

**Министерство образования и науки Российской Федерации**  
**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**  
**УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,**  
**МЕХАНИКИ И ОПТИКИ**

Факультет программной инженерии и компьютерной техники  
Направление подготовки 09.03.04 Программная инженерия

Дисциплина «Алгоритмы и структуры данных»

**ОТЧЁТ**

по лабораторной работе Введение (Stepic)

Молодецкий Арсений  
группа Р3217

Санкт-Петербург

2019 г.

## Содержание

Задача 1: небольшое число Фибоначчи .....	3
Исходный код к задаче 1 .....	3
Задача 2: последняя цифра большого числа Фибоначчи .....	3
Исходный код к задаче 2 .....	4
Задача 3: огромное число Фибоначчи по модулю .....	4
Исходный код к задаче 3 .....	4
Задача 4: наибольший общий делитель .....	5
Исходный код к задаче 4 .....	5
Тест: правила работы с логарифмами .....	<b>Ошибка! Закладка не определена.</b>
Тест: правильная скорость роста .....	<b>Ошибка! Закладка не определена.</b>
Тест: правильная скорость роста .....	<b>Ошибка! Закладка не определена.</b>
Тест повышенной сложности: правильная скорость роста .....	<b>Ошибка! Закладка не определена.</b>

## Задача 1: небольшое число Фибоначчи

Дано целое число  $1 \leq n \leq 40$

, необходимо вычислить  $n$ -е число Фибоначчи (напомним, что  $F_0=0$ ,  $F_1=1$  и  $F_n=F_{n-1}+F_{n-2}$  при  $n \geq 2$

).

### Sample Input:

3

### Sample Output:

2

## Исходный код к задаче 1

```
elements = []

elements.append(1)

elements.append(1)

def fib(n):

    if n <= len(elements):

        return elements[n - 1]

    value = fib(n-1) + fib(n-2)

    elements.append(value)

    return value

def main():

    n = int(input())

    print(fib(n))

if __name__ == "__main__":

    main()
```

## Задача 2: последняя цифра большого числа Фибоначчи

Дано число  $1 \leq n \leq 10^7$ , необходимо найти последнюю цифру  $n$ -го числа Фибоначчи.

Как мы помним, числа Фибоначчи растут очень быстро, поэтому при их вычислении нужно быть аккуратным с переполнением. В данной задаче, впрочем, этой проблемы можно избежать, поскольку нас интересует только последняя цифра числа Фибоначчи: если  $0 \leq a, b \leq 9$

— последние цифры чисел  $F_i$  и  $F_{i+1}$  соответственно, то  $(a+b)\bmod 10$  — последняя цифра числа  $F_{i+2}$

### Sample Input:

317457

### Sample Output:

2

### Исходный код к задаче 2

```
def fib_digit(n):  
    f = [1,1]  
    for i in range(2,n):  
        f[i % 2] = (f[i % 2] + f[(i + 1) % 2]) % 10  
    return f[(n - 1) % 2]  
  
def main():  
    n = int(input())  
    print(fib_digit(n))  
  
if __name__ == "__main__":  
    main()
```

### Задача 3: огромное число Фибоначчи по модулю

Даны целые числа  $1 \leq n \leq 10^{18}$  и  $2 \leq m \leq 10^5$ , необходимо найти остаток от деления  $n$ -го числа Фибоначчи на  $m$

### Sample Input:

10 2

### Sample Output:

1

### Исходный код к задаче 3

```
class Lab1_3  
{  
    private static long CalcPisanoPeriods(long divider)  
    {  
        long a = 0;  
        long b = 1;  
        long c;  
  
        for (long i = 0L; i < divider * divider; i++)  
        {  
            c = (a + b) % divider;  
            a = b;  
            b = c;  
        }  
    }  
}
```

```

        if (a == 0 && b == 1)
        {
            return i + 1;
        }
    }
    return a;
}

private long GetHugeFibonacci(long elementNumber, long divider)
{
    long remainder = elementNumber % CalcPisanoPeriods(divider);

    long first = 0L;
    long second = 1L;

    long result = remainder;

    for (int i = 1; i < remainder; i++)
    {
        result = (first + second) % divider;
        first = second;
        second = result;
    }

    return result % divider;
}

private void DoWork(string[] args)
{
    var input = Console.ReadLine().Split(' ').Select(long.Parse).ToArray();
    Console.WriteLine(this.GetHugeFibonacci(input[0], input[1]));
}

public static void Main(string[] args)
{
    var app = new Lab1_3();
    app.DoWork(args);
}
}

```

## Задача 4: наибольший общий делитель

По данным двум числам  $1 \leq a, b \leq 2 \cdot 10^9$  найдите их наибольший общий делитель.

### Sample Input 1:

18 35

### Sample Output 1:

1

### Sample Input 2:

14159572 63967072

### Sample Output 2:

4

## Исходный код к задаче 4

```

def gcd(a, b):
    if a == 0:
        return b

```

```
elif b == 0:
    return a
elif a >= b:
    return gcd(a % b, b)
else:
    return gcd(a, b % a)

def main():
    a, b = map(int, input().split())
    print(gcd(a, b))

if __name__ == "__main__":
    main()
```