# Лабораторная работа № 2. Численные методы решения нелинейных уравнений

## Инструменты

Использовала интерпретатор, встроенный в Python.

## Задача

Решить нелинейное уравнение методами дихотомии, хорд и Ньютона.

на отрезке от -10 до 10 с точностью .

## Код программы

import math  
  
A = -10  
B = 10  
X0\_newton = 2  
E = 10 \*\* (-6)  
  
  
def function(x):  
 return x \*\* 2 - 3 + 0.5 \*\* x  
  
  
def derivative\_function(x):  
 return 2 \* x + 0.5 \*\* x \* math.log(0.5)  
  
  
def menu(): # Меню  
 print("Решить нелинейное уравнение методом:\n1. Дихотомии\n2. Хорд\n"  
 "3. Ньютона\n4. Выйти из программы")  
 user\_input = int(input())  
 if user\_input == 1:  
 dichotomy()  
 elif user\_input == 2:  
 chord()  
 elif user\_input == 3:  
 newton()  
 elif user\_input == 4:  
 pass  
 else:  
 print("Введите число от 1 до 4.\n")  
 menu()  
  
  
def dichotomy(): # Метод половинного деления (дихотомии)  
 a = A  
 b = B  
 x = (a + b) / 2  
 while abs(function(x)) >= E:  
 x = (a + b) / 2  
 a, b = (x, b) if function(x) \* function(b) < 0 else (a, x)  
 result = (a + b) / 2  
 print("Результат при решении методом дихотомии: {:.5f}".format(result) + ".\n")  
 menu()  
  
  
def chord(): # Метод хорд  
 previous\_x = A  
 current\_x = B  
 while abs(current\_x - previous\_x) >= E:  
 x = current\_x - ((current\_x - previous\_x) \* function(current\_x)) / \  
 (function(current\_x) - function(previous\_x))  
 previous\_x = current\_x  
 current\_x = x  
 print("Результат при решении методом хорд: {:.5f}".format(current\_x) + ".\n")  
 menu()  
  
  
def newton(): # Метод Ньютона  
 previous\_x = X0\_newton  
 current\_x = previous\_x - function(previous\_x) / derivative\_function(previous\_x)  
 while abs(current\_x - previous\_x) >= E:  
 previous\_x = current\_x  
 current\_x = previous\_x - function(previous\_x) / derivative\_function(previous\_x)  
 print("Результат при решении методом Ньютона: {:.5f}".format(current\_x) + ".\n")  
 menu()  
  
  
menu()

## Результаты







## Анализ результатов

Все методы дают одинаковый результат — 1,63658. Но если выводить больше знаков после запятой, то получится, что метод хорд точнее метода дихотомии, а метод Ньютона точнее метода хорд. Также программа может вычислить только один корень, хотя моё уравнение [имеет два корня](https://www.wolframalpha.com/input/?i=solve%28x%5E2-3%2B%280.5%29%5Ex%2Cx%29) на заданном участке.