

## Численные методы решения нелинейных уравнений

### Код программы:

```
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
#include <locale.h>
float f(float x);
float df(float x);
void glm();
void dih();
void nuton();
void horda();
void end();
int main()
{
    setlocale(LC_ALL, "");
    glm();
    getchar();
    return 0;
}
void glm()
{
    int i;
    printf("\nГлавное меню\n\nРешение нелинейного уравнения:\n1. Методом  
дихотомии\n2. Методом Ньютона\n3. Методом хорд\n4. Выйти из  
программы\n\nВыберите пункт меню: ");
    scanf("%d",&i);
    printf("\n");
    switch(i)
    {
        case 1: dih(); break;
        case 2: nuton(); break;
        case 3: horda(); break;
        case 4: end(); break;
    }
}
float f(float x)
{
    float y;
```

```

/*y = pow(x,3)-12*x-8; */[-3,4]*/
if (x==0) x=0,000001;
y = x*log(fabs(x+1))/log(3)-1; */[0,10]*/
return y;
}
float df(float x)
{
    float dy;
    /*dy=36*x-30; */y'=6x-5, y''=6, y'*y''=(6x-5)*6*/
    if (x==0) x=0,000001;
    dy=((x+2)*(x+(x+1)*log(fabs(x+1))))/(pow(x+1,3)*pow(log(3),2));
    /*y'=(x+(x+1)ln(x+1))/((x+1)ln(3)), y''=(x+2)/((x+1)^2*ln(3))*/
    return dy;
}
void dih() /*метод дихотомии*/
{
    float a=0,b=0,c=0,x=0,e=0;
    int k=0;
    printf("Введите интервал для решения уравнения\nВведите a: ");
    scanf("%f",&a);
    printf("Введите b: ");
    scanf("%f",&b);
    printf("Введите точность вычисления: ");
    scanf("%f",&e);
    do
    {
        x=(a+b)/2;
        if(f(a)*f(x)<0) b=x;
        else a=x;
        k++;
    }
    while (b-a>2*e);
    printf("Результат: %g\nКоличество иттераций: %d\n",x,k);
    glm();
}
void nuton() /*метод касательных (Ньютона)*/
{
    float a=0,b=0,c=0,x=0,x0=0,e=0,h=0.0001;
    int k=0;
    printf("Введите интервал для решения уравнения\nВведите a: ");
    scanf("%f",&a);
    printf("Введите b: ");

```

```

scanf("%f",&b);
printf("Введите точность вычисления: ");
scanf("%f",&e);
if (f(a)*df(a)>0) x0=a;
else x0=b;
c=1;
while (c>e)
{
    x=x0-f(x0)*h/(f(x0+h)-f(x0));
    c=fabs(x-x0);
    x0=x;
    k++;
}
printf("Результат: %g\nКоличество итераций: %d\n",x,k);
glm();
}
void horda() /*метод хорд*/
{
    float a=0,b=0,c=0,x=0,x0=0,e=0;
    int k=0;
    printf("Введите интервал для решения уравнения\nВведите a: ");
    scanf("%f",&a);
    printf("Введите b: ");
    scanf("%f",&b);
    printf("Введите точность вычисления: ");
    scanf("%f",&e);
    if (f(a)*df(a)>0)
    {
        x=a;
        c=b;
    }
    else
    {
        x=b;
        c=a;
    }
    do
    {
        x0=x;
        x=x0-f(x0)*(c-x0)/(f(c)-f(x0));
        k++;
    }
}

```

```

while (fabs(x-x0)>e);
printf("Результат: %g\nКоличество иттераций: %d\n",x,k);
glm();
}
void end()
{
    printf("Работа завершена.\n");
}

```

## Результат:

Интерфейс главного меню:

```

Главное меню
Решение нелинейного уравнения:
1. Методом дихотомии
2. Методом Ньютона
3. Методом хорд
4. Выйти из программы
Выберите пункт меню: 1

```

$$x^3-12*x-8=0$$

Ожидаемые результаты (посчитано с помощью wolframalpha.com):

Solutions:

$$x \approx -3.06417777247591$$

$$x \approx -0.694592710667721$$

$$x \approx 3.75877048314363$$

## Метод дихотомии

```

Введите интервал для решения уравнения
Введите a: -3
Введите b: 4
Введите точность вычисления: 0,000001
Результат: -0,694592
Количество иттераций: 22

```

## Метод Ньютона

```
Введите интервал для решения уравнения
Введите a: -3
Введите b: 4
Введите точность вычисления: 0,000001
Результат: 3,75877
Количество итераций: 4
```

Метод хорд

```
Введите интервал для решения уравнения
Введите a: -3
Введите b: 4
Введите точность вычисления: 0,000001
Результат: -3,06418
Количество итераций: 7
```

$$x \cdot \log(x+1) / \log(3) - 1 = 0$$

Ожидаемые результаты (посчитано с помощью wolframalpha.com):

$$x \approx 1.3112942043836504983$$

Метод дихотомии

```
Введите интервал для решения уравнения
Введите a: 0
Введите b: 10
Введите точность вычисления: 0,000001
Результат: 1,31129
Количество итераций: 23
```

Метод Ньютона

```
Введите интервал для решения уравнения
Введите a: 0
Введите b: 10
Введите точность вычисления: 0,000001
Результат: 1,31129
Количество итераций: 6
```

Метод хорд

```
Введите интервал для решения уравнения
Введите a: 0
Введите b: 10
Введите точность вычисления: 0,000001
Результат: 1,31129
Количество итераций: 40
```

## **Вывод**

Самым быстрым методом оказался метод Ньютона или метод касательных. За наименьшее количество итераций методом Ньютона был получен такой же точный результат, как и другими методами.

Объяснить различность результатов для первого уравнения я могу тем, что суть методов разная, и они показывают только первый встретившийся результат.