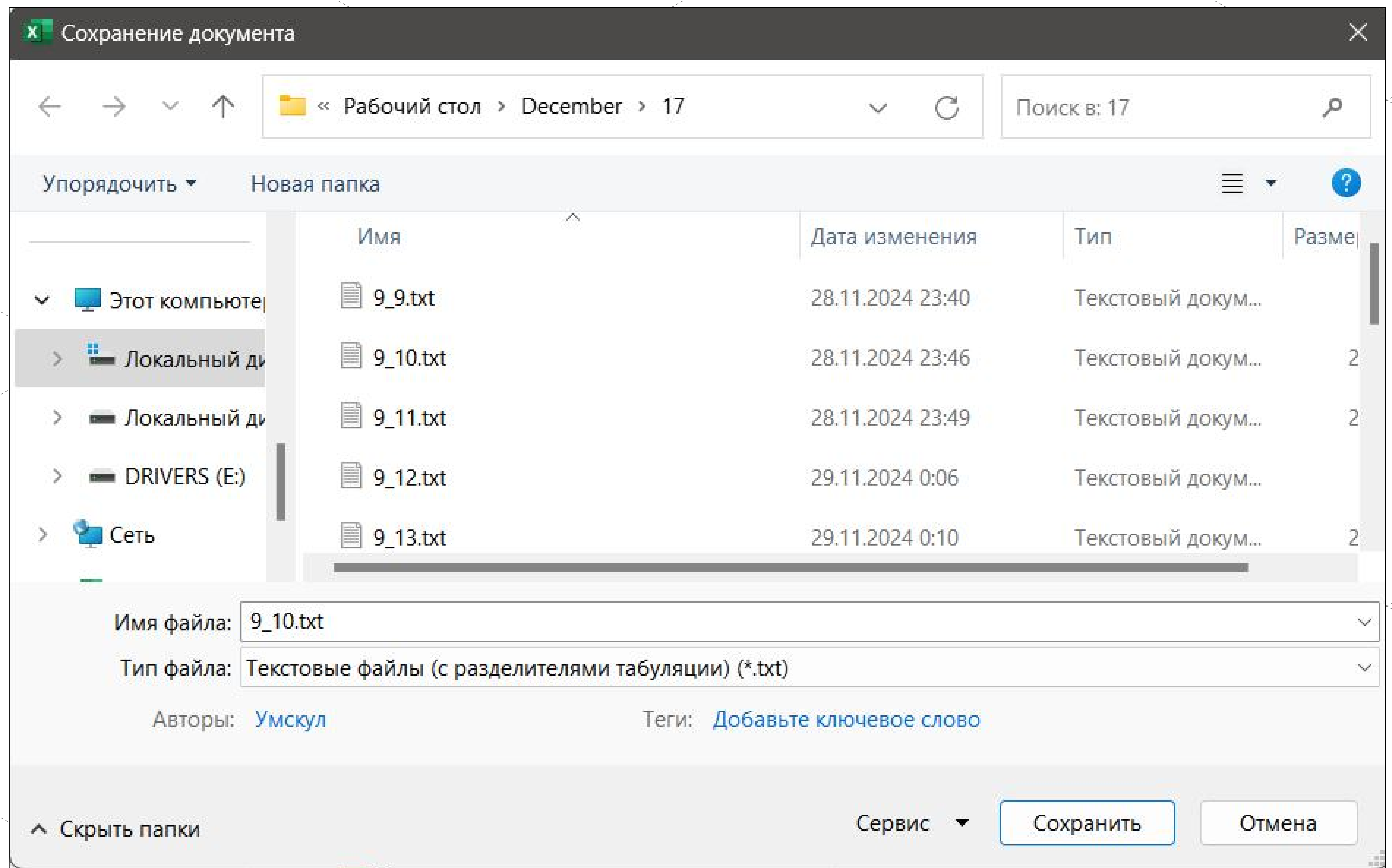


# Программирование и работа с файлами.

## Теория

Чтобы без проблем перенести данные из файла электронной таблицы в текстовый файл с расширением .txt, необходимо сохранить исходный файл вот так:



\*Выбираем именно формат сохранения с разделителями табуляции!

## Считывание данных из файла

Есть два способа считывания чисел из файла: с помощью цикла и с помощью генератора.

Варианты считывания всех чисел из файла:

```
f = open("task9.txt")
a = [int(x) for x in f]
```

```
f = open("task9.txt")
a = []
for s in f:
    a.append(int(s))
```



# Программирование и работа с файлами.

## Теория

### Повторяющиеся числа в строке

Функция `set()` в Python используется для создания множества — неупорядоченной коллекции уникальных элементов. Это позволяет избавиться от повторяющихся значений в данных.

```
numbers = [1, 2, 3, 2, 1, 4, 5, 5]
unique_numbers = set(numbers)
print(unique_numbers)
# Вывод: {1, 2, 3, 4, 5}
```

### Нахождение длины числа

Чтобы найти длину числа, нужно привести число к строке и найти длину числа с помощью функции `len()`.

#### Запомним!

Если в файле есть отрицательные числа, то перед тем, как приводить число к строке, нужно взять модуль у числа, так как у отрицательных чисел минус считается за символ.

### Нахождение чисел, оканчивающихся на n

Есть два способа, чтобы найти необходимые числа.

Варианты считывания всех чисел из файла:

```
max_n = max([x for x in a if abs(x) % 100 == 22])
```

Необходимо брать модуль, так как у отрицательных чисел остатки вычисляются не так, как у натуральных.

```
max_n = max([x for x in a if str(x)[-2:] == "22"])
```





## Пример задания №9

Откройте файл электронной таблицы, содержащей в каждой строке шесть натуральных чисел. Определите количество строк таблицы, содержащих числа, для которых выполнены оба условия:

- в строке только одно число повторяется трижды, остальные числа различны;
- квадрат суммы всех повторяющихся чисел строки больше квадрата суммы всех её неповторяющихся чисел.

В ответе запишите только число.

## Решение задания

- 1 Сохраним таблицу в виде формата текстового файла с разделителями табуляции.
- 2 Пройдемся по каждому строке файла, строку преобразуем в список с целыми числами.
- 3 Списки `povt` и `nepovt` заполним элементами, которые повторяются 3 и 1 раз соответственно.
- 4 Если количество повторяющихся чисел три раза равно 3, а количество уникальных числе в строке равно 3 и квадрат суммы всех повторяющихся чисел строки больше квадрата суммы всех её неповторяющихся чисел, то добавляем список `a` в список `ans`.
- 5 Выводим длину списка `ans` и получаем ответ.

### Полный код Python:

```
f = open("9.txt")
ans = []
for s in f:
    a = list(map(int, s.split()))
    povt = [x for x in a if a.count(x) == 3]
    nepovt = [x for x in a if a.count(x) == 1]
    if len(povt) == 3 and len(nepovt) == 3 and \
        sum(povt) ** 2 > sum(nepovt) ** 2:
        ans.append(a)
print(len(ans))
```



# Заметки

