Вероятностные алгоритмы проверки чисел на простоту.

Кейела Патачона

11 декабря, 2021, Москва, Россия

Российский Университет Дружбы Народов

Цели и задачи

Цель лабораторной работы

Изучить вероятностные алгоритмы проверки чисел на простоту.

Выполнение лабораторной работы

Тест Ферма

Вход. Нечетное целое цисло n>=5.

Выход. “Число n,вероятно,простое” или “Число n составное”.

1. Выбрать случайное целое число a, 2<=a<=2.
2. Вычислить $r = a^{(n-1)} (\mathrm{mod}\ n)$.
3. Если r = 0 результат : “Число n,вероятно,простое”.В противном случае результат: “Число n составное”.

Символ Якоби

Вход. Нечетное целое цисло n>=3, целое число a,0 <= a < n.

Выход. Символ Якоби.

1. g=1
2. если a =0 результат: 0
3. если a =1 результат: g
4. прадствить a в виде $a = 2^k a_1$, где a_1 нечетное.
5. при четном k положить s=1, при нечетном положить s=1, если n=abs(1(mod8));положить s=-1, если n=abs(3(mod8))

Символ Якоби

6. при a_1 результат: gs
7. если n = 3(mod4) and a_1 = 3(mod4) , то s = -s
8. положить a = n mod(a_1) n = a_1 g = gs и вернуться на шаг 2

Алгоритм , реализующий тест Соловея - Штрассена

Вход. Нечетное целое цисло n>=5.

Выход. “Число n,вероятно,простое” или “Число n составное”.

1. Выбрать случайное целое число a, 2<=a<=2.
2. Вычислить $r = a^{(n+1)/2} (\mathrm{mod}\ n)$
3. Если r не равен 1 и n-1 реузультат: “Число n составное”.
4. Вычислить символ Якоби s = (a/n)
5. Если r = s(mod n) реузультат: “Число n составное”, иначе “Число n,вероятно,простое”.

Алгоритм , реализующий тест Миллера - Рабина

Вход. Нечетное целое цисло n>=5.

Выход. “Число n,вероятно,простое” или “Число n составное”.

1. представить n-1 в виде $n-1 = 2^s r$, где r нечетное
2. выбрать случайное целое число a, 2<=a<=2
3. вычислить $y = a^r \pmod n$
4. при y не равном 1 и n-1 выполнить следующее

4.1. положить j = 1

Алгоритм , реализующий тест Миллера - Рабина

```
4.2. если j <= s-1 и y не равен n-1 ,то

    4.2.1. положить y = y^2^(mod n)

    4.2.2. при y = 1 результат: "Число n составное"

    4.2.3. положить j = j+1

4.3. при y не равном n-1 результат: "Число n составное"
```

5. Результат: “Число n,вероятно,простое”

Контрольные примеры

laboratory > lab05 > task_1.py > ...

```python
import random

n = int(input('Введите нечетное целое число n>=5: '))
a = random.randint(2, n - 2)
r = (a ** (n - 1)) % n
if r == 1:
    print(f'Число {n} ,вероятно, простое')
else:
    print(f'Число {n} - составное')

```

ПРОБЛЕМЫ ВЫХОДНЫЕ ДАННЫЕ ТЕРМИНАЛ КОНСОЛЬ ОТЛАДКИ

```
PS C:\Users\patat\Desktop\Master Rudn\Git_work\2021-2022\Cybersecurity> & C:/Users/patat/AppDat
sktop/Master Rudn/Git_work/2021-2022/Cybersecurity/laboratory/lab05/task_1.py"
Введите нечетное целое число n>=5: 11
Число 11 ,вероятно, простое
PS C:\Users\patat\Desktop\Master Rudn\Git_work\2021-2022\Cybersecurity>
```

Figure 1: Тест Ферма

laboratory > lab05 > task_2.py > jacobian_symbol

```python
def jacobian_symbol(a, n):
    g = 1

    while True:
        if a == 0:
            return 0
        elif a == 1:
            return g
        else:
            k, a1 = 0, a
            while a1 % 2 == 0:
                k += 1
```

Контрольные примеры

```
laboratory > lab05 > task_3.py > ...
1   import random
2   from task_2 import jacobian_symbol
3
4   n = int(input('Введите нечетное целое число n>=5: '))
5   a = random.randint(2, n - 2)
6   r = a ** ((n - 1) / 2) % n
7   if r != 1 and r != n - 1:
8       print(f'Число {n} - составное')
9   else:
10      s = jacobian_symbol(a, n)
11      if r % n == s:
12          print(f'Число {n} составное')

ПРОБЛЕМЫ  ВЫХОДНЫЕ ДАННЫЕ  ТЕРМИНАЛ  КОНСОЛЬ ОТЛАДКИ

PS C:\Users\patat\Desktop\Master Rudn\Git_work\2021-2022\Cybersecurity> & C:/Users/patat/AppData/Local/Programs/Pytho
sktop/Master Rudn/Git_work/2021-2022/Cybersecurity/laboratory/lab05/task_3.py"
Введите нечетное целое число n>=5: 17
Число 17 ,вероятно, простое
PS C:\Users\patat\Desktop\Master Rudn\Git_work\2021-2022\Cybersecurity>
```

Figure 3: Алгоритм , реализующий тест Соловея - Штрассена

```
laboratory > lab05 > task_4.py > ...
1   import random
2
3   n = int(input('Введите нечетное целое число n>=5: '))
4   s, r = 0, n - 1
5   while r % 2 == 0:
6       s += 1
7       r //= 2
8   a = random.randint(2, n - 2)
9   y = (a ** r) % n
10  if y != 1 and y != n - 1:
11      j = 1
12      if j <= s - 1 and y != n - 1:

ПРОБЛЕМЫ  ВЫХОДНЫЕ ДАННЫЕ  ТЕРМИНАЛ  КОНСОЛЬ ОТЛАДКИ

PS C:\Users\patat\Desktop\Master Rudn\Git_work\2021-2022\Cybersecurity> & C:/Users/patat/AppDat
```

Выводы

Результаты выполнения лабораторной работы

Мной были узчены вероятностные алгоритмы проверки чисел на простоту.