**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 С.Э. Забанова

(Подпись)

\_\_24\_\_\_ \_\_\_\_апрель\_\_\_\_\_\_\_ 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 с использованием итерационных методов.

**Теоретическая часть**

Метод деления отрезка пополам

Идея метода:

* Найти середину отрезка [*a, b*]: *c=*(*a+b*)*/2*. Корень остался на одной из частей: [*a, c*] или [*c, b*]. Если *f*(*a*)*\* f*(*с*)*<0****,***токорень попал на отрезок [*a, c*], тогда деление отрезка можно повторить, приняв в качестве нового правого конца точку *c*, т.е. *b=c*. В противном случае корень попал на половину [*c, b*], и необходимо изменить значение левого конца отрезка: *a=c*. Поскольку корень всегда заключен внутри отрезка, итерационный процесс можно останавливать, если длина отрезка станет меньше заданной точности: |*b – a|<**ε*

Метод простых итераций

Идея метода:

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

Метод Ньютона (касательных)

* Метод определяется формулой:



* 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**

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

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

**Программная реализация**

1.Метод деления отрезка пополам:

**program** LB\_09\_1;

**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(dihotomy(1, 2, eps))

**end**.

**Ответ**

X = 1.9453125

2. Метод простых 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** LB9\_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(newton(1, 2, eps))

**end**.

**Ответ**

X = 1.94047935224908

**Задание 2**

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

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

**Программная реализация:**

Метод деления отрезка пополам

**program** LB\_09\_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(dihotomy(0, 1, eps))

**end**.

**Ответ**

X = 0.4423828125

2. Метод простых итераций

**program** LB9\_2;

**const**

eps = 1e-3;

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

**begin**

result := x - (exp(x) + x - 2) / 3.7183;

**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 = 0.442795159460669

3. Метод Ньютона

**rogram** LB9\_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(newton(0, 1, eps))

**end**.

**Ответ**

Х =0.442854401004033

**Задание 3**

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

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

**Программная реализация:**

1.Метод деления отрезка пополам

**program** LB\_09\_3;

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

**end**.

**Ответ**

X = 0.567153930664063

2.Метод простых итераций

**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** LB9\_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 / sqrt(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**

Х = 0.567143290399369

**Задание 4**

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

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

**Программная реализация**

1.Метод деления отрезка пополам

**program** LB\_09\_4;

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

**end**.

**Ответ**

Х = 0.476718139648438

2.Метод простых итераций

**program** LB9\_4;

**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** LB\_9\_4;

**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 данной лабораторной работы были итерационные методы решения нелинейных уравнений, а также составлены программы с их использованием.

**Выводы**

В ходе работы …