

Отчет по лабораторной работе номер 5

Хамбалеев Булат Галимович

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение работы	7
4	Библиография	11
5	Выводы	12

List of Tables

List of Figures

3.1	рис.1. Алгоритм.	7
3.2	рис.2. Якоби.	8
3.3	рис.3. Теста Соловья-Штрассена.	9
3.4	рис.4. Миллера-Рабин.	10

1 Цель работы

Реализовать алгоритмы определения простоты числа.

2 Задание

Задание подразумевает реализацию алгоритма определения простоты числа на языке программирования Python.

3 Выполнение работы

1. Реализуем функцию алгоритма Ферма.(рис. 1)

```
def Ferma(n):  
    a = random.randint(2,n-2)  
    r = (a**(n-1))%n  
    if r==1:  
        print('Число',n,'вероятно, простое.')  
    else:  
        print('Число',n,'составное.')
```

```
Ferma(5)
```

Число 5 ,вероятно, простое.

```
Ferma(100)
```

Число 100 составное.

```
Ferma(773)
```

Число 773 ,вероятно, простое.

```
Ferma(2343)
```

Число 2343 составное.

Figure 3.1: рис.1. Алгоритм.

2. Реализуем алгоритм нахождения числа Якоби. (рис. 2)

```

def Jacobi(n,a):
    g = 1
    while True:
        if a ==0:
            return 0
        if a==1:
            return g
        k = 1
        while (a/2**k)%2==0:
            k+=1
        a1 = a/2**k
        if k%2==0:
            s=1
        else:
            if n==1%8 or n==-1%8:
                s=1
            else:
                s=-1
        if a1==1:
            return g*s
        if n==3%4 and a1==3%4:
            s=-s
        a = n%a1
        n = a1
        g = g*s

```

Jacobi(3,1)

1

Jacobi(3,2)

-1

Jacobi(4,0)

0

Jacobi(7,2)

1

Figure 3.2: рис.2. Якоби.

3. Реализуем алгоритм теста Соловья-Штрассена.

```
def Shtrassen(n):  
    a = random.randint(2,n-3)  
    r = (a**((n-1)/2))%n  
    if r!=1 and r!=n-1:  
        return 'Число составное'  
    s = Jacobi(n,a)  
    if r==s%n:  
        return 'Число простое'  
    else:  
        return 'Число составное'
```

```
Shtrassen(10)
```

'Число составное'

```
Shtrassen(100)
```

'Число составное'

```
Shtrassen(5)
```

'Число простое'

```
Shtrassen(120)
```

'Число составное'

Figure 3.3: рис.3. Теста Соловья-Штрассена.

4. Реализуем алгоритм теста Миллера-Рабина.

```
def Miller(n):
    s = 1
    while ((n-1)/2**s)%2==0:
        s+=1
    r = ((n-1)/2**s)
    a = random.randint(2,n-3)
    y = (a**r)%n
    if y!=1 and y!=n-1:
        j=1
        if j<=s-1 and y!=n-1:
            y=(y**2)%n
            if y==1:
                return 'Составное'
            j+=1
        if y!=n-1:
            return 'Составное'
    return 'Простое'
```

Miller(5)

'Простое'

Miller(10)

'Составное'

Miller(13)

'Простое'

Miller(122)

'Составное'

Figure 3.4: рис.4. Миллера-Рабин.

4 Библиография

1. ТУИС РУДН

5 Выводы

Во время выполнения лабораторной работы я на практике реализовал несколько алгоритмов на определение простоты числа.