import math

def F(x):

return 3 \* x \*\* 4 - x \*\* 3 + 10 \* x \*\* 2 - 5 \* x - 3 # f(x)

def D1(x):

return 12 \* x \*\* 3 - 3 \* x \*\* 2 + 20 \* x - 5 # f'(x)

def D2(x):

return 36 \* x \*\* 2 - 6 \* x + 20 # f''(x)

def metKomb(a, b):

if D1(a) \* D2(a) < 0:

a0 = b

b0 = a

else:

a0 = a

b0 = b

xp1 = (a0 - F(a0)) / D1(a0) #a1

xp2 = (b0 - F(b0) \* (b0 - a0)) / (F(b0) - F(a0)) #b1

while xp2 - xp1 < 0.0001:

xn1 = (xp1 - F(xp1) \* (xp2 - xp1)) / (F(xp2) - F(xp1))

xn2 = xp2 - F(xp2 / D1(xp2))

xp1 = xn1

xp2 = xn2

x = (xp1 + xp2) / 2

return print(' Metod: kombinovanyi\n x = ', x)

metKomb(-2, 6)