**Лабораторна робота 2**

**Тема: «Чисельні методи розв’язання нелінійних рівнянь.**

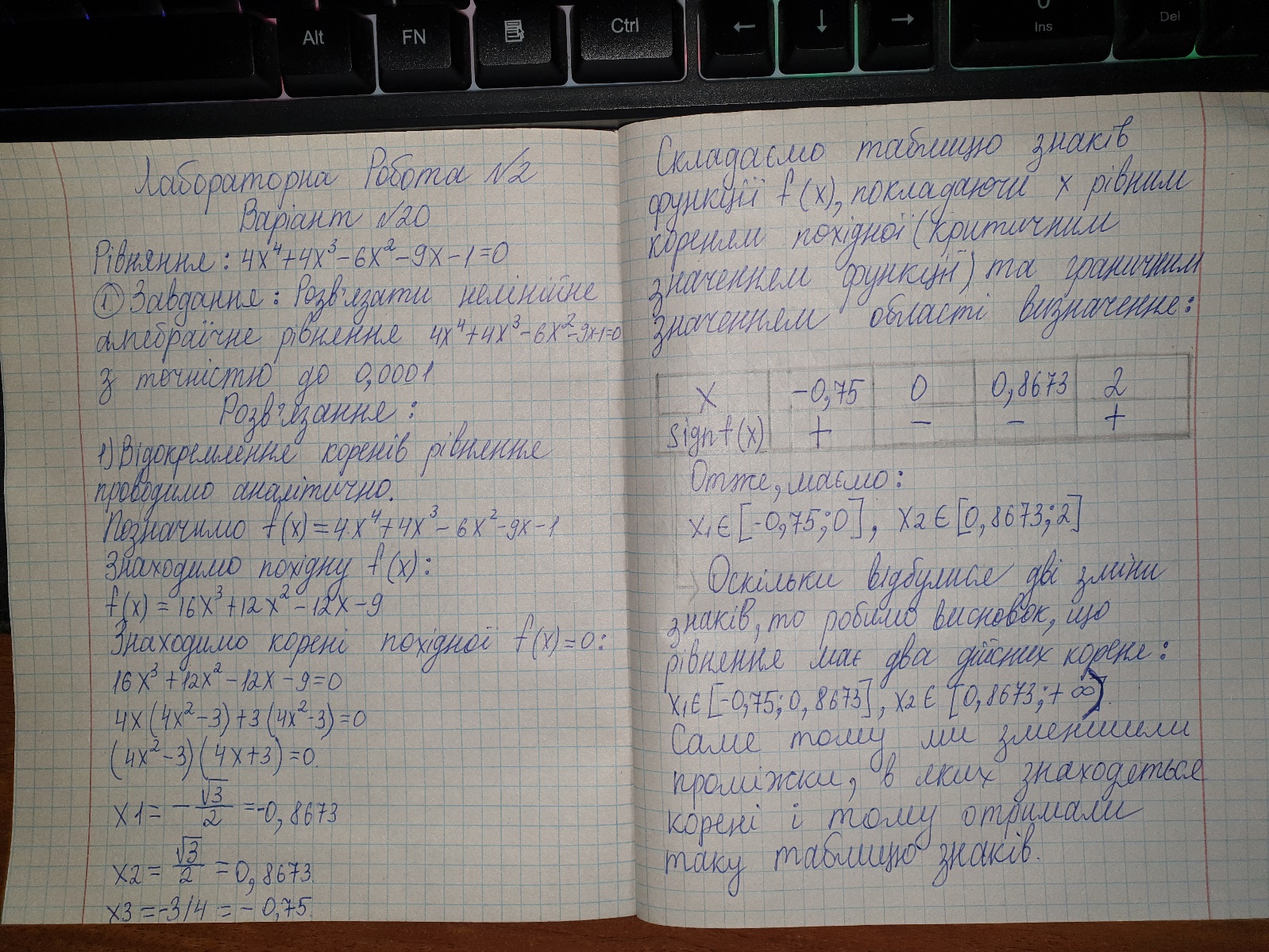
**Завдання:** Завдання: Розв’язати нелінійне алгебраїчне рівняння f(x)=0 з точністю до 0,0001. Відокремлення коренів виконати аналітично. Уточнення коренів провести методом половинного ділення та методом хорд.

**Варіант №20**

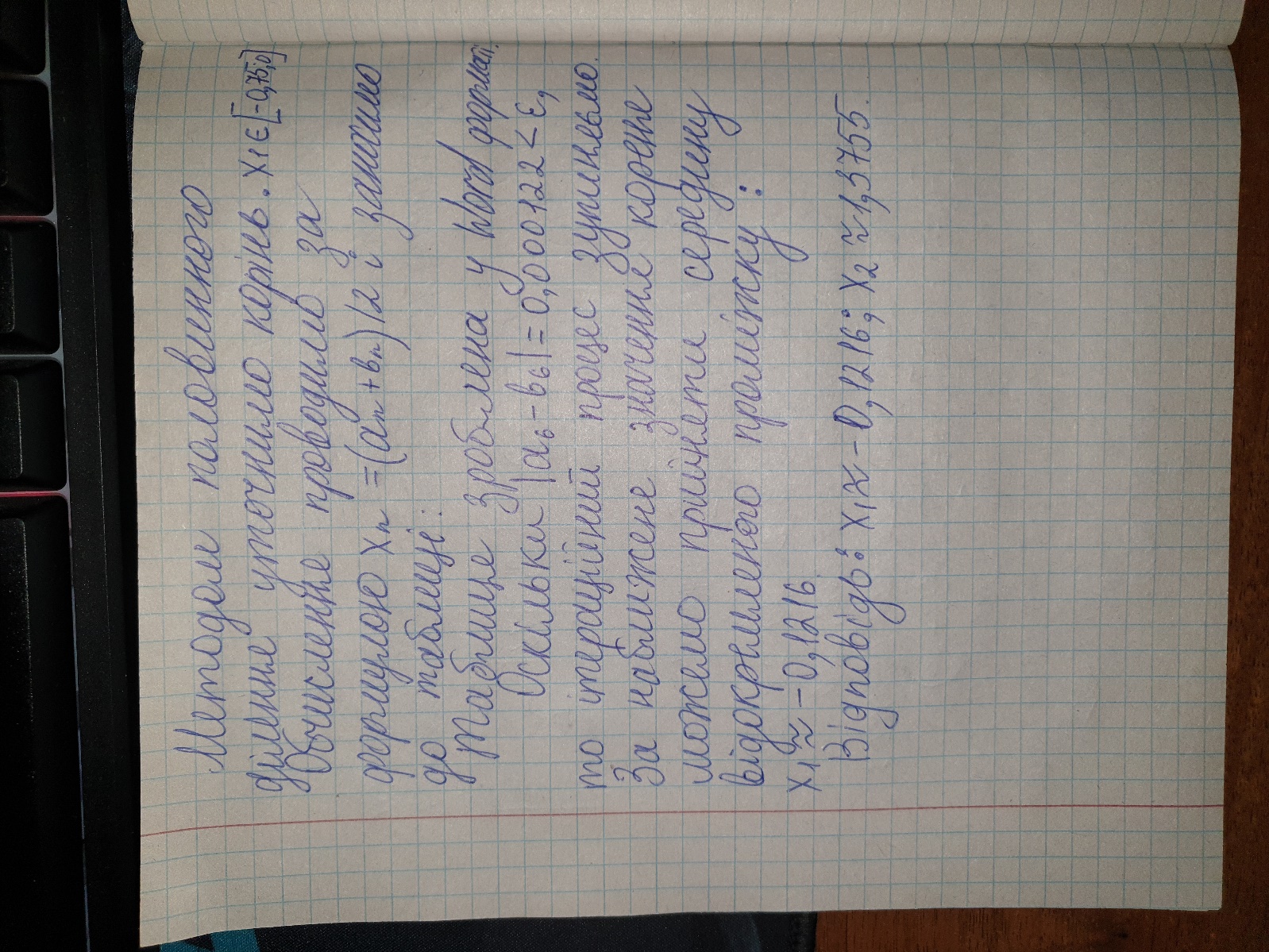
**Рівняння:** ****

**Виконання:**

1. **Розв’язання нелінійного алгебраїчного рівняння f(x)=0 з точністю до 0,0001. Виконання аналітичного відокремлення коренів.**

****

1. **Проведення уточнення коренів методом половинного ділення та методом хорд.**



Методом половинного ділення уточнимо корінь . 

Обчислення проводимо за формулою  і заносимо до таблиці:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
| 0 | -0.75 | 0 | -0.375 | 1.3994 | 0.75 |
| 1 | -0.375 | 0 | -0.1875 | 0.4551 | 0.375 |
| 2 | -0.1875 | 0 | -0.09375 | -0.212 | 0.1875 |
| 3 | -0.09375 | -0.0625 | -0.0781 | -0.3354 | 0.03125 |
| 4 | -0.1094 | -0.09375 | -0.1094 | -0.09206 | 0.01563 |
| 5 | -0.1172 | -0.1094 | -0.1172 | -0.03339 | 0.00781 |
| 6 | -0.1211 | -0.1172 | -0.1211 | -0.00438 | 0.00391 |
| 7 | -0.123 | -0.1211 | -0.123 | 0.01004 | 0.00195 |
| 8 | -0.1221 | -0.1211 | -0.1221 | 0.00284 | 0.000977 |
| 9 | -0.1216 | -0.1211 | -0.1216 | -0.00077 | 0.000488 |
| 10 | -0.1218 | -0.1216 | -0.1218 | 0.00103 | 0.000244 |
| 11 | -0.1217 | -0.1216 | -0.1217 | 0.000132 | 0.000122 |

Оскільки , то ітераційний процес зупиняємо. За наближене значення кореня можемо прийняти середину відокремленого проміжку:

*Відповідь*: , .

Алгоритм методу половинного ділення

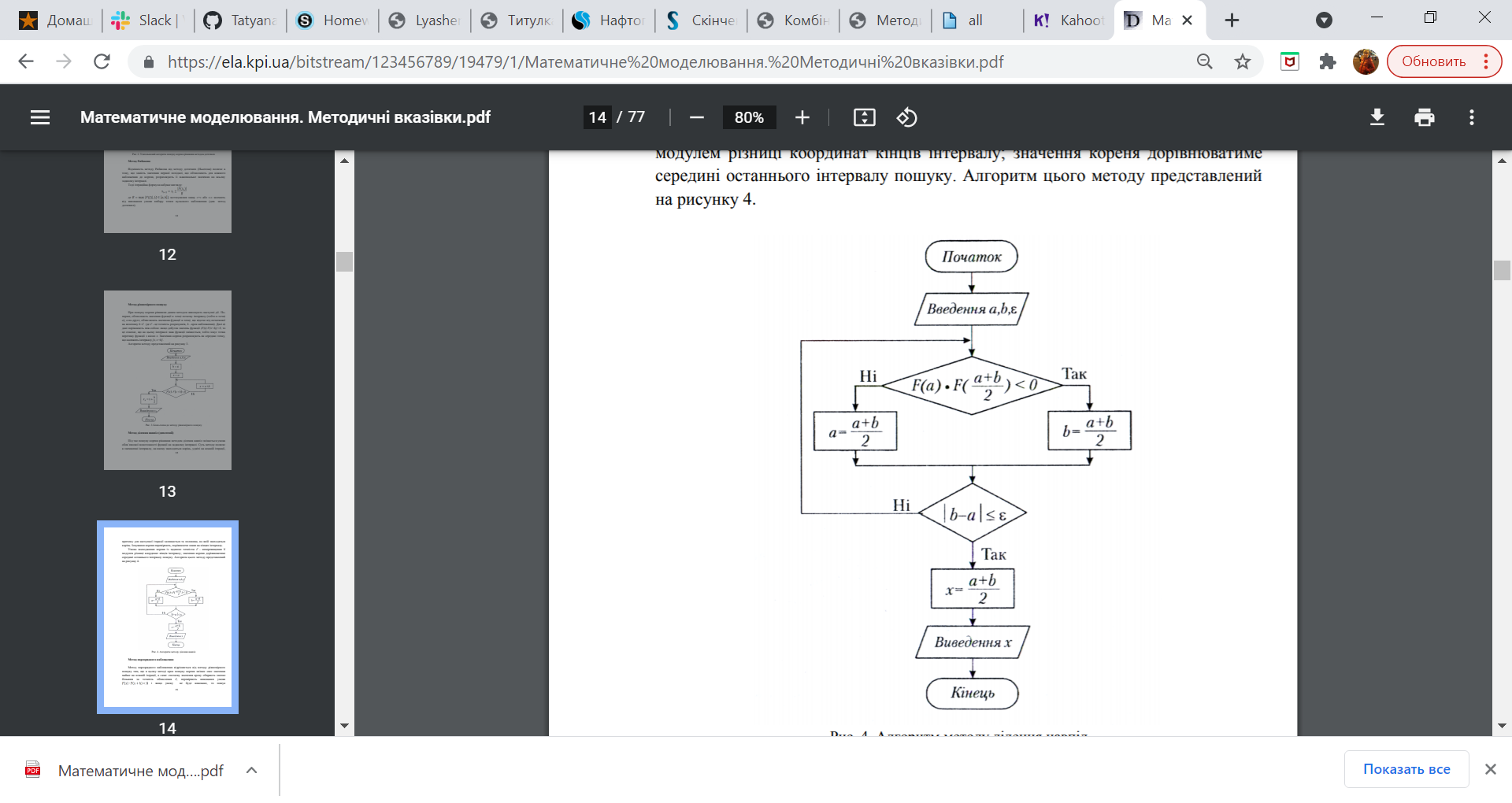


Рис. 1 Алгоритм методу половинного ділення

**Код**

import numpy as np

import math

import matplotlib.pyplot as plt

def f(x):

# 4\*x^4+4\*x^3-6\*x^2-9\*x-1=0

return 4\*x\*\*4+4\*x\*\*3-6\*x\*\*2-9\*x-1

a = -0.75

b = 0

eps = 0.0001 # точність

def rec\_dyhotomy(a, b, eps):

if abs(f(b) - f(a)) < eps:

print('Обчислюємо корінь')

return

mid = (a + b) / 2

if f(mid) == 0 or abs(f(mid)) < eps:

print(f'Корінь знаходиться в точці x = {mid}')

elif f(a) \* f(mid) < 0:

rec\_dyhotomy(a, mid, eps)

else:

rec\_dyhotomy(mid, b, eps)

rec\_dyhotomy(a, b, eps)

x = np.arange(a, b, 0.01)

plt.plot(x, f(x))

plt.xlabel('x')

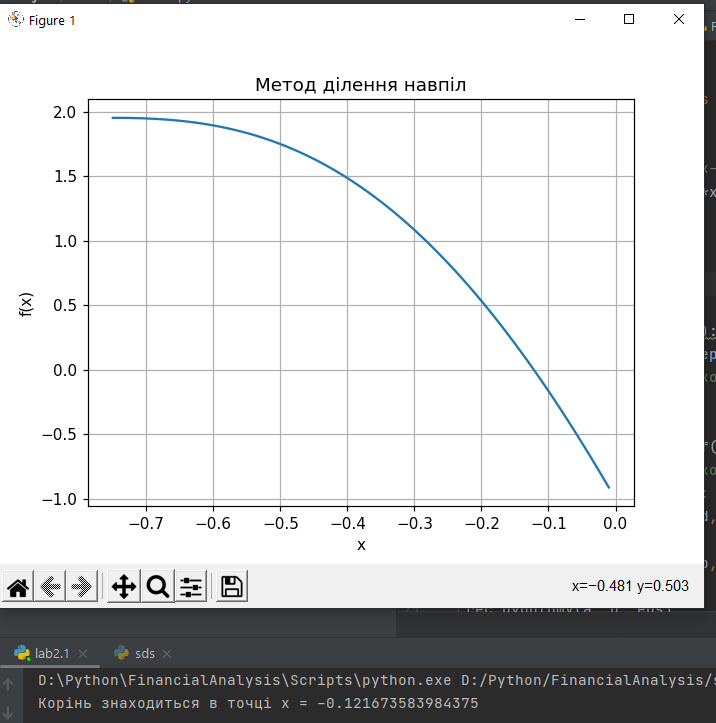
plt.ylabel('f(x)')

plt.title('Метод ділення навпіл')

plt.grid()

plt.show()

**Скрін**



Алгоритм методу хорд

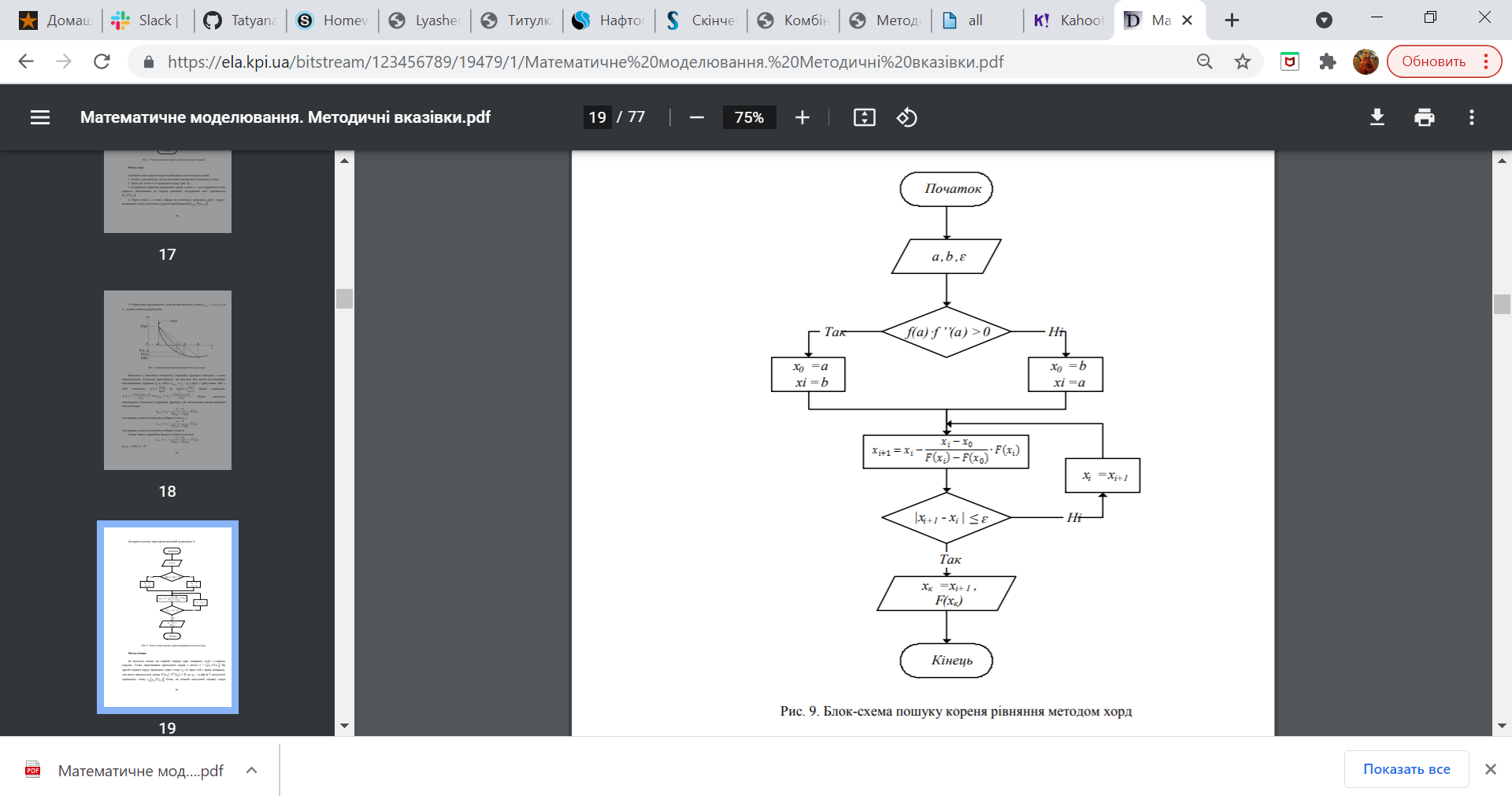


Рис. 2. Алгоритм методу хорд

**Код**

import numpy as np

import math

from scipy.misc import derivative

def f(x):

# 4\*x^4+4\*x^3-6\*x^2-9\*x-1=0

return 4\*x\*\*4+4\*x\*\*3-6\*x\*\*2-9\*x-1

a = 0

b = 2

eps = 0.0001 # точність

def hord(a, b, eps):

if abs(f(b) - f(a)) < eps:

print('немає кореня')

return

if (f(a) \* derivative(f, a, n=2)):

x0 = a

xi = b

else:

x0 = b

xi = a

xi\_1 = xi - (xi - x0) \* f(xi) / (f(xi) - f(x0))

while (abs(f(xi\_1) - f(xi)) > eps):

xi = xi\_1

xi\_1 = xi - (xi - x0) \* f(xi) / (f(xi) - f(x0))

else:

print(f'Корінь знаходиться в точці x =', xi\_1)

hord(a, b, eps)

