

# Численные методы решения нелинейных уравнений.

Фролов А.А., Курылев Г.А., 2к, ИВТ-2

Ссылка на скринкаст:

[Я.Диск](#)

Ссылки на программу (github):

[nonlinear\\_eq.py](#)

[interface.py](#)

[main.py](#)

Научиться решать нелинейные уравнения используя методы:

метод касательных (Ньютона)

метод дихотомии (деления отрезка пополам)

метод хорд.

Реализовать решение на используемом языке программирования.

ПК, язык программирования - **python**

## Общие функции и уравнения

$$x^3 - 12x - 0$$

Отрезок :  $[-1, 1]$

$$x^3 + 4x - 6$$

Отрезок :  $[-0.1, 1.2]$

*nonlinear\_eq.py*

```
1  #Первая вычисляемая функция
2  def f1(x):
3      return x**3 - 12 * x - 8
4
5  #Вторая вычисляемая функция
6  def f2(x):
7      return x**3 + 4 * x - 6
8
9  #Вычисление первой производной
10 def diff(f, x, h=1e-6):
11     return (f(x + h) - f(x - h)) / (2 * h)
12
13 #Вычисление второй производной
14 def second_diff(f, x, h=1e-6):
15     return (f(x + h) - 2 * f(x) + f(x - h)) / (h**2)
```

## Метод касательных

*nonlinear\_eq.py*

```
1  def tangent_method(f, a, b):
2
3      if f(a) * f(b) > 0:
```

```

4         return None
5
6     x0 = a
7
8     while True:
9         x1 = x0 - (f(x0) / diff(f, x0))
10        if abs(x1 - x0) ≤ 10**-6:
11            return x1
12        x0 = x1

```

## Метод дихотомии

*nonlinear\_eq.py*

```

1  def bisec_method(f, a, b):
2
3      if f(a) * f(b) > 0:
4          return None
5
6      while True:
7          x = (a + b) / 2
8
9          if f(x) == 0:
10             return x
11          elif f(a) * f(x) < 0:
12              b = x
13          elif f(x) * f(b) < 0:
14              a = x
15          else:
16              return None
17
18          if abs(a - b) ≤ 10**-6:
19              return x

```

## Метод хорд

*nonlinear\_eq.py*

```

1  def chord_method(f, a, b):
2
3      if f(a) * f(b) > 0:
4          return None
5
6      f_diff = diff(f, a)
7      f_second_diff = second_diff(f, a)
8
9      if f_diff * f_second_diff > 0:
10         x0 = a
11         C = b
12     elif f_diff * f_second_diff < 0:
13         x0 = b
14         C = a
15
16     while abs(f(x0)) > 1e-6:
17         x0 = x0 - (f(x0) / (f(C) - f(x0))) * (C - x0)
18
19     return x0

```

## Интерфейс и результаты

Дополнено пунктом 4. Нелинейное уравнение

```
== Главное меню ==  
1. Численное интегрирование  
2. Дифференциальное уравнение  
3. Элементарная функция  
4. Нелинейное уравнение  
0. Завершить работу
```

Выберите пункт меню: |

Сначала выбирается уравнение

```
=== Нелинейные уравнения ===  
1.  $x^3 - 12x - 8$ ,  $[-1, 1]$   
2.  $x^3 + 4x - 6$ ,  $[-0.1, 1.2]$   
0. Назад  
Выберите уравнение: |
```

Выводятся результаты решения всеми методами

Первое уравнение:

```
Результаты вычислений:  
Метод касательных: -0.694593  
Метод бисекции: -0.694592  
Метод хорд: -0.694593
```

Второе уравнение:

```
Результаты вычислений:  
Метод касательных: 1.134728  
Метод бисекции: 1.134729  
Метод хорд: 1.134729
```

Чтобы продолжить нажмите enter ...|