**ПРАКТИЧЕСКАЯ РАБОТА № 3**

**«Оценка сложности рекурсивных алгоритмов»**

**Порядок работы**

1. Запустили проект на выполнение, убедились в правильности работы программы:

global a

def fib(n: int) -> int:

    global a

    a += 1

    if n == 1 or n == 2:

        return 1

    return fib(n-1) + fib(n-2)

def fact(n: int) -> int:

    global a

    a += 1

    if n == 1:

        return 1

    return n \* fact(n - 1)

def main():

    for i in range(1, 20):

        global a

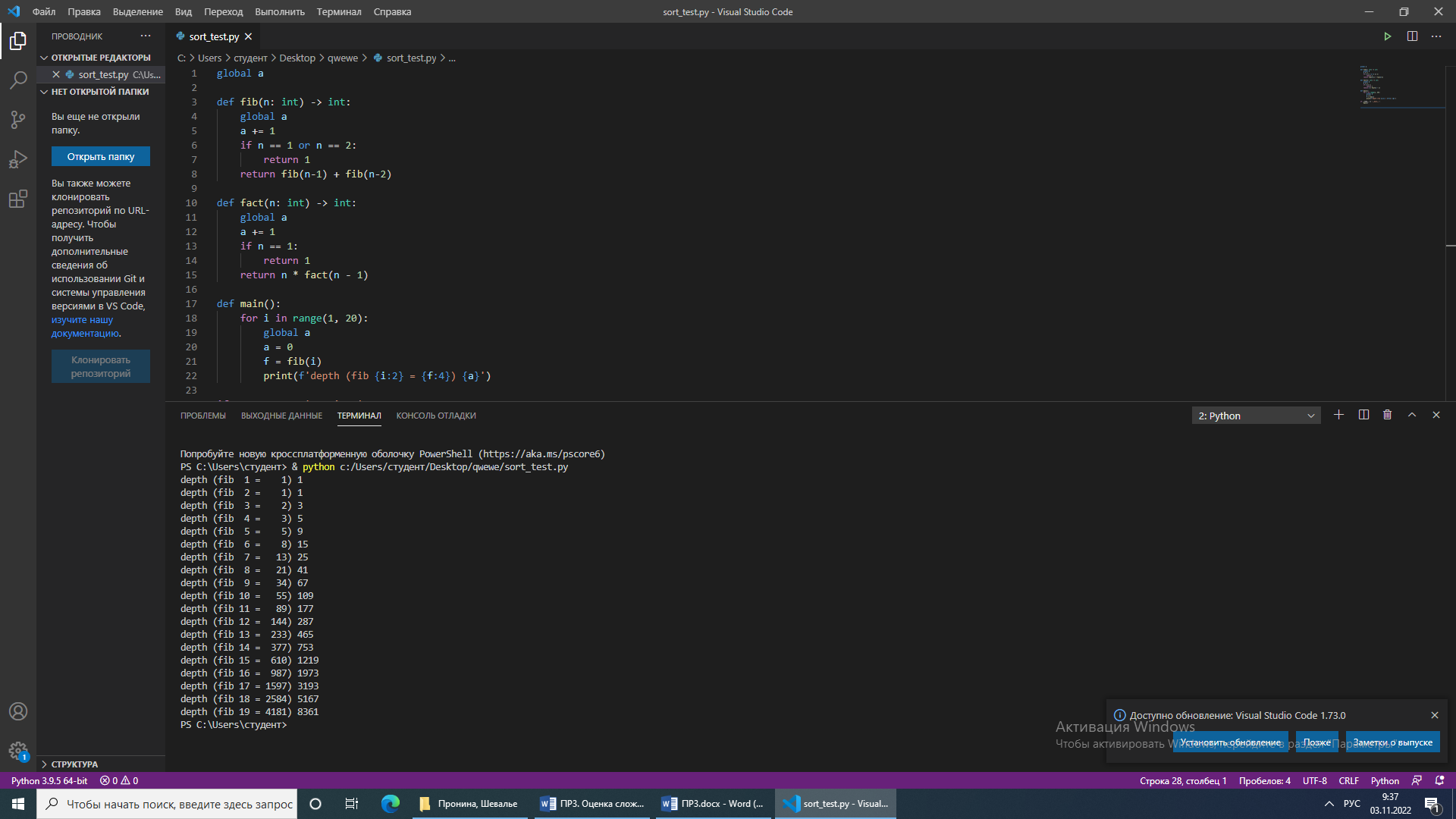
        a = 0

        f = fib(i)

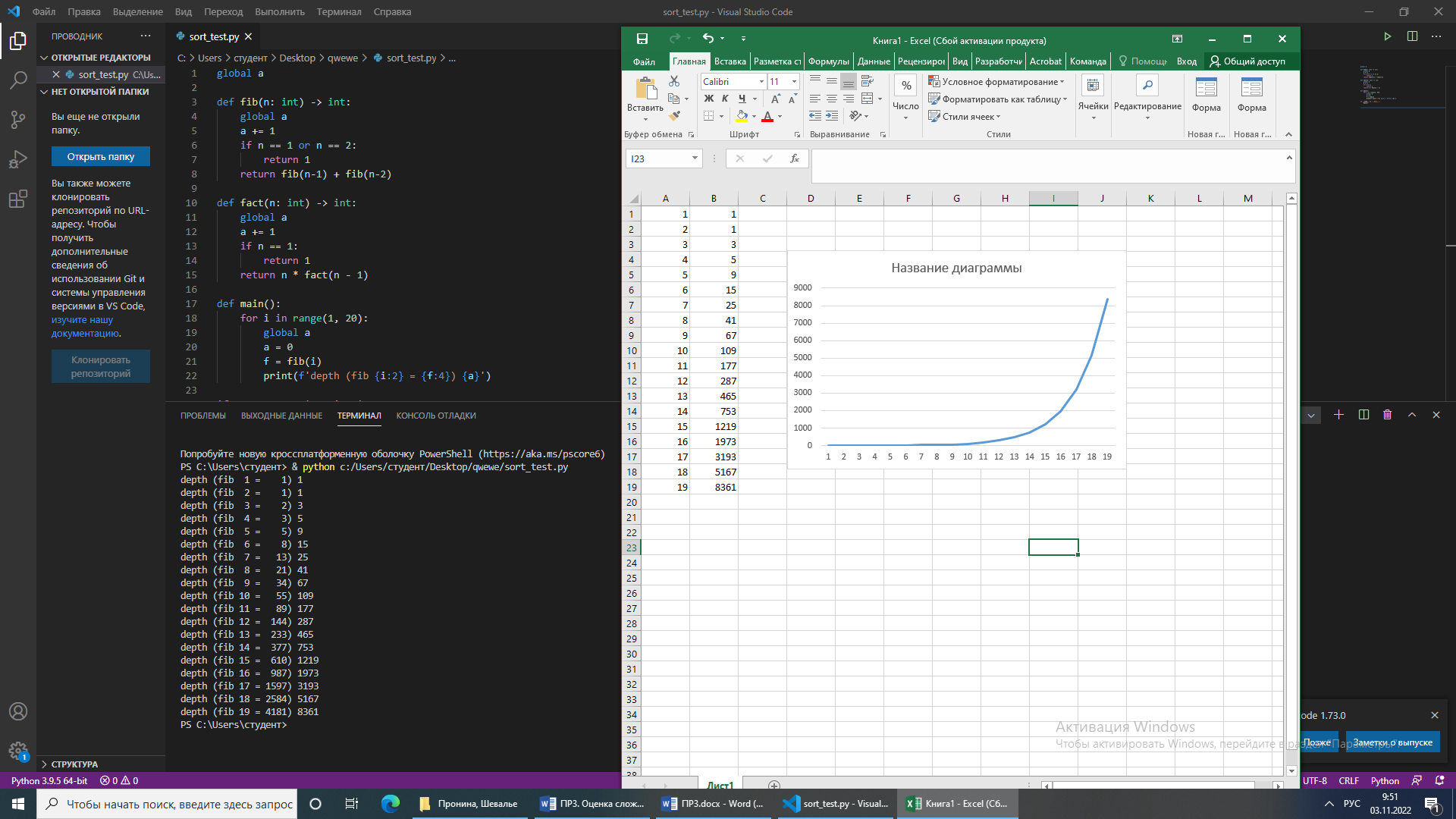
        print(f'depth (fib {i:2} = {f:4}) {a}')

if \_\_name\_\_ == '\_\_main\_\_':

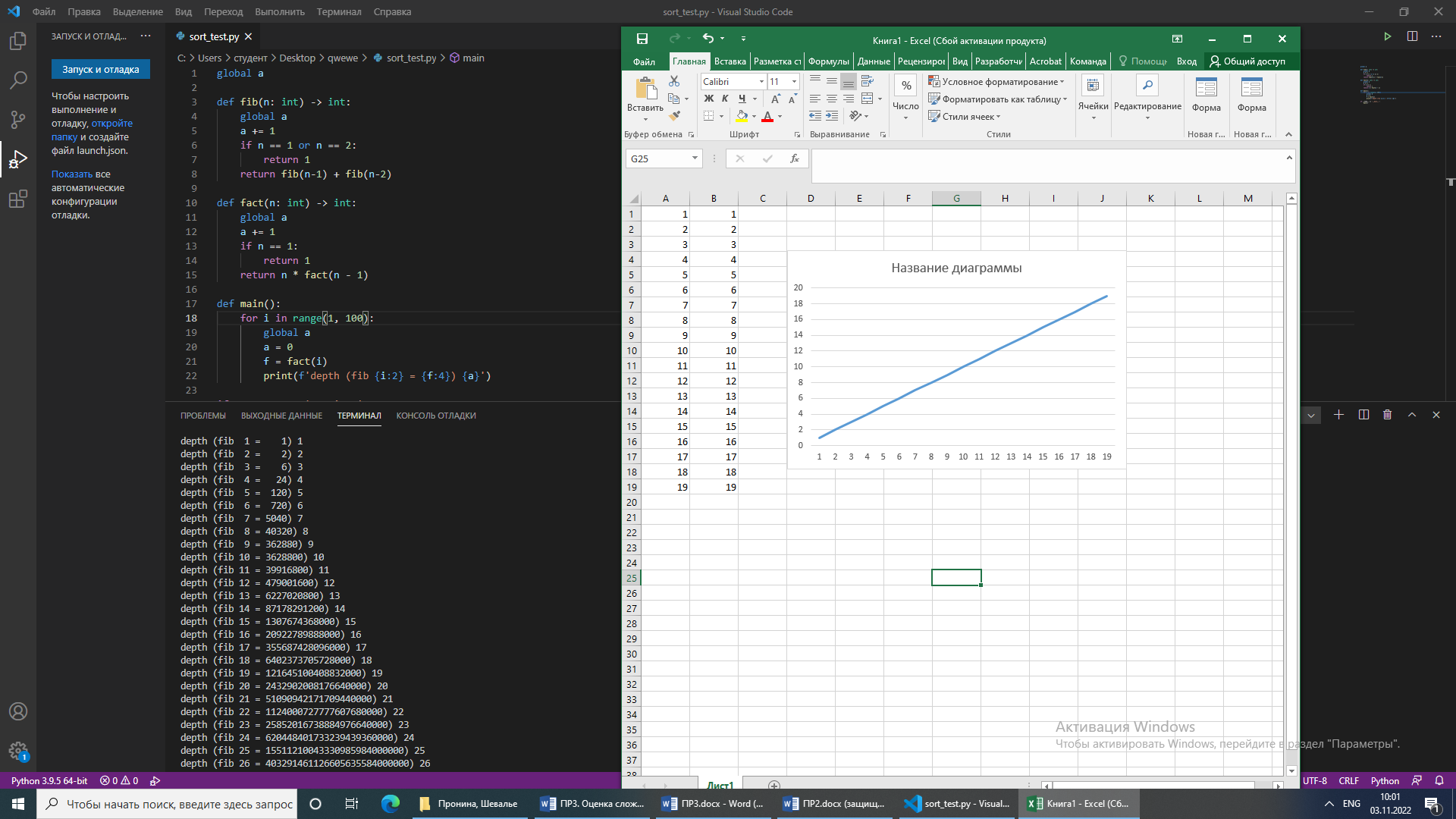
    main()



**2.** Записали результаты выполнения в таблицу Excel и построили график зависимости времени выполнения от кол-ва массива:



**3.** Изменили алгоритм поиска, внесли изменения в программу и записали результаты выполнения в таблицу Excel, а также построили график:



**Вывод:** основываясь на первом алгоритме, мы смогли определить, что зависимость является не прямой, а квадратичной и что линия графика возрастает не прямолинейно, а по экспоненте и поэтому действие алгоритма выполняется медленнее, нежели второй алгоритм график которого возрастает прямолинейно.