

САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ
ФАКУЛЬТЕТ ИНФОКОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ

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

Выполнила:
Иванова Аайа Гаврильевна
К3141

Проверил:
Афанасьев А.В.

Санкт-Петербург
2024 г.

Содержание отчета

Содержание отчета	2
Задачи по варианту	3
Задание №1. Ввод-вывод	3
Задание №2. Число Фибоначчи	7
Задание №3. Еще про числа Фибоначчи	8
Задание №4. Тестирование ваших алгоритмов	10
Вывод	12

Задачи по варианту

Задание №1. Ввод-вывод

Вам необходимо выполнить 4 следующих задачи:

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

1. Задача $a + b$.

Листинг кода:

```
a, b = map(int, input().split(' '))
if -10**9 <= a <= 10**9 and -10**9 <= b <= 10**9:
    print(a+b)
else:
    print('Numbers are out of range')
```

Текстовое объяснение задачи:

В первой строке вводим переменные a и b , используя функцию `input()`, разделяем их методом `split()` и приводим к числовому типу данных. Затем проверяем полученные числа на соответствие условию: если они подходят, производим вывод их суммы $a + b$, иначе выводим сообщение о несоответствии условию.

Результат работы кода на примерах задачи:

```
/Users/carlia/PycharmProjects/alg_lab0_task11/.venv/bin/python /Users/carlia/PycharmProjects/alg_lab0_task11/main.py
1 2
3
Process finished with exit code 0
```

```
/Users/carlia/PycharmProjects/alg_lab0_task11/.venv/bin/python /Users/carlia/PycharmProjects/alg_lab0_task11/main.py
1000000000 -1000000000
0
Process finished with exit code 0
```

2. Задача $a + b^2$.

Листинг кода:

```
a, b = map(int, input().split(' '))
if -10**9 <= a <= 10**9 and -10**9 <= b <= 10**9:
    print(a+b*b)
else:
    print('Numbers are out of range')
```

Текстовое объяснение задачи:

В первой строке вводим переменные a и b , используя функцию `input()`, разделяем их методом `split()` и приводим к числовому типу данных. Затем проверяем полученные числа на соответствие условию: если они подходят, производим вывод их суммы $a + b^2$, иначе выводим сообщение о несоответствии условию.

Результат работы кода на примерах задачи:

```
/Users/carlia/PycharmProjects/alg_lab0_task12/.venv/bin/python /Users/carlia/PycharmProjects/alg_lab0_task12/main.py
1 3
10
Process finished with exit code 0
```

```
/Users/carlia/PycharmProjects/alg_lab0_task12/.venv/bin/python /Users/carlia/PycharmProjects/alg_lab0_task12/main.py
289 1000000000
Numbers are out of range
Process finished with exit code 0
```

3. Задача $a + b$ с использованием файлов.

Листинг кода:

```
with open('input.txt','r') as f:
```

```

    a,b = map(int, f.readline().split())
f.close()

if -10**9 <= a <= 10**9 and -10**9 <= b <= 10**9:
    with open('output.txt', 'w') as f:
        f.write(str(a + b))
    f.close()
else:
    print('Numbers are out of range')

```

Текстовое объяснение задачи:

В первой строке открываем файл для чтения, затем во второй строке получаем из файла переменные a и b , приводя их к числовому типу данных. Закрываем файл. Проверяем числа на соответствие условию. Если числа подходят, открываем выходной файл и записываем туда $a + b$, закрываем файл. Иначе выводим сообщение о том, что числа не подходят.

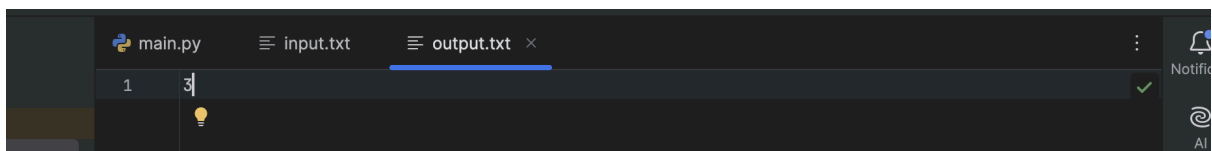
Результат работы кода на примерах задачи:

```

/Users/carlia/PycharmProjects/alg_lab0_task13/.venv/bin/python /Users/carlia/PycharmProjects/alg_lab0_task13/main.py
1 2

Process finished with exit code 0

```

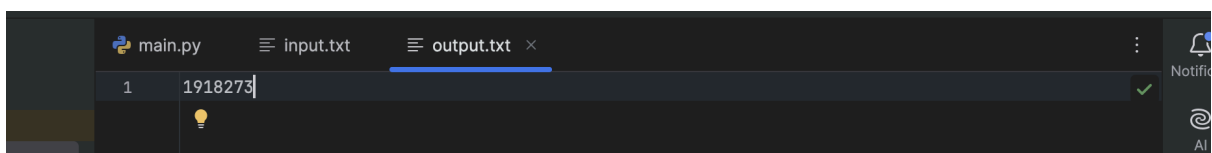


```

/Users/carlia/PycharmProjects/alg_lab0_task13/.venv/bin/python /Users/carlia/PycharmProjects/alg_lab0_task13/main.py
918273 1000000

Process finished with exit code 0

```



4. Задача $a + b^2$ с использованием файлов.

Листинг кода:

```

with open('input.txt','r') as f:
    a,b = map(int, f.readline().split())
f.close()

```

```
if -10**9 <= a <= 10**9 and -10**9 <= b <= 10**9:
    with open('output.txt', 'w') as f:
        f.write(str(a + b*b))
    f.close()
else:
    print('Numbers are out of range')
```

Текстовое объяснение задачи:

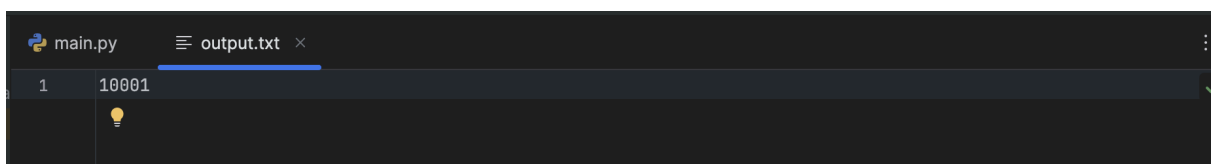
В первой строке открываем файл для чтения, затем во второй строке получаем из файла переменные *a* и *b*, приводя их к числовому типу данных. Закрываем файл. Проверяем числа на соответствие условию. Если числа подходят, открываем выходной файл и записываем туда $a + b^2$, закрываем файл. Иначе выводим сообщение о том, что числа не подходят.

Результат работы кода на примерах задачи:

```
/Users/carlia/PycharmProjects/alg_lab0_task14/.venv/bin/python /Users/carlia/PycharmProjects/alg_lab0_task14/main.py
1000000001 9889
Numbers are out of range

Process finished with exit code 0
```

```
/Users/carlia/PycharmProjects/alg_lab0_task14/.venv/bin/python /Users/carlia/PycharmProjects/alg_lab0_task14/main.py
1 100
Process finished with exit code 0
```



Вывод по задаче: Вспомнила, как считывать данные с входной строки и файла и записывать данные в файл

Задание №2. Число Фибоначчи

Определение последовательности Фибоначчи:

$$\begin{aligned} F_0 &= 0 \\ F_1 &= 1 \\ F_i &= F_{i-1} + F_{i-2} \text{ для } i \geq 2. \end{aligned} \tag{1}$$

Таким образом, каждое число Фибоначчи представляет собой сумму двух предыдущих, что дает последовательность

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, ...

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

Листинг кода:

```
def calc_fib(n):
    if n <= 1:
        return n
    else:
        s = [0, 1]
        for i in range(2, n+1):
            s.append(s[i-1] + s[i-2])
        return s[n]

with open('input.txt', 'r') as f:
    n = int(f.readline())
f.close()

if 0 <= n <= 45:
    a = calc_fib(n)
    with open('output.txt', 'w') as f:
        f.write(str(a))
    f.close()
else:
    print('Numbers are out of range')
```

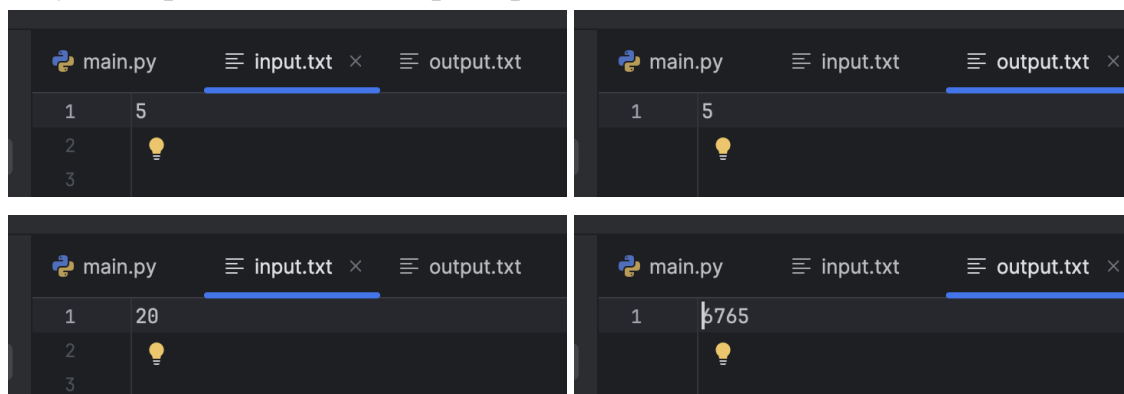
Текстовое объяснение задачи:

1. Создаем функцию, которая будет находить число Фибоначчи. На входе подается одно число - номер числа Фибоначчи. Если $n = 0$ или $n = 1$, число Фибоначчи равно номеру этого числа. В случае $n > 1$

создаем список с двумя элементами - 0 и 1, и добавляем к этому списку следующие элементы с помощью функции `append()`, n -ый элемент этого списка - число Фибоначчи с номером n .

2. В первой строке открываем файл для чтения, затем во второй строке получаем из файла переменную n . Закрываем файл. Проверяем соответствует ли число условию. Если соответствует, открываем выходной файл и пишем туда a , закрываем файл. Иначе выводим, что число не подходит.

Результат работы кода на примерах задачи:



Вывод по задаче: Вспомнила, как использовать функции, а также подумала над эффективным алгоритмом для нахождения числа Фибоначчи.

Задание №3. Еще про числа Фибоначчи

Листинг кода:

```
def calc_fib(n):
    if n <= 1:
        return n
    else:
        s1, s2 = 0, 1
        for i in range(2, n+1):
            s1, s2 = s2, (s1 + s2) % 10
        return s2

with open('input.txt', 'r') as f:
    n = int(f.readline())
f.close()
```



```

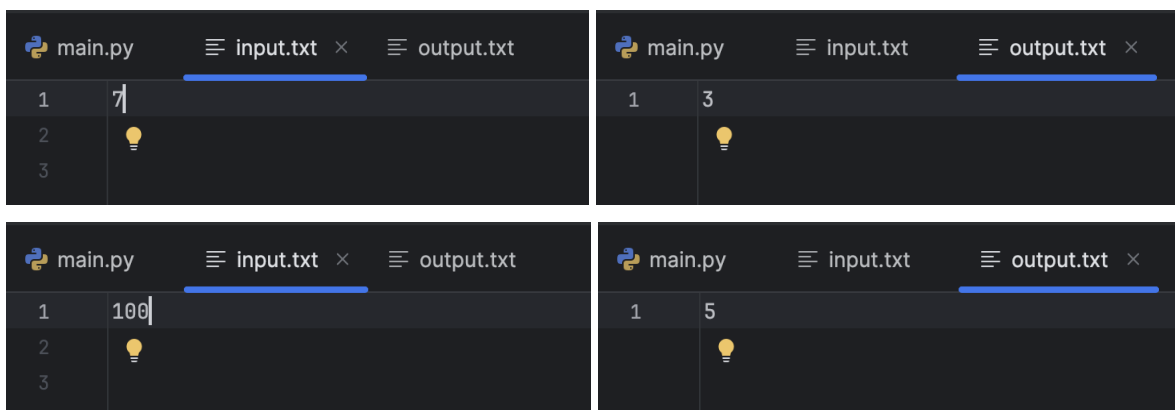
if 0 <= n <= 10**7:
    a = calc_fib(n)
    with open('output.txt', 'w') as f:
        f.write(str(a))
    f.close()
else:
    print('Numbers are out of range')

```

Текстовое объяснение задачи:

3. Создаем функцию, которая будет находить последнюю цифру числа Фибоначчи. На входе подается одно число - номер числа Фибоначчи. Если $n = 0$ или $n = 1$, последняя цифра равна номеру этого числа. В случае $n > 1$ присваиваем к $s1$ и $s2$ значения 0 и 1 соответственно, далее присваиваем к ним последующие последние цифры чисел Фибоначчи.
4. В первой строке открываем файл для чтения, затем во второй строке получаем из файла переменную n . Закрываем файл. Проверяем соответствует ли число условию. Если соответствует, открываем выходной файл и пишем туда a , закрываем файл. Иначе выводим, что число не подходит.

Результат работы кода на примерах задачи:



Вывод по задаче: Вспомнила, как использовать функции, а также подумала над эффективным алгоритмом для нахождения последней цифры числа Фибоначчи.

Задание №4. Тестирование ваших алгоритмов

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

Дополнительно: вы можете протестировать объем используемой памяти при выполнении вашего алгоритма.

Задача №2:

```
import time
t_start = time.perf_counter()

def calc_fib(n):
    if n <= 1:
        return n
    else:
        s = [0, 1]
        for i in range(2, n+1):
            s.append(s[i-1] + s[i-2])
        return s[n]

with open('input.txt', 'r') as f:
    n = int(f.readline())
f.close()

if 0 <= n <= 45:
    a = calc_fib(n)
    with open('output.txt', 'w') as f:
        f.write(str(a))
    f.close()
else:
    print('Numbers are out of range')

print(time.perf_counter() - t_start)
```

Задача №3:

```
import time
t_start = time.perf_counter()

def calc_fib(n):
    if n <= 1:
        return n
```

```

else:
    s1, s2 = 0, 1
    for i in range(2, n+1):
        s1, s2 = s2, (s1 + s2) % 10
    return s2

with open('input.txt', 'r') as f:
    n = int(f.readline())
f.close()

if 0 <= n <= 10**7:
    a = calc_fib(n)
    with open('output.txt', 'w') as f:
        f.write(str(a))

print(time.perf_counter() - t_start)

```

Текстовое объяснение задачи:

В первой строке подключаем библиотеку time для подсчета времени выполнения код, начинаем отсчет времени работы программы. В последней строке кода отнимаем от времени на момент окончания работы программы значение t_start.

Результат работы кода на примерах задачи:

Задача №2:

The screenshots illustrate the execution of the code for two different input values:

- Input: 0** - The output field shows a lightbulb icon, indicating a calculation or error.
- Input: 45** - The output field shows the result of the calculation, which is 1134903170.

The terminal output for the first execution (input 0) shows the following command and result:

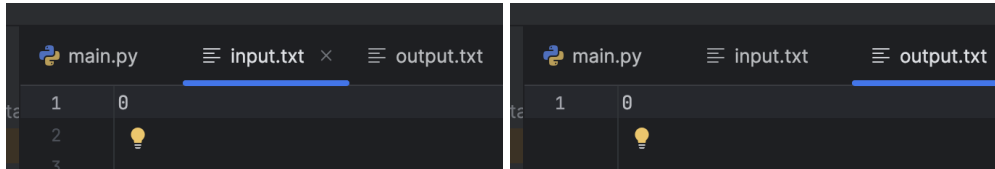
```

/Users/carlia/PycharmProjects/alg_lab0_task42/.venv/bin/python /Users/carlia/PycharmProjects/alg_lab0_task42/main.py
0.00045257300371304154
Process finished with exit code 0

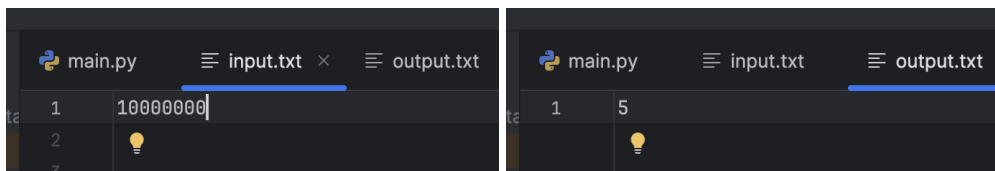
```

```
/Users/carlia/PycharmProjects/alg_lab0_task42/.venv/bin/python /Users/carlia/PycharmProjects/alg_lab0_task42/main.py  
0.0011068389940191992  
  
Process finished with exit code 0
```

Задача №3:



```
/Users/carlia/PycharmProjects/alg_lab0_task43/.venv/bin/python /Users/carlia/PycharmProjects/alg_lab0_task43/main.py  
0.0013770459991064854  
  
Process finished with exit code 0
```



```
/Users/carlia/PycharmProjects/alg_lab0_task43/.venv/bin/python /Users/carlia/PycharmProjects/alg_lab0_task43/main.py  
0.783498894998047  
  
Process finished with exit code 0
```

Вывод по задаче: Научилась вычислять время выполнения программы.

Вывод

Использовала различные инструменты при работе с файлами. Разработала эффективные алгоритмы для нахождения числа Фибоначчи и последней цифры числа Фибоначчи по номеру числа Фибоначчи. Измерила время выполнения этих программ.