Расчётно-графическая работа по функциональному анализу

В приведённых ниже вариантах $k \in \{1, 3, 6, 7, 8, 9, 10, 12\}$ — номер группы, $l = 15$, если номер студента в списке группы не больше 10, $l = 10$, если номер студента в списке группы от 11 до 20, $l = 5$, если номер студента не меньше 21. Вариант выбирается как остаток от деления номера по списку группы на 10.

Задание I

Докажите, что приведённое ниже отображение $T : C[0; 1] \to C[0; 1]$ является сжимающим. Определите число итераций, необходимое для поиска неподвижной точки этого отображения с точностью $\varepsilon \in \{10^{-1}, 10^{-2}, 10^{-3}\}$ с помощью метода сжимающих отображений. С помощью вычислительной техники постройте график функции, являющейся неподвижной точкой отображения T. Проверьте результаты при различных значениях ε и различных начальных приближениях в методе сжимающих отображений. Продемонстрируйте несколько графиков, получающихся при промежуточных вычислениях.

Вариант 7

7) $T(x) = \begin{cases} -\frac{1}{1+k}x(3t), & 0 \leqslant t \leqslant \frac{1}{3}; \\ -\frac{1}{1+k}\left(1 + x(1) - \cos\left(\frac{3l\pi}{5}\left(t - \frac{1}{3}\right)\right)\right), & \frac{1}{3} < t < \frac{2}{3}; \\ -\frac{1}{1+k}x(3t-2) + C, & \frac{2}{3} \leqslant t \leqslant 1, \end{cases}$ где C — константа такая, что $T(x)$ — непрерывная функция;

вариант 7

$$T(x)=\begin{cases} -\frac{1}{1+k}\,x(3t), & 0\le t\le \frac{1}{3}\\ -\frac{1}{1+k}\left(1+x(1)-\cos\left(\frac{3l\pi}{5}\left(t-\frac{1}{3}\right)\right)\right), & \frac{1}{3}<t<\frac{2}{3}\\ -\frac{1}{1+k}\,x(3t-2)+C, & \frac{2}{3}\le t\le 1\end{cases}$$

$k=7,\ l=10$

$$T(x)=\begin{cases} -\frac{1}{8}\,x(3t), & 0\le t\le \frac{1}{3}\\ -\frac{1}{8}\left(1+x(1)-\cos\left(6\pi\left(t-\frac{1}{3}\right)\right)\right), & \frac{1}{3}<t<\frac{2}{3}\\ -\frac{1}{8}\,x(3t-2)+C, & \frac{2}{3}\le t\le 1\end{cases}$$

Константу C найдем из условия непрерывности

$$-\frac{1}{8}\left(1+x(1)-\cos\left(6\pi\left(t-\frac{1}{3}\right)\right)\right)\Big|_{t=\frac{2}{3}}=-\frac{1}{8}\,x(3t-2)\Big|_{t=\frac{2}{3}}+C \Rightarrow$$

$$C=-\frac{1}{8}\left(1+x(1)-\cos 2\pi-x(0)\right)=-\frac{1}{8}\left(x(1)-x(0)\right)$$

$$T(x)=\begin{cases} -\frac{1}{8}\,x(3t), & 0\le t\le \frac{1}{3}\\ -\frac{1}{8}\left(1+x(1)-\cos\left(6\pi\left(t-\frac{1}{3}\right)\right)\right), & \frac{1}{3}<t<\frac{2}{3}\\ -\frac{1}{8}\left(x(3t-2)+x(1)-x(0)\right), & \frac{2}{3}\le t\le 1\end{cases}$$

Докажем, что обображение $T: C[0,1] \to C[0,1]$ является сжимающим

Нужно показать, что $\forall\, x(t), y(t) \in C[0,1] \Rightarrow$

$$\Rightarrow \rho(T(x(t)), T(y(t))) \le \alpha \cdot \rho(x(t), y(t)),$$

где $\alpha \in (0, 1)$, $\rho(x(t), y(t)) = \|x - y\| = \max\limits_{t\in[0,1]} |x(t) - y(t)|$.

$$\rho(T(x(t)), T(y(t))) = \max \{$$

$$\max_{t\in[0,\frac{1}{3}]} \left|-\frac{1}{8}x(3t) + \frac{1}{8}y(3t)\right|,$$

$$\max_{t\in(\frac{1}{3},\frac{2}{3})} \left|-\frac{1}{8}\left(1 + x(1) - \cos\left(6\pi\left(t - \frac{1}{3}\right)\right)\right) + \frac{1}{8}\left(1 + y(1) - \cos\left(6\pi\left(t - \frac{1}{3}\right)\right)\right)\right|,$$

$$\max_{t\in[\frac{2}{3},1]} \left|-\frac{1}{8}(x(3t-2) + x(1) - x(0)) + \frac{1}{8}(y(3t-2) + y(1) - y(0))\right|$$

$$\}$$

Рассмотрим по отдельности:

1) $\max\limits_{t\in[0,\frac{1}{3}]} \left|-\frac{1}{8}x(3t) + \frac{1}{8}y(3t)\right| = \frac{1}{8}\max\limits_{s\in[0,1]} |x(s) - y(s)|$, $s = 3t$

2) $\max\limits_{t\in(\frac{1}{3},\frac{2}{3})} \left|-\frac{1}{8}(x(1) - y(1))\right| = \frac{1}{8}|x(1) - y(1)|$

3) $\max\limits_{t\in[\frac{2}{3},1]} \left|-\frac{1}{8}(x(3t-2) + x(1) - x(0)) + \frac{1}{8}(y(3t-2) + y(1) - y(0))\right| =$

$= \frac{1}{8}\max\limits_{t\in[\frac{2}{3},1]} |x(3t-2) - y(3t-2) + x(1) - y(1) - x(0) + y(0)| = [s = 3t-2] \le$

$\le \frac{1}{8}\max\limits_{s\in[0,1]} \left(|x(s) - y(s)| + |x(1) - y(1)| + |x(0) - y(0)|\right) =$

$= \frac{1}{8}\left(\max\limits_{s\in[0,1]} |x(s) - y(s)| + |x(1) - y(1)| + |x(0) - y(0)|\right) \le \frac{3}{8}\max\limits_{s\in[0,1]} |x(s) - y(s)|$

Таким образом:

$\rho(T(x(t)), T(y(t))) \le \max \{$

$\frac{1}{8} \max\limits_{s\in[0,1]} |x(s) - y(s)|,$

$\frac{1}{8} |x(1) - y(1)|,$

$\frac{3}{8} \max\limits_{s\in[0,1]} |x(s) - y(s)|$

$\} = \frac{3}{8} \max\limits_{s\in[0,1]} |x(s) - y(s)| = \frac{3}{8} \cdot \rho(x(t), y(t))$

$\Rightarrow \alpha = \frac{3}{8} \in (0, 1) \Rightarrow$ отображение $T(x)$ - сжимающее, ч.т.д.

Определим число итераций, необходимое для поиска неподвижной точки этого отображения с точностью $\varepsilon \in \{0,1;\ 0,01;\ 0,001\}$ с помощью метода сжимающих отображений.

$$\rho(x^{(n)}, x^{*}) \le \frac{\alpha^n}{1-\alpha}\rho(x^{(1)}, x^{(0)}) < \varepsilon$$

$$\Rightarrow n > \frac{1}{\ln\alpha} \ln\frac{\varepsilon(1-\alpha)}{\rho(x^{(1)}, x^{(0)})}$$

В качестве начального приближения примем $x^{(0)}(t) = 0$.
Тогда $x^{(1)}(t) = T(x^{(0)})$.

$$x^{(1)}(t) = \begin{cases} 0 &, 0 \le t \le \frac{1}{3} \\ -\frac{1}{8}\left(1 - \cos\left(6\pi\left(t - \frac{1}{3}\right)\right)\right) &, \frac{1}{3} < t < \frac{2}{3} \\ 0 &, \frac{2}{3} \le t \le 1 \end{cases}$$

$$\rho(x^{(1)}(t), x^{(0)}(t)) = \max_{t\in[0,1]} |x^{(1)}(t) - x^{(0)}(t)| =$$

$$= \max_{t\in(\frac{1}{3},\frac{2}{3})} \left|-\frac{1}{8}\left(1-\cos\left(6\pi\left(t-\frac{1}{3}\right)\right)\right)\right| = \frac{1}{8}\max_{t\in(\frac{1}{3},\frac{2}{3})}\left|1-\cos\left(6\pi\left(t-\frac{1}{3}\right)\right)\right|$$

но т. к. $\cos x \in [-1, 1]$, то $(1-\cos(x)) \in [0, 2]$

$\Rightarrow \max|1-\cos x|$ будет достигаться в $\cos x = -1 \Leftrightarrow x = \pi + 2\pi k$

Принимает ли выражение $6\pi(t-\frac{1}{3})$ значение $\pi + 2\pi k$ на $(\frac{1}{3}, \frac{2}{3})$?

$t \in (\frac{1}{3}, \frac{2}{3}) \Rightarrow (t - \frac{1}{3}) \in (0, \frac{1}{3}) \Rightarrow 6\pi(t-\frac{1}{3}) \in (0, 2\pi)$ - принимает

$\Rightarrow \rho(x^{(1)}(t), x^{(0)}(t)) = \frac{1}{8}\cdot 2 = \frac{1}{4}$.

Итого, количество итераций:

$$n_{\varepsilon_1} > \frac{1}{\ln\frac{3}{8}} \cdot \ln\frac{0{,}1\cdot\frac{5}{8}}{\frac{1}{4}} \approx 1{,}4 \quad \Rightarrow \quad n_{\varepsilon_1} = 2$$

$$n_{\varepsilon_2} > \frac{1}{\ln\frac{3}{8}} \cdot \ln\frac{0{,}01\cdot\frac{5}{8}}{\frac{1}{4}} \approx 3{,}8 \quad \Rightarrow \quad n_{\varepsilon_2} = 4$$

$$n_{\varepsilon_3} > \frac{1}{\ln\frac{3}{8}} \cdot \ln\frac{0{,}001\cdot\frac{5}{8}}{\frac{1}{4}} \approx 6{,}1 \quad \Rightarrow \quad n_{\varepsilon_3} = 7$$

Краткое описание

Зададим начальное приближение функции x0, массив точек ts, в которых будем вычислять функцию, а также определим словари prevX и curX, которые по порядковому номеру точки из массива ts хранят значение функции в этой точке. Словарь prevX хранит значение для функции, вычисленной на предыдущей итерации (изначально туда записываются значения функции x0), а словарь curX вычисляется на текущей итерации. В конце каждой итерации вычисляется текущая погрешность, и, если она не превышает заданный эпсилон, цикл завершается.

Код

```
from random import random
import numpy as np
import matplotlib.pyplot as plt

ALPHA = 0.375
COUNT_POINTS = int(1e4 - 1)
EPS = 1e-1

def plot(x, y, iter):
    plt.plot(x, y, linestyle='-', color=(random(), random(), random()),
label=f"{iter} итераций")

def x0(t):
    return 0
# def x0(t):
#     return t
# def x0(t):
#     return t ** 2 - t

ts = np.linspace(0, 1, COUNT_POINTS)
prevX = dict()
for i in range(COUNT_POINTS):
    prevX[i] = x0(ts[i])

iter = 1
while (True):
    curX = dict()
    isPlot = iter > 4

    # Строим функцию
    for i in range(COUNT_POINTS):
        t = i/(COUNT_POINTS-1)
        prevX_0 = prevX[0]
        prevX_1 = prevX[COUNT_POINTS-1]

        if 0 <= i < COUNT_POINTS/3:
            curX[i] = -1/8 * prevX[3 * i]
```

```
        elif COUNT_POINTS/3 <= i < 2*COUNT_POINTS/3:
            curX[i] = -1/8 * (1 + prevX_1 - np.cos(6 * np.pi * (t - 1/3)))
        elif 2*COUNT_POINTS/3 <= i < COUNT_POINTS:
            curX[i] = -1/8 * (prevX[3 * i - 2*COUNT_POINTS] + prevX_1 - prevX_0)

    if (isPlot):
        plot(ts, curX.values(), iter)

    # Расстояние
    rho = 0
    for i in range(COUNT_POINTS):
        rho = max(rho, abs(curX[i] - x0(ts[i])))

    # Погрешность
    curEps = ALPHA ** iter / (1 - ALPHA) * rho

    if (curEps < EPS):
        if (not isPlot):
            plot(ts, curX.values(), iter)
        break

    iter += 1
    prevX = curX

print(f"{iter} итераций")

plt.title(f"Неподвижная точка (точность {EPS})")
plt.legend()

plt.grid(True)
plt.show()
```

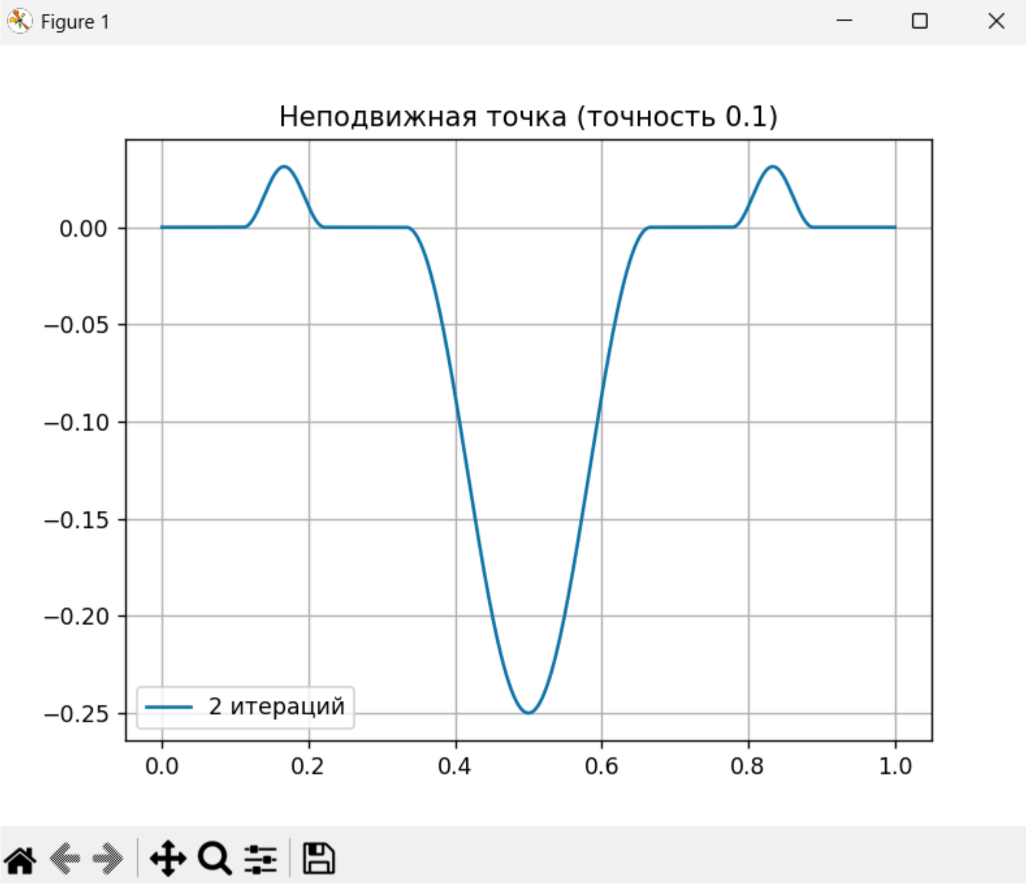
Поиск неподвижной точки

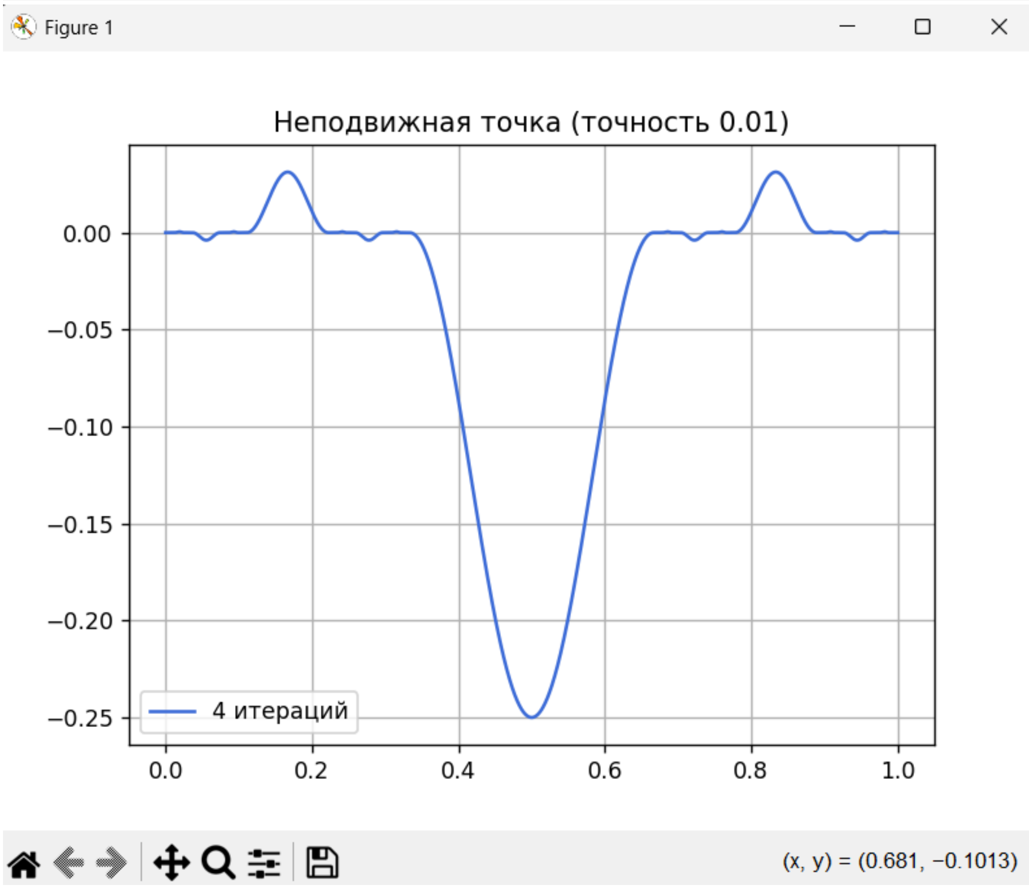
Далее программа тестировалась с различными начальными приближениями и различной точностью. На скриншотах видно, за сколько итераций удалось достичь нужной точности при разных начальных приближениях.

Заметим, что количество итераций, сделанных численно, совпало с теоретической оценкой.

После первого приближения нулем стало видно, что неподвижная точка отображения походит на параболу t^2 - t. Поэтому, задав начальное приближение этой параболой, получили меньшее число итераций для точности 0.001 по сравнению с начальным приближением нулем.

Начальное приближение x0(t) = 0:





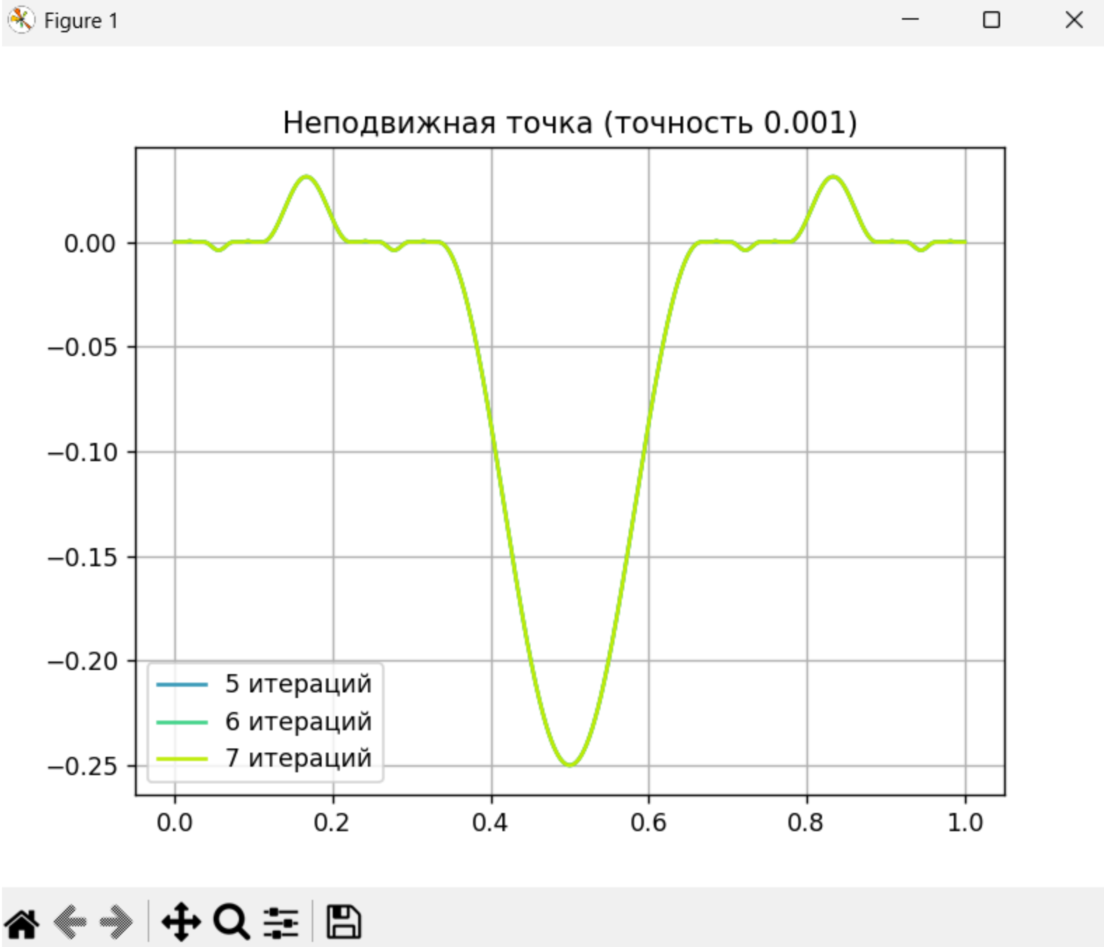


Figure 1
Неподвижная точка (точность 0.001)
0.00
−0.05
−0.10
−0.15
−0.20
−0.25
0.0
0.2
0.4
0.6
0.8
1.0
5 итераций
6 итераций
7 итераций

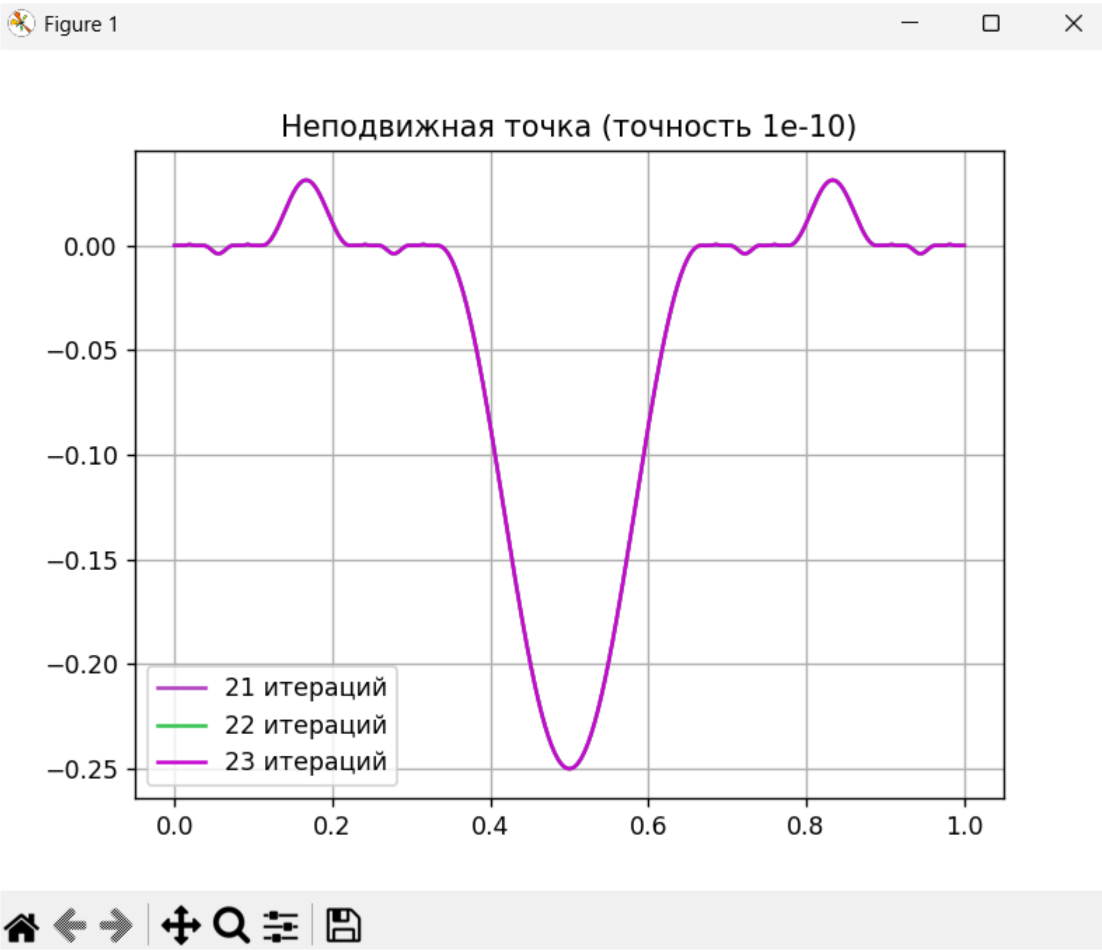
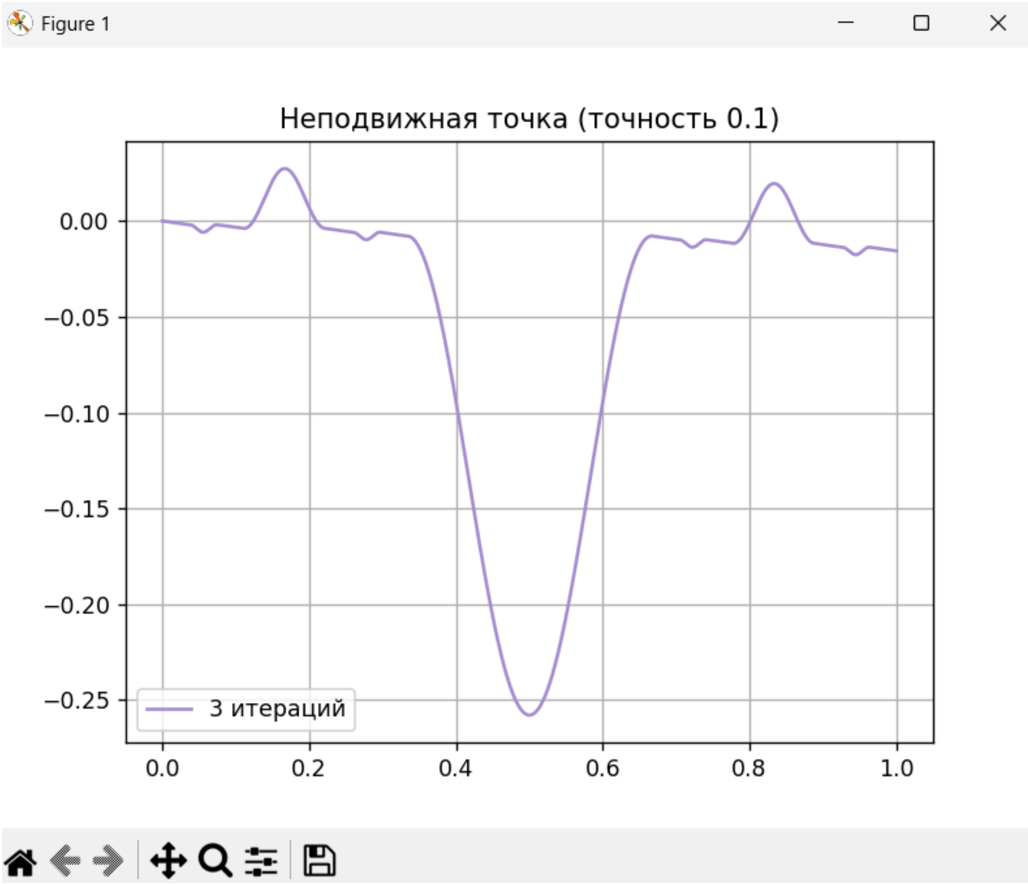
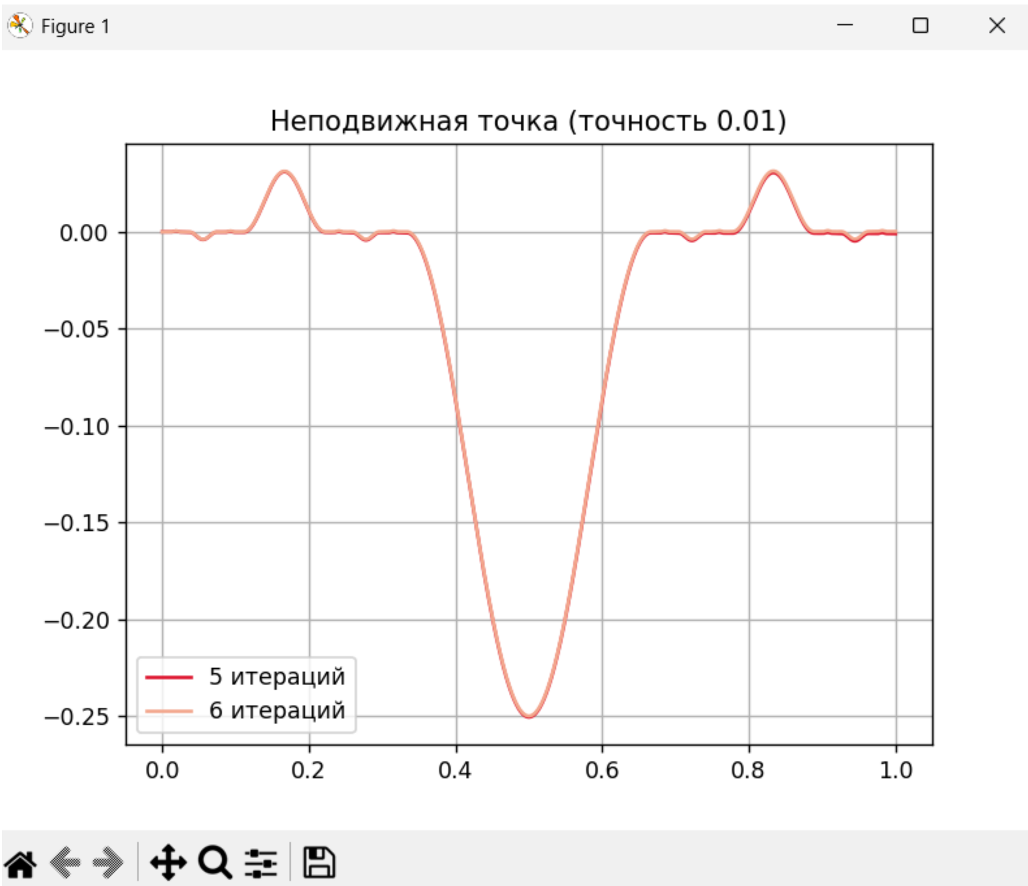


Figure 1
Неподвижная точка (точность 1e-10)
0.00
−0.05
−0.10
−0.15
−0.20
−0.25
0.0
0.2
0.4
0.6
0.8
1.0
21 итераций
22 итераций
23 итераций

Начальное приближение x0(t) = t:





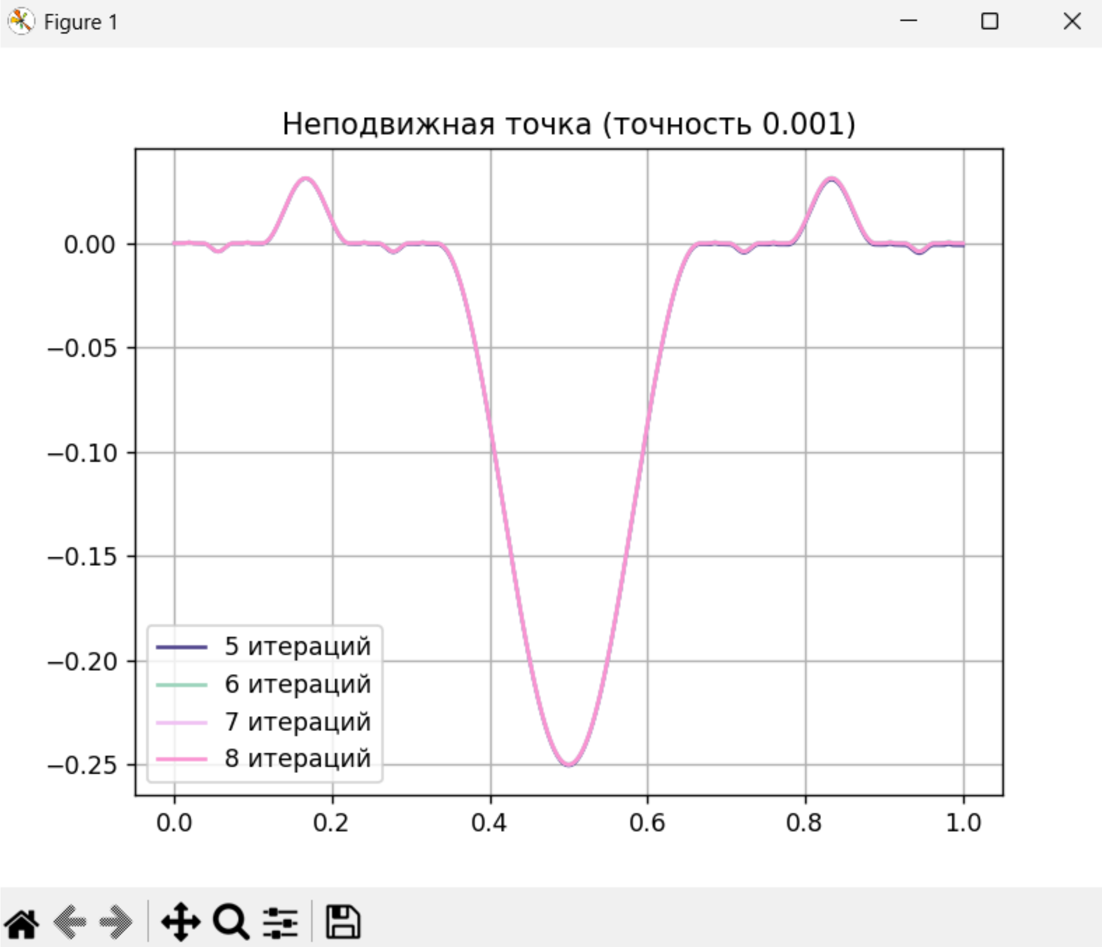


Figure 1
Неподвижная точка (точность 0.001)
0.00
−0.05
−0.10
−0.15
−0.20
−0.25
0.0
0.2
0.4
0.6
0.8
1.0
5 итераций
6 итераций
7 итераций
8 итераций

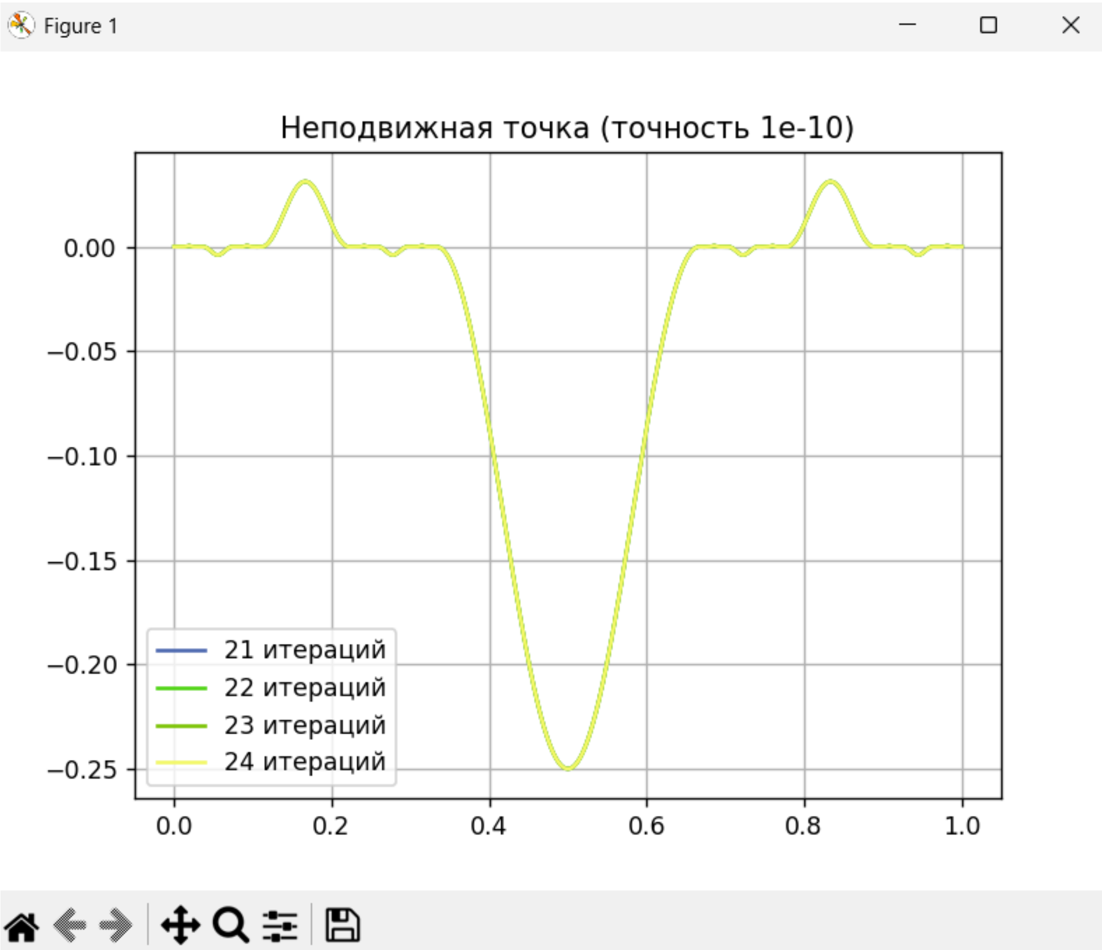
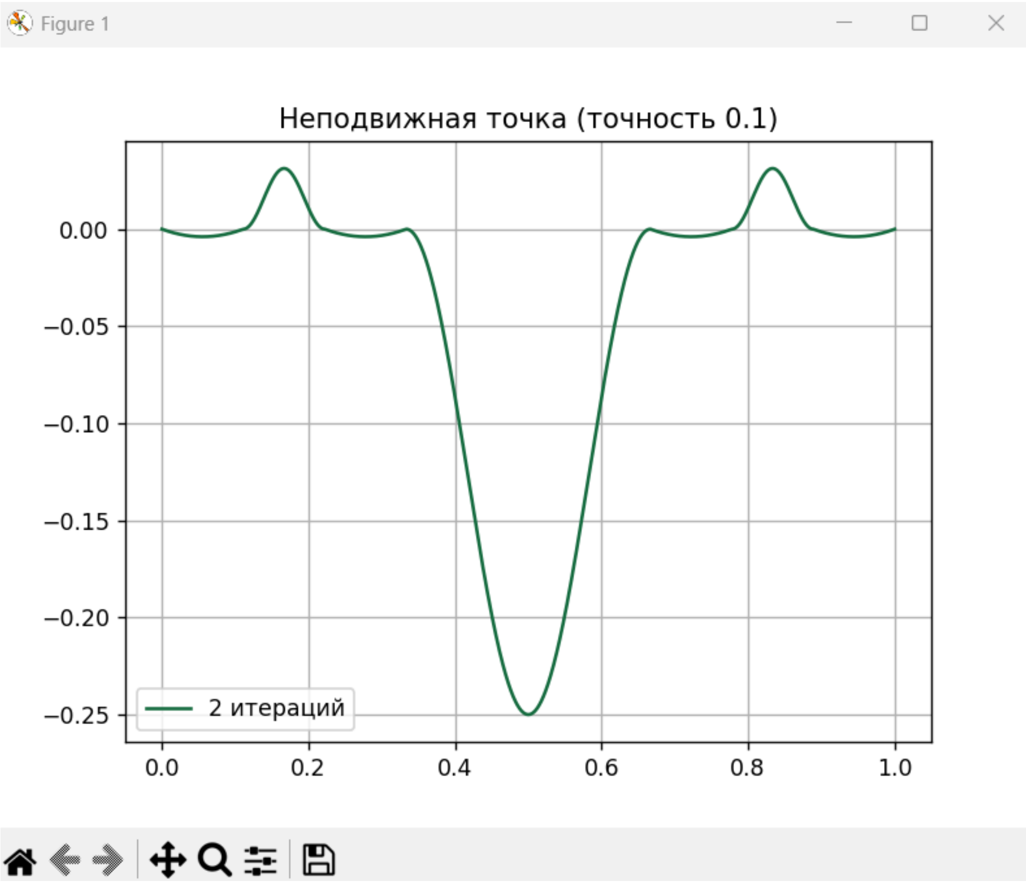
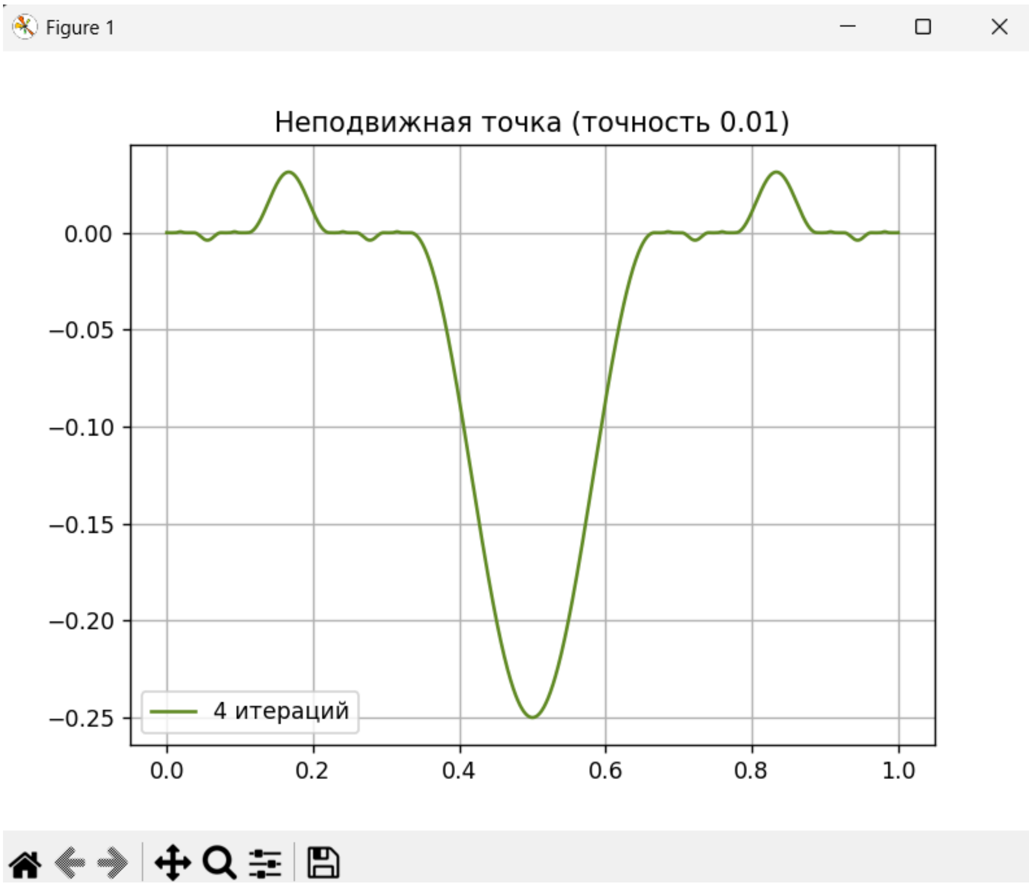


Figure 1
Неподвижная точка (точность 1e-10)
0.00
−0.05
−0.10
−0.15
−0.20
−0.25
0.0
0.2
0.4
0.6
0.8
1.0
21 итераций
22 итераций
23 итераций
24 итераций

Начальное приближение $x0(t) = t^2 - t$:





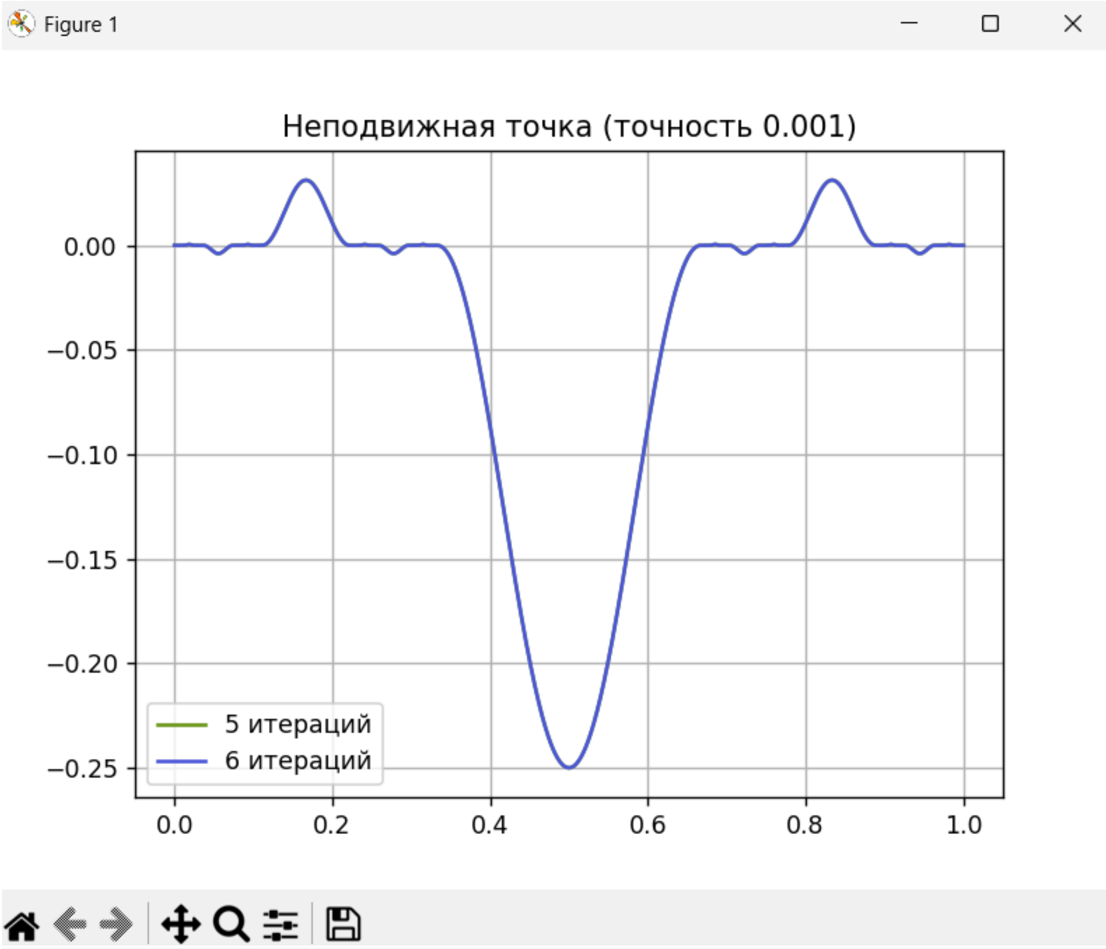


Figure 1
Неподвижная точка (точность 0.001)
0.00
−0.05
−0.10
−0.15
−0.20
−0.25
0.0
0.2
0.4
0.6
0.8
1.0
5 итераций
6 итераций

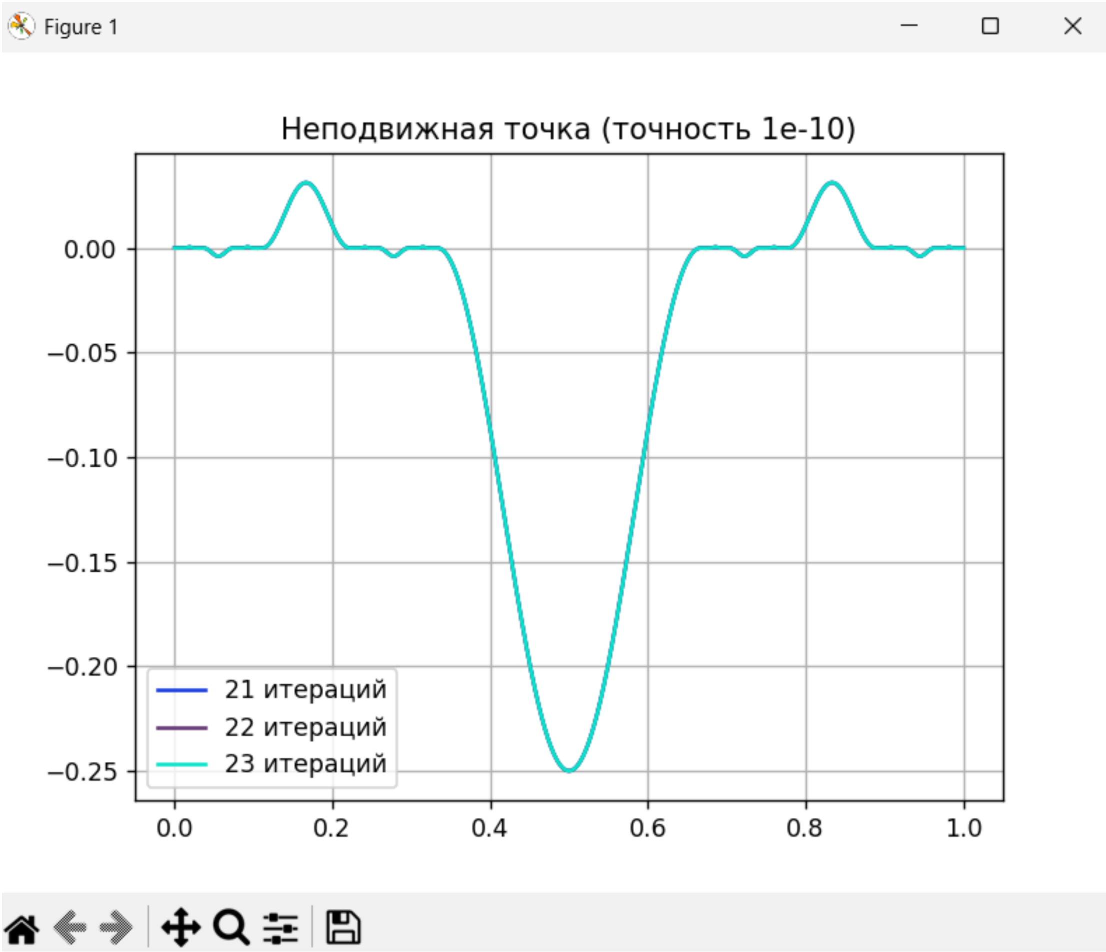


Figure 1
Неподвижная точка (точность 1e-10)
0.00
−0.05
−0.10
−0.15
−0.20
−0.25
0.0
0.2
0.4
0.6
0.8
1.0
21 итераций
22 итераций
23 итераций