

Лабораторная работа №5

Вероятностные алгоритмы проверки чисел на простоту

Кузнецов Юрий Владимирович

Содержание I

- 1 Введение
- 2 Тест Ферма
- 3 Символ Якоби
- 4 Тест Соловья-Штрассена
- 5 Тест Миллера-Рабина

Содержание II



Заключение

Section 1

Введение

Введение

В данной презентации будет представлена реализация вероятностных алгоритмов проверки чисел на простоту

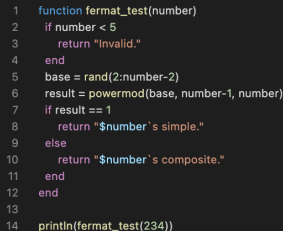
Основные темы

- Тест Ферма
- Символ Якоби
- Тест Соловья-Штрассена
- Тест Миллера-Рабина

Section 2

Тест Ферма

Тест Ферма



```
1 function fermat_test(number)
2   if number < 5
3     return "Invalid."
4   end
5   base = rand(2:number-2)
6   result = powermod(base, number-1, number)
7   if result == 1
8     return "$number's simple."
9   else
10    return "$number's composite."
11  end
12 end
13
14 println(fermat_test(234))
```

Рис. 1: Тест Ферма

Section 3

Символ Якоби

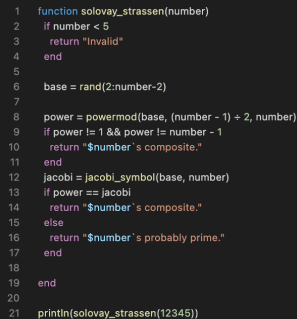
Символ Якоби

```
1 function jacobi_symbol(a, n)
2   if n < 3 || a >= n || a < 0
3     return "Invalid"
4   end
5   result = 1
6   while a != 0
7     count = 0
8     while a % 2 == 0
9       count += 1
10      a += 2
11    end
12
13    if count % 2 == 1 && (n % 8 == 3 || n % 8 == 5)
14      result *= -1
15    end
16
17    if n % 4 == 3 && a % 4 == 3
18      result *= -1
19    end
20
21    a, n = n % a, a
22  end
23  return n == 1 ? result : 0
24 end
25
26 println(jacobi_symbol(23, 32))
```

Section 4

Тест Соловья-Штрассена

Тест Соловья-Штрассена



```
1 function solovay_strassen(number)
2   if number < 5
3     return "Invalid"
4   end
5
6   base = rand(2:number-2)
7
8   power = powermod(base, (number - 1) ÷ 2, number)
9   if power != 1 && power != number - 1
10    return "$number's composite."
11  end
12  jacobi = jacobi_symbol(base, number)
13  if power == jacobi
14    return "$number's composite."
15  else
16    return "$number's probably prime."
17  end
18
19 end
20
21 println(solovay_strassen(12345))
```

Рис. 3: Тест Соловья-Штрассена

Section 5

Тест Миллера-Рабина

Тест Миллера-Рабина

```
1 function miller_rabin(number)
2   if number < 5
3     return "invalid"
4   end
5   remainder = number - 1
6   power = 0
7   while remainder % 2 == 0
8     power += 1
9     remainder -= 2
10  end
11  base = rand(2,number-2)
12  current = powmod(base, remainder, number)
13  if current != 1 && current != number-1
14    for _ in 1:power-1
15      current = (current*2) % number
16      if current == 1
17        return "$number" s composite."
18      end
19    end
20    if current != number-1
21      return "$number" s composite."
22    end
23  end
24  return "$number" s simple."
25 end
26
27 println(miller_rabin(12349))
```

Рис. 4: Тест Миллера-Рабина

Section 6

Заключение

Заключение

В ходе выполнения лабораторной работы, были изучены и запрограммированы вероятностные алгоритмы проверки чисел на простоту