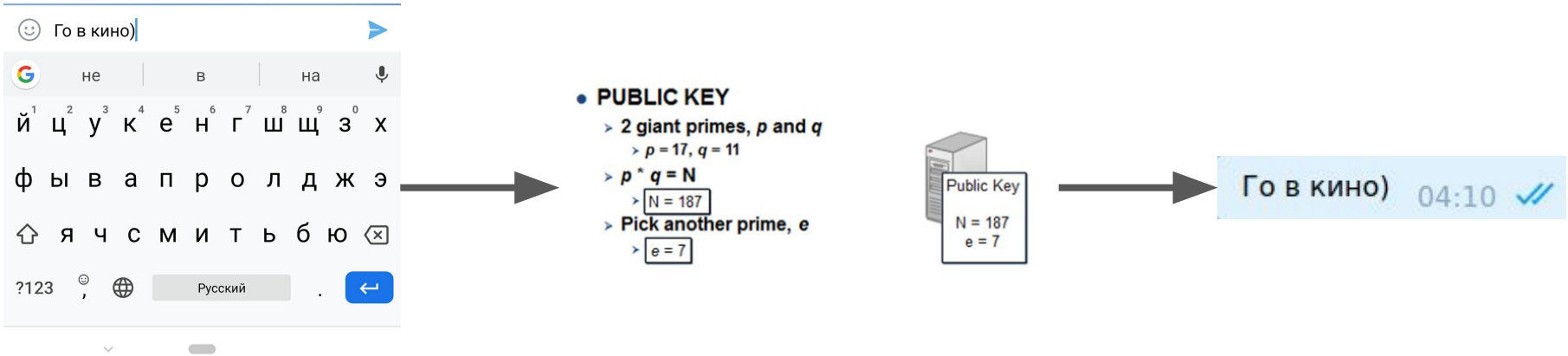
Вероятностная проверка на простоту

Тест Миллера-Рабина

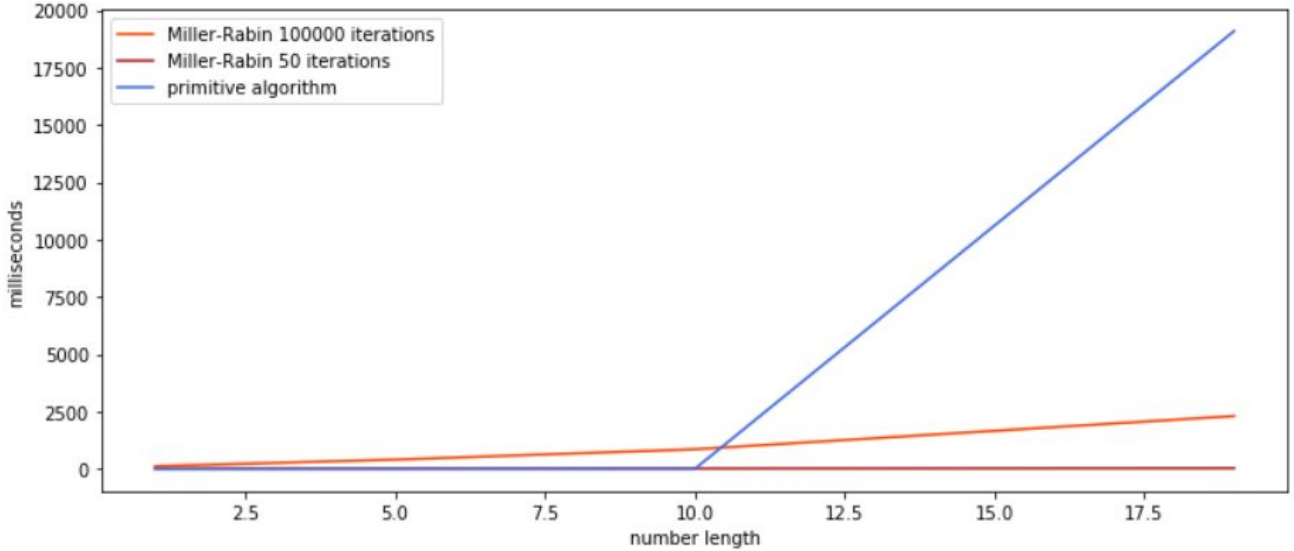
Матвей Волков
Курс по Kotlin, MIPT

2019

Как позвать девушку на свидание



Зачем вероятностный алгоритм



64-битное число проверяется **20 секунд**

Рекомендуемая длина RSA ключа **2048 бит**

Для его генерации потребуются минуты

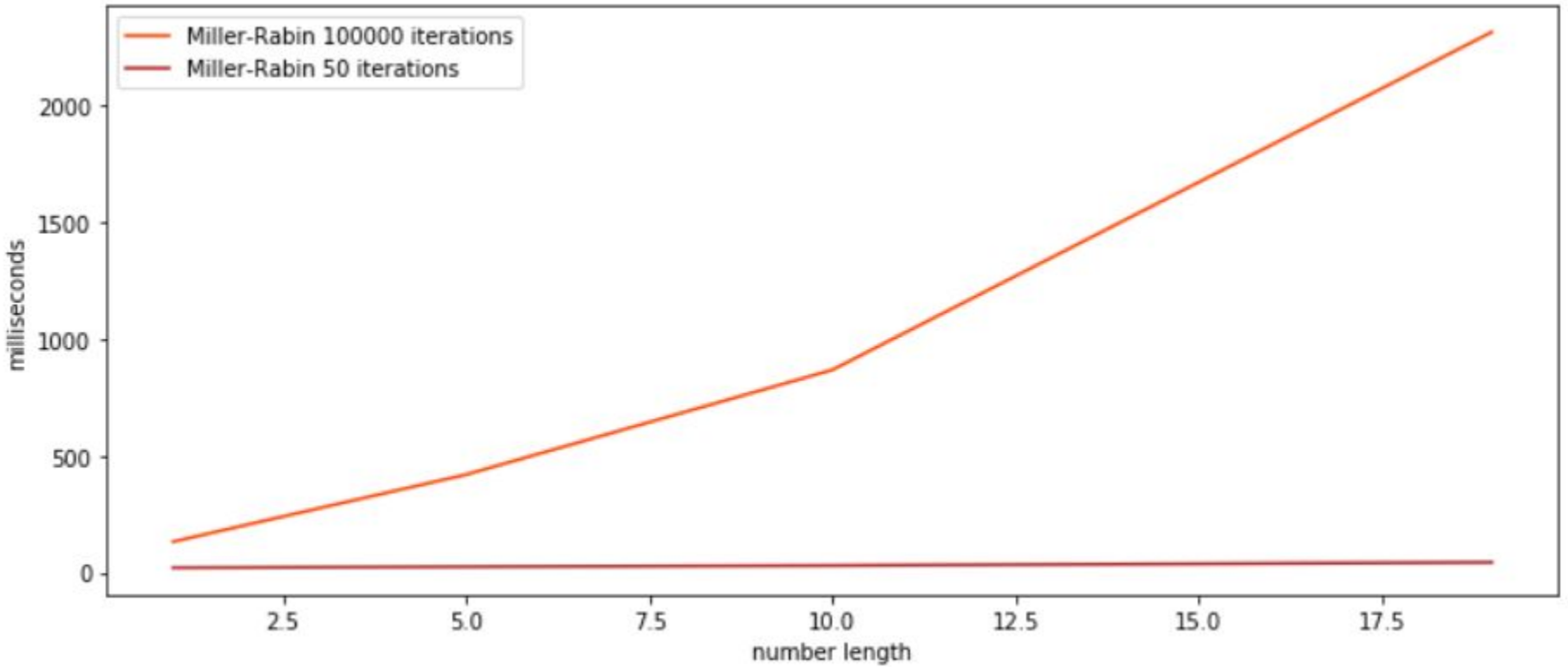
Как поможет вероятность?

Пусть n — простое число и $n - 1 = 2^s d$, где d — нечётно. Тогда для любого a из $\mathbb{Z}_n$ выполняется хотя бы одно из условий:

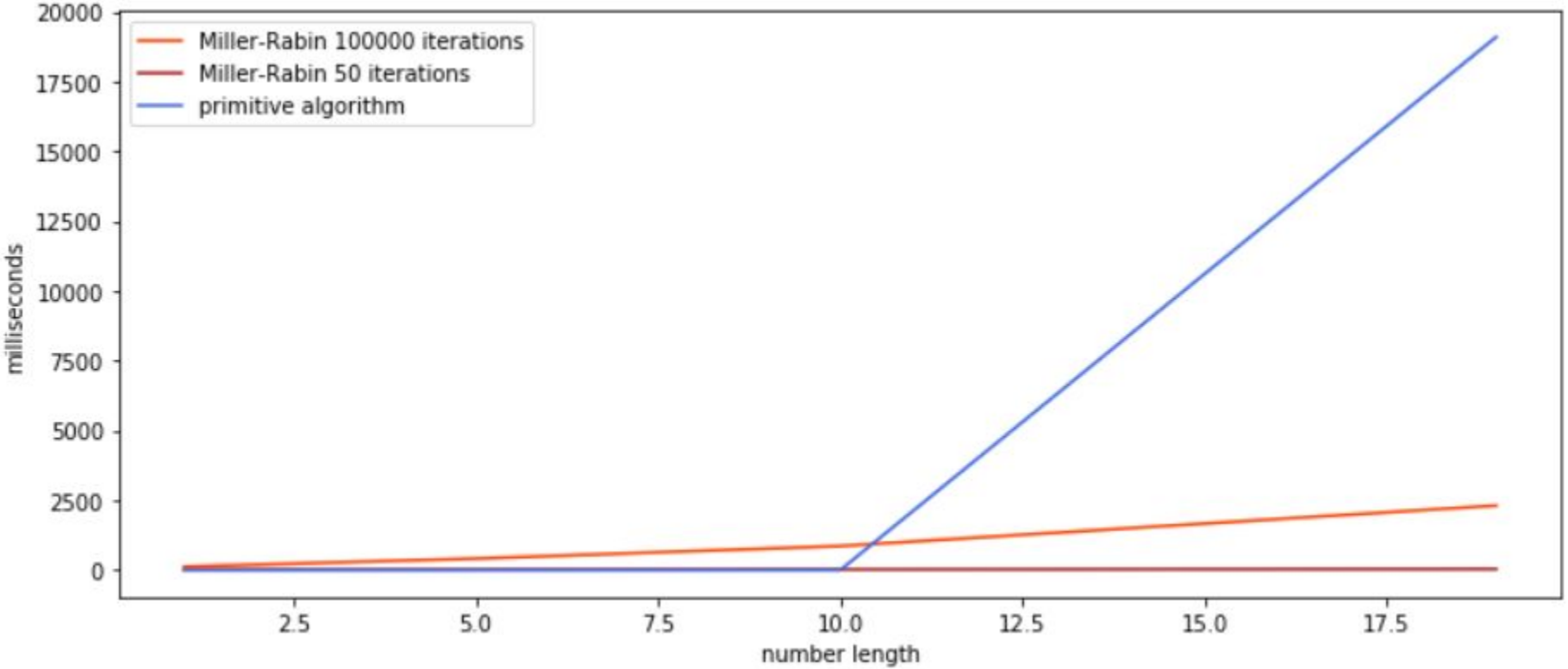
1. $a^d \equiv 1 \pmod{n}$
2. Существует целое число $r < s$ такое что $a^{2^r d} \equiv -1 \pmod{n}$

Вероятность ошибки **4^(-k)**
При 10 итерациях это 0,000000954

Бенчмарки


Miller-Rabin 100000 iterations
Miller-Rabin 50 iterations
2000
1500
1000
500
0
milliseconds
2.5
5.0
7.5
10.0
12.5
15.0
17.5
number length

Бенчмарки


Miller-Rabin 100000 iterations
Miller-Rabin 50 iterations
primitive algorithm
20000
17500
15000
12500
10000
7500
5000
2500
0
milliseconds
2.5
5.0
7.5
10.0
12.5
15.0
17.5
number length

Сложность работы

Используя быстрое возведение в степень по модулю, $O(k \log^3 n)$

Наибольший вклад вносит возведение в степень

Если использовать быстрое преобразование Фурье, $\widetilde{O}(k \log^2 (n))$


Теперь данные в безопасности