**Лабораторная работа №4**

**Полный перебор**

Решить методом полного p-ичного перебора следующие задачи:

16. Суммы

В текстовый файл записана последовательность целых чисел, состоящая не более чем из 20 элементов. Hайти наименьшее натуpальное число, котоpое нельзя получить суммой никаких элементов массива. Сумма может содеpжать каждый элемент последовательности не более одного pаза.

**Код программы. (Python)**

def plus(a, i):

while a[i] == 1:

a[i] = 0

i -= 1

a[i] = 1

f = open('input.txt', 'r')

a = []

while True:

now = f.readline()

if now == "" :

break

now = int(now)

a.append(now)

f.close()

print(a)

comb = [0] \* len(a)

comb\_sum = []

for i in range(2 \* \*len(a)) :

sum1 = 0

for j in range(len(a)) :

if comb[j] == 1 :

sum1 += a[j]

comb\_sum.append(sum1)

plus(comb, len(a) - 1)

i = 0

while (True):

i += 1

if (not(i in comb\_sum)):

print('Minimum Natural Number = ',i)

Break

**Алгоритм программы**

Функция plus: прибавление 1 к текущему двоичному числу.

Открываем файл “input.txt” для считывания и добавление элементов в массив a. Выводим массив a.

Создаётся массив комбинаций comb, состоящий из нулей длинной массива a. Также создаётся массив comb\_sum, для хранения всевозможных сум элементов массива a.

Цикл for I in range(2\*\*len(a)) - этот цикл, который проходит по всем комбинациям. Объявления переменной sum = 0.

Цикл for j in range(len(a)) - этот цикл проходит по каждому элементу массиву из комбинации. Если выбранный элемент массива комбинаций равен 1, тогда к сумме прибавляем значение из массива a под номер элемента из массива комбинаций, который равен 1.

Пройдя все элементы выбранной комбинации мы в массив comb\_sumb добавляем сумму комбинации и вызываем функцию plus для увеличения текущего двоичного числа (комбинации) на единицу.

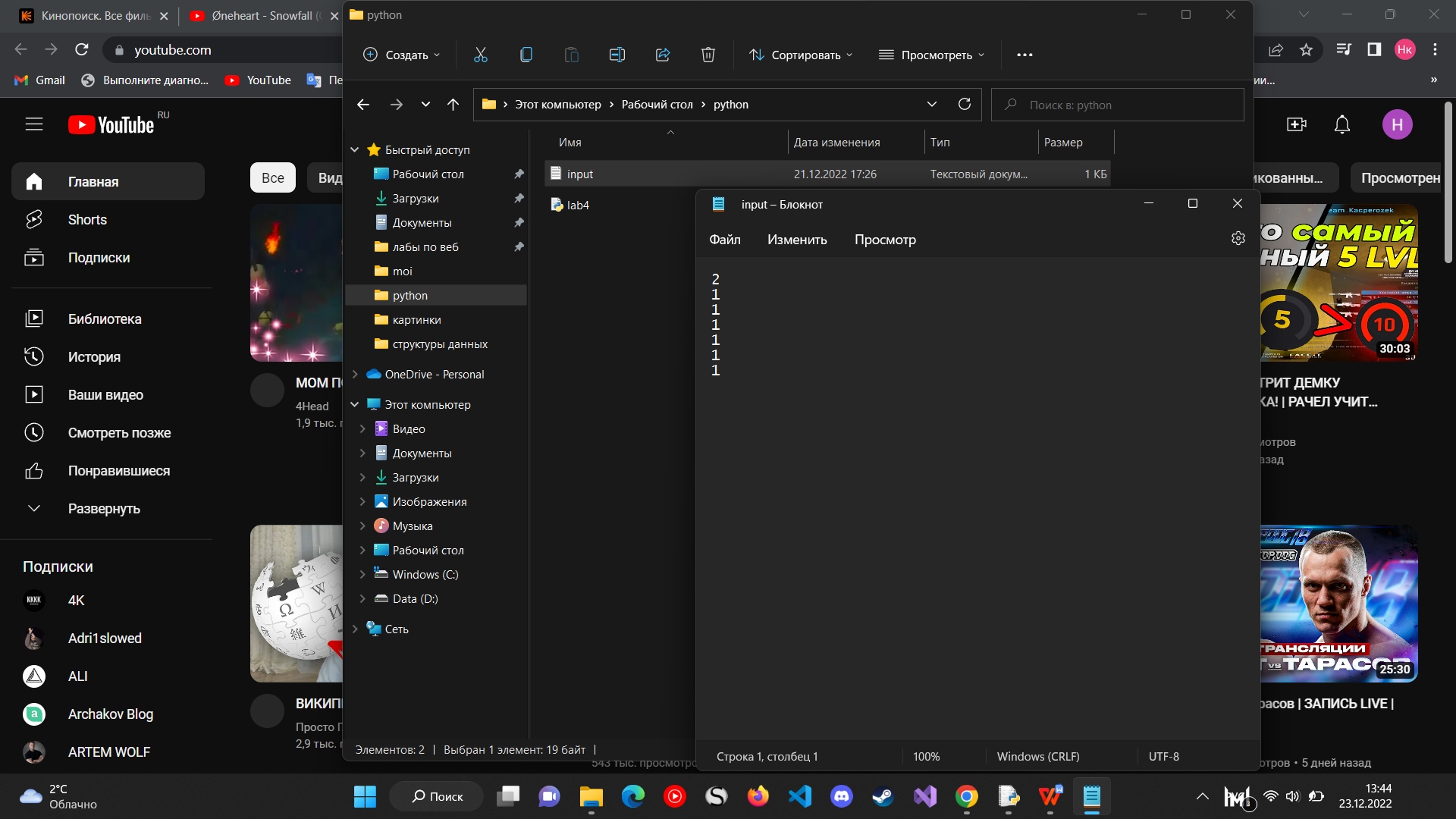
Объявление I = 0 - минимальное натуральное число, которое нельзя получить суммой элементов массива a.

Создаём цикл while, который всегда срабатывает. Увеличиваем I на единицу. Если i нет в массиве comb\_sum, то выводим значение I и тогда выходим из цикла с помощью break.

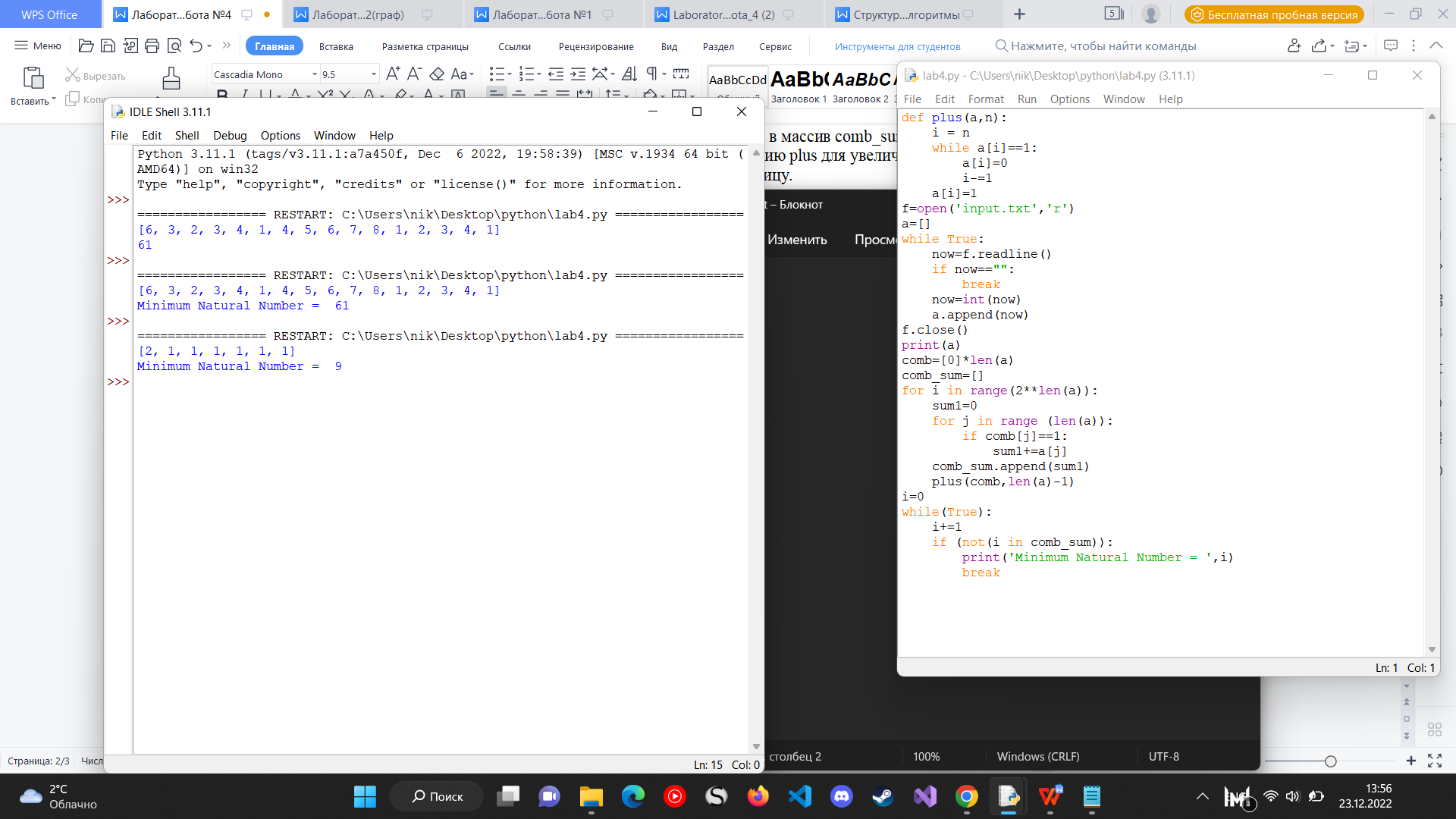
**Тесты.**

Тест 1.

Файл input.txt

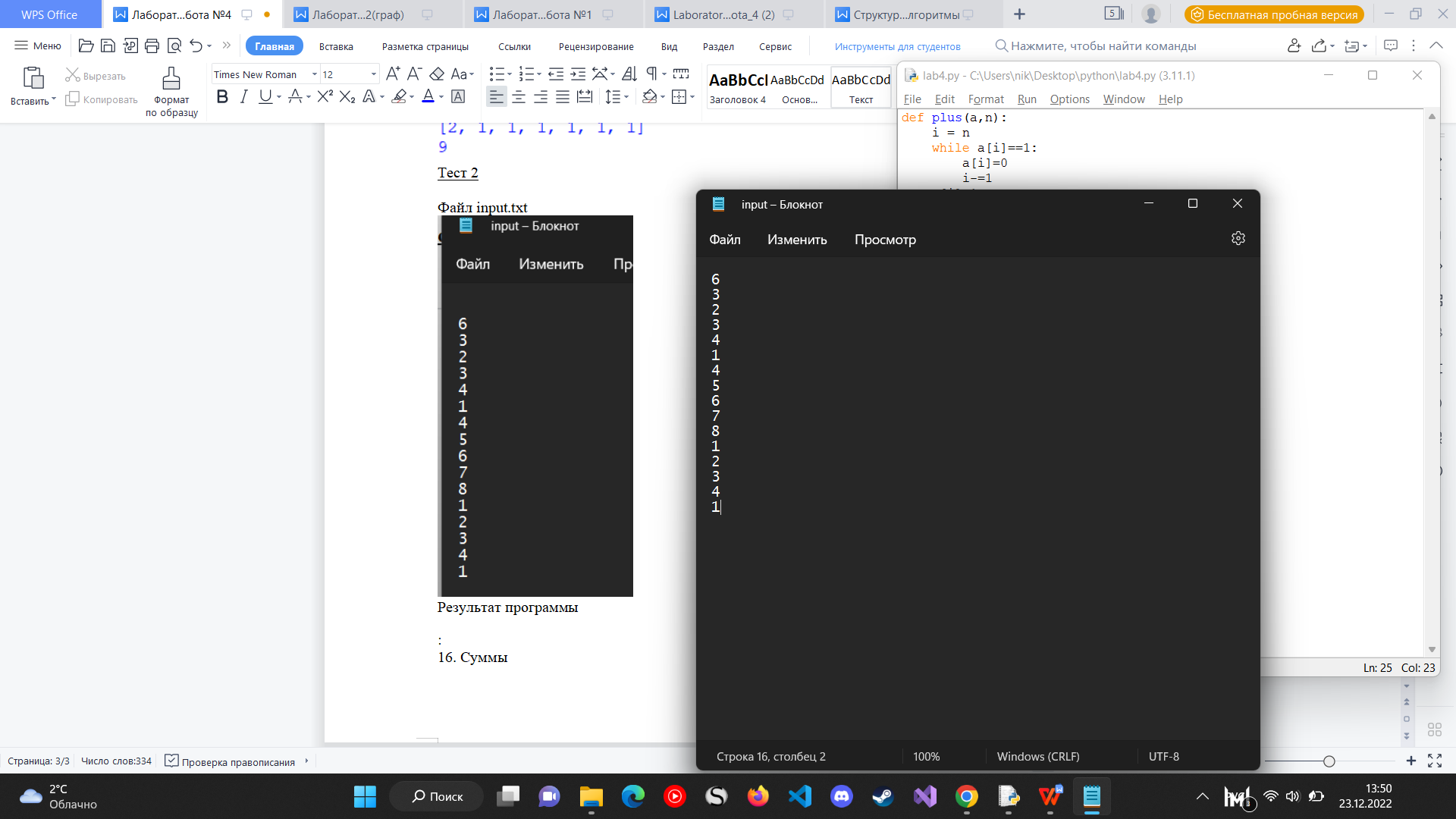


Результат программы



Тест 2

Файл input.txt



Результат программы

