

Отчет по лабораторной работе №0  
по курсу «Алгоритмы и структуры  
данных»

Тема: Введение

Выполнил:

Гашимов И.Ф.

К3139

Проверил:

Афанасьев. А.В.

Санкт-Петербург

2024 г.

# Содержание

<b>Содержание.....</b>	<b>2</b>
<b>Задание 1.....</b>	<b>3</b>
1.1.....	3
1.2.....	3
1.3.....	4
1.4.....	5
<b>Задание 2.....</b>	<b>5</b>
<b>Задание 3.....</b>	<b>6</b>
<b>Задание 4.....</b>	<b>7</b>

# Задание 1

## 1.1

1. Задача  $a + b$ . В данной задаче требуется вычислить сумму двух заданных чисел. Вход: одна строка, которая содержит два целых числа  $a$  и  $b$ . Для этих чисел выполняются условия  $-10^9 \leq a, b \leq 10^9$ . Выход: единственное целое число — результат сложения  $a + b$ .

```
a, b = map(int, input().split())
if a > 10**9 or a < -10**9 or b > 10**9 or b < -10**9:
    print("число должно быть в диапазоне от -10^9 до 10^9")
else:
    print(a + b)
```

- input сделал int потому-что знаки  $<$   $>$   $<=$   $>=$  работают с int'ами
- если число не в промежутке  $-10^9$  и  $10^9$ , то print выдаст “число должно бы”, иначе сумма  $a$  и  $b$

## 1.2

2. Задача  $a + b^2$ . В данной задаче требуется вычислить значение  $a + b^2$ . Вход: одна строка, которая содержит два целых числа  $a$  и  $b$ . Для этих чисел выполняются условия  $-10^9 \leq a, b \leq 10^9$ . Выход: единственное целое число — результат сложения  $a + b^2$ .

```
a, b = map(int, input().split())
if a > 10**9 or a < -10**9 or b > 10**9 or b < -10**9:
    print("число должно быть в диапазоне от -10^9 до 10^9")
else:
    print(a+b**2)
```

- здесь уже используем  $**$  чтобы возвести в степень, так же степень в приоритете выше, чем  $+$ , поэтому код работает правильно

## 1.3

3. Выполните задачу  $a + b$  с использованием файлов.

- Имя входного файла: input.txt
- Имя выходного файла: output.txt
- Формат входного файла. Входной файл состоит из одной строки, которая содержит два целых числа  $a$  и  $b$ . Для этих чисел выполняются условия  $-10^9 \leq a, b \leq 10^9$ .
- Формат выходного файла. Выходной файл единственное целое число — результат сложения  $a + b$ .

Примеры.

input.txt	12 25	130 61
output.txt	37	191

```
f = open('input.txt', 'r')
a = f.read()
a = a.split(" ")
a[0], a[1] = int(a[0]), int(a[1])
f.close()
if a[0] > 10 ** 9 or a[0] < -10 ** 9 or a[1] > 10 ** 9 or a[1] <
    -10 ** 9:
    print("число должно быть в диапазоне от -10^9 до 10^9")
else:
    a = a[0] + a[1]
    a = str(a)
    z = open('output.txt', 'w')
    z.write(a)
    z.close()
```

- с помощью переменной открываем файл input.txt, можно было обойтись без аргумента 'r', потому-что по default'у она в режиме чтения

- после с помощью `split` мы разделяем их используя сеператор “(пробел)“, после чего она становится списком
- по стандарту файлы читаются и записывают формат `str`, поэтому в начале работы переводим их в `int`, а в конце когда хотим записать, то в `str`

## 1.4

4. Выполните задачу  $a+b^2$  с использованием файлов аналогично предыдущему пункту.

```
f = open('input.txt', 'r')
a = f.read()
a = a.split(" ")
a[0],a[1]=int(a[0]),int(a[1])
f.close()
if a[0] > 10 ** 9 or a[0] < -10 ** 9 or a[1] > 10 ** 9 or a[1] <
    -10 ** 9:
    print("число должно быть в диапазоне от -10^9 до 10^9")
else:
    a = a[0] + a[1]**2
    a = str(a)
    z = open('output.txt','w')
    z.write(a)
    z.close()
```

- здесь то же самое, что и в предыдущем, только уже со степенью

## Задание 2

Ваша цель – разработать эффективный алгоритм для подсчета чисел Фибоначчи. Вам предлагается начальный код на Python, который содержит наивный рекурсивный алгоритм:

```
import time
t_start = time.perf_counter()
f0=0
f1=1
f = open('input.txt')
n = f.read()
f.close()
n = int(n)
if n < 0 or n > 45 :
    print("[Ошибка] Число должно быть не меньше 0 и не больше 45!")
else:
    def fibonacci(number):
        if number <= 1:
            return number
        a, b = 0, 1
        for _ in range(2, number + 1):
            a, b = b, b + a
        return b
    file = open('output.txt','w')
    file.write(str(fibonacci(n)))
    file.close()
print("В секундах" , (time.perf_counter() - t_start))
```

- создаем def с нахождением числа фиб.
- в начале даем значения для 0 и 1 числа значения соответственно
- дальше аналогично с предыдущими заданиями работаем с файлами

## Задание 3

Определение последней цифры большого числа Фибоначчи. Числа Фибоначчи растут экспоненциально. Например,

```
a = open('input.txt', 'r')
b = a.read()
a.close()

b = int(b)

def fibonacci(number):
    if number <= 1:
        return number
    a, b = 0, 1
    for _ in range(2, number + 1):
        a, b = b, b+a
    return b

if b < 0 or b > 10**7:
    print("Диапазон чисел нарушен")
else:
    m = open('output.txt', 'w')
    m.write(str(fibonacci(b) % 10))
    m.close()
```

- аналогично с предыдущими заданиями
- берем последнюю цифру с помощью % 10

- диапазон так же от 0 до  $10^7$
- выводит последнюю цифру для фибоначчи
- input = 10, тогда output = 5, т.к. последняя цифра "55" равняется "5"

## Задание 4

**Задача:** вам необходимо протестировать время выполнения вашего алгоритма в Задании 2 и Задании 3.

```
import time
t_start = time.perf_counter()
#код
f0=0
f1=1
f = open('input.txt')
n = f.read()
f.close()
n = int(n)
if n < 0 or n > 45 :
    print("[Ошибка] Число должно быть не меньше 0 и не больше 45!")
else:
    def fibonacci(number):
        if number <= 1:
            return number
        a, b = 0, 1
        for _ in range(2, number + 1):
            a, b = b, b + a
        return b
    file = open('output.txt','w')
    file.write(str(fibonacci(n)))
    file.close()
#код
print("В секундах" , (time.perf_counter() - t_start))
```



- используем модуль time
- в начале получаем начальное время, в конце от нынешнего отнимаем начальное время

- В секундах 0.0005889000021852553