**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 Ю.Н.Милованова

(Подпись)

16.04.2020 г.

Отчет принят:

Преподаватель

доцент ОХИ ИШПР, к.т.н. В.А. Чузлов

(Подпись)

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

Томск 2020 г.

**Цель работы:** ознакомиться с основными способами итерационных методов решения нелинейных уравнений, писать программы с их использованием.

**Теоретическая часть**

Выделено три основных метода: метод деления отрезка пополам, метод простых итераций и метод Ньютона (касательных), каждый из которых может применяться для решения нелинейных уравнений.

**Практическая часть**

**Задание:** составить программу для решения нелинейных уравнений методом половинного деления, простых итераций и методом Ньютона:

1. **Исходные данные**: , Интервал [1; 2], допустимая точность 10-2.

**Программная реализация**:

Метод деления отрезка пополам:

**Program** lb911;

**const**

eps = 1e-2;

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

**begin**

result:=x\*x\*x\*x+3\*x-20

**end**;

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

**var**

x:real;

**begin**

**repeat**

x:=(a+b)/2;

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

a:=x

**else**

b:=x

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

result:=x

**end**;

**begin**

writeln (noteee(1,2,eps))

**end**.

Ответ: 1.9453125

Метод простых итераций:

**Program** lb912;

**const**

eps=1e-2;

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

**begin**

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

**end**;

**function** ogo (c,d: real; eps: real):real;

**var**

x: real;

**begin**

result:=c;

**repeat**

x:=g(result);

result:=g(x)

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

**end**;

**begin**

writeln(ogo(1,2,eps))

**end**.

**Ответ:** 1.94037733840934

Метод Ньютона:

**Program** lb913;

**const**

eps=1e-2;

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

**begin**

result:=x\*x\*x\*x+3\*x-20

**end**;

**function** klas1 (x:real):real;

**begin**

result:= 4\*x\*x\*x+3

**end**;

**function** klas2 (x:real):real;

**begin**

result:=12\*x\*x;

**end**;

**function** metod (c,d:real; eps:real):real;

**var**

x:real;

**begin**

**if** klas(c)\* klas2(c) > 0 **then**

result:=c

**else**

**if** klas(d)\* klas2(d) > 0 **then**

result:=d

**else**

**begin**

writeln ('Метод Ньютона не применим ');

**exit**

**end**;

**repeat**

x:=result;

result:=x-klas(x)/klas1(x)

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

**end**;

**begin**

writeln(metod(1,2,eps))

**end**.

**Ответ:** 1.94047935224908

1. **Исходные данные: ,** Интервал [0; 1], допустимая точность 10-3.

**Программная реализация:**

Метод деления отрезка пополам:

**Program** lb921;

**const**

eps = 1e-3;

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

**begin**

result:=exp(x)+x-2

**end**;

**function** noteee(d,f: real; eps: real): real;

**var**

x:real;

**begin**

**repeat**

x:=(d+f)/2;

**if** eee(d)\*eee(x) > 0 **then**

d:=x

**else**

f:=x

**until** (abs(d - f) <= eps) **or** (eee(x) = 0);

result:=x

**end**;

**begin**

writeln (noteee(0,1,eps))

**end**.

**Ответ:** 0.4423828125

Метод простых итераций:

**Program** lb922;

**const**

eps=1e-3;

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

**begin**

result:= ln(2-x)

**end**;

**function** f (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(f(0,1,eps))

**end**.

**Ответ:** 0.442509950010955

Метод Ньютона:

**Program** lb923;

**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** metod (c,d:real; eps:real):real;

**var**

x:real;

**begin**

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

result:=c

**else**

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

result:=d

**else**

**begin**

writeln ('Метод Ньютона не применим ');

**exit**

**end**;

**repeat**

x:=result;

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

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

**end**;

**begin**

writeln(metod(0,1,eps))

**end**.

**Ответ:** 0.442854401004033

1. **Исходные данные: ,** Интервал [0.5; 1.5], допустимая точность 0.2\*10-4.

**Программная реализация:**

Метод деления отрезком пополам:

**Program** lb931;

**const**

eps = 0.5e-4;

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

**begin**

result:=ln(x)+x

**end**;

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

**var**

x:real;

**begin**

**repeat**

x:=(a+b)/2;

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

a:=x

**else**

b:=x

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

result:=x

**end**;

**begin**

writeln (lp(0.5,1.5,eps))

**end**.

**Ответ:** 0.567169189453125

Метод простых итераций:

**Program** lb932;

**const**

eps=0.2e-4;

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

**begin**

result:= exp(-x)

**end**;

**function** ogo (c,d: real; eps: real):real;

**var**

x: real;

**begin**

result:=c;

**repeat**

x:=g(result);

result:=g(x)

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

**end**;

**begin**

writeln(ogo(0.5,1.5,eps))

**end**.

**Ответ:** 0.567140763269807

Метод Ньютона:

**Program** lb933;

**const**

eps=0.5e-4;

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

**begin**

result:=ln(x)+x

**end**;

**function** h1 (x:real):real;

**begin**

result:= (1/x)+1

**end**;

**function** h2 (x:real):real;

**begin**

result:=1/(x\*x)

**end**;

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

**var**

x:real;

**begin**

**if** h(a)\* h2(a) > 0 **then**

result:=a

**else**

**if** h(b)\*h2(b) > 0 **then**

result:=b

**else**

**begin**

writeln ('Метод Ньютона не применим');

**exit**

**end**;

**repeat**

x:=result;

result:=x-h(x)/h1(x)

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

**end**;

**begin**

writeln(metod(0.5,1.5,eps))

**end**.

**Ответ:** 0.567143290409753

1. **Исходные данные: ,** Интервал [0.2; 1.5], допустимая точность 0.5\*10-4.

**Программная реализация:**

Метод деления отрезком пополам:

**Program** lb941;

**const**

eps = 0.2e-4;

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

**begin**

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

**end**;

**function** b(c,d: real; eps: real): real;

**var**

x:real;

**begin**

**repeat**

x:=(c+d)/2;

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

c:=x

**else**

d:=x

**until** (abs(c - d) <= eps) **or** (a(x) = 0);

result:=x

**end**;

**begin**

writeln (b(0.2,1.5,eps))

**end**.

**Ответ:** 0.476737976074219

Метод простых итераций:

**Program** lb942;

**const**

eps=0.2e-2;

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

**begin**

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

**end**;

**function** ogo (c,d: real; eps: real):real;

**var**

x: real;

**begin**

result:=c;

**repeat**

x:=g(result);

result:=g(x)

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

**end**;

**begin**

writeln(ogo(0.2,1.5,eps))

**end**.

**Ответ:** 0.476721637710739

Метод Ньютона:

**Program** lb943;

**const**

eps=0.2e-4;

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

**begin**

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

**end**;

**function** yyy1 (x:real):real;

**begin**

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

**end**;

**function** yyy2 (x:real):real;

**begin**

result:=0.01\*exp(-0.1\*x)

**end**;

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

**var**

x:real;

**begin**

**if** yyy(a)\* yyy2(a) > 0 **then**

result:=a

**else**

**if** yyy(b)\*yyy2(b) > 0 **then**

result:=b

**else**

**begin**

writeln ('Метод Ньютона не применим');

**exit**

**end**;

**repeat**

x:=result;

result:=x-yyy(x)/yyy1(x)

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

**end**;

**begin**

writeln(metod(0.2,1.5,eps))

**end**.

**Ответ:** 0.476724042833792

**Выводы**

В ходе лабораторной работы были изучены основные итерационные методы решения линейных уравнений: метод деления отрезка пополам, простых итераций, Ньютона. При решении одной и той же задачи, с использованием трех этих методов, получаются ответы, имеющие незначительные различия, что, в целом, может свидетельствовать о возможности применения любого метода для решения задачи. За исключением метода Ньютона, который в некоторых случаях может быть не применим.