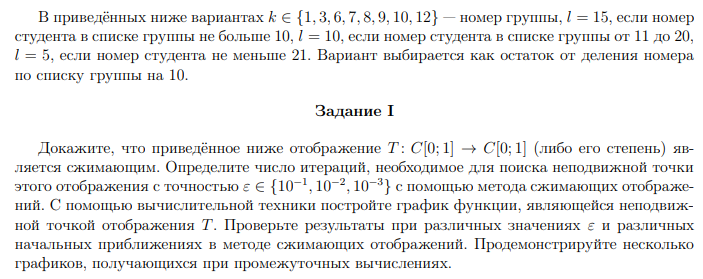
**РГР 1**

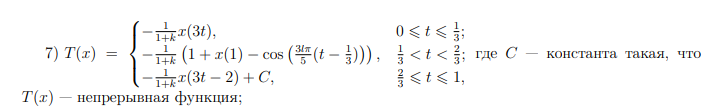
**КУЦЕНКО М.Д.**

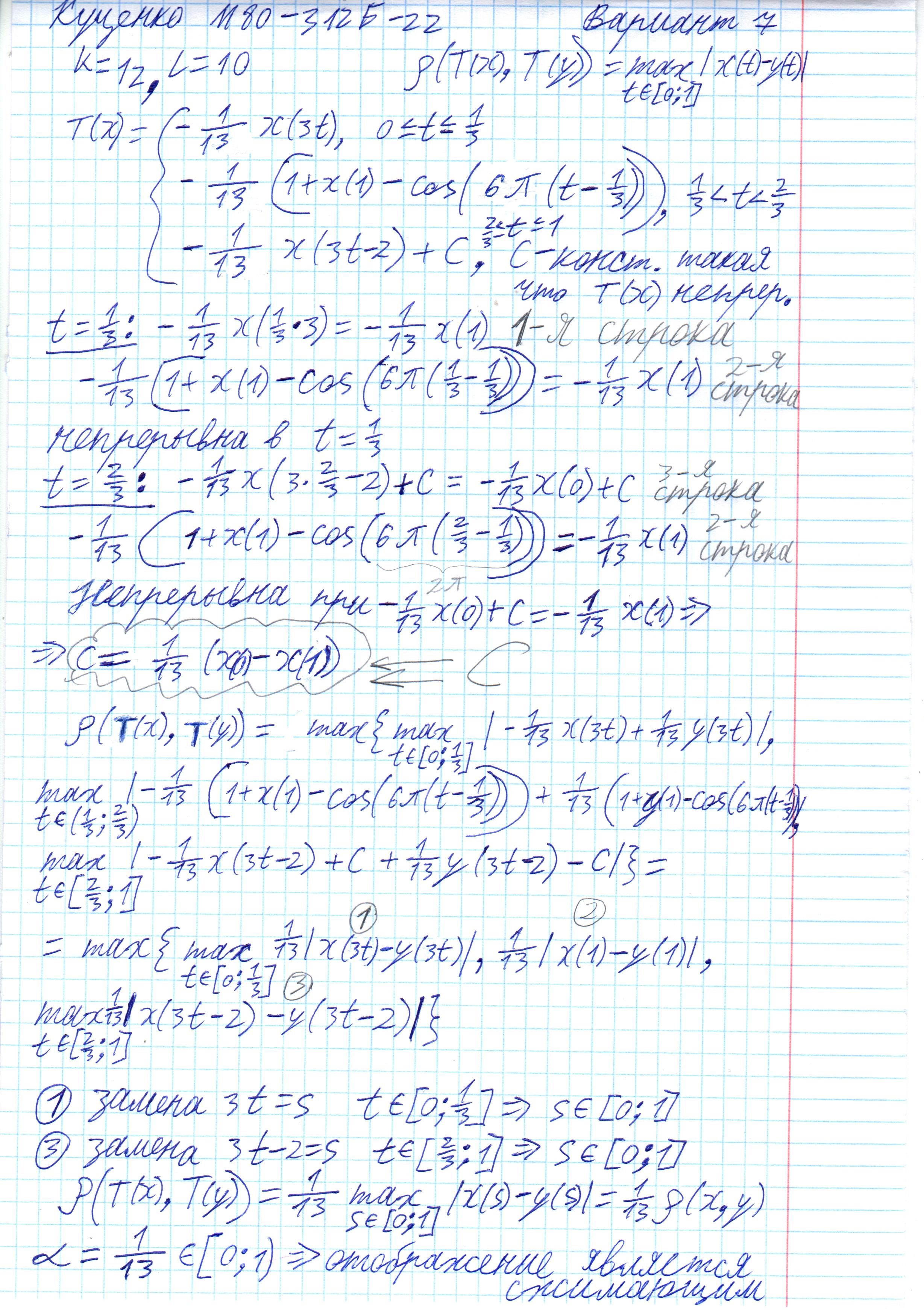
**М8О-312Б-22**

Задание:



Вариант 7:





Код:

|  |
| --- |
| import numpy as np from matplotlib import pyplot as plt  # Точность поиска precision = 1e-4   # Разбиение [0;1] на точки для вычислений split = 600 x = np.linspace(0, 1, split) array = x.copy()  # Функция получения индекса массива из значения t def get\_index(val) -> int:   return int(val\*(split-1))  # Функция сжатия def compression(cur\_array) -> list:   new\_array = np.zeros\_like(cur\_array) # Создание нового массива для заполнения   for t in x: # Проведение вычислений по каждому t из разбиения [0;1]       if t >= 0 and t <= 1/3:           new\_array[get\_index(t)] = -1/13 \* cur\_array[get\_index(3 \* t)]       if t > 1/3 and t < 2/3:           new\_array[get\_index(t)] = -1/13 \* (1 + cur\_array[-1] - np.cos(6 \* np.pi \* (t - 1/3)))       if t >= 2/3 and t <= 1: # В конце           new\_array[get\_index(t)] = -1/13 \* cur\_array[get\_index(3 \* t - 2)] + 1/13 \* (cur\_array[0] - cur\_array[-1])   return new\_array  # Функция расстояния в ЛП def distance(array1, array2):   return max(abs(array1 - array2))  # Выбор числа итераций для заданной точности num\_iters = int(np.log(precision\*(1-1/13)/distance(array, compression(array)))/np.log(1/13)) + 1  # Учитывание "нулевую" итерацию num\_iters += 1  # Создание области для отрисовки графиков fig, axes = plt.subplots(1, num\_iters, figsize=(5\*num\_iters, 5)) fig.suptitle(f'Графики сжимающего отображения для ε = {precision}', fontsize=16)  # Построение графиков for i in range(num\_iters):   axes[i].plot(x, array)   axes[i].set\_title(f'Итерация {i}')   array = compression(array) # Проведение сжатия  # Отображение графиков plt.tight\_layout() plt.show() |

Результаты работы кода со сравнением числа итераций для разных значение точности:

