**;

**function** f1(x:real):real;

**begin**

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

**end**;

**function** f2(x:real):real;

**begin**

result:=-0.01\*exp(-0.1\*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(newton(0.5, 1.5, eps));

**end**.

**Ответ:**

1. 0.47671813964;
2. 0.47672308600;
3. Метод Ньютона, решений нет! 0

**Выводы**

В ходе лабораторной работы научились решать нелинейные уравнения итерационными методами:методом половинного деления, методом простых итераций и методом Ньютона.