

Пример: сумма троек

- Сумма троек. Возьмем N целых чисел. Сколько сумм трех чисел равно нулю?

```
% more 8ints.txt
8
30 -40 -20 -10 40 0 10 5

% java ThreeSum 8ints.txt
4
```

	a[i]	a[j]	a[k]	sum
1	30	-40	10	0
2	30	-20	-10	0
3	-40	40	0	0
4	-10	0	10	0

Сумма троек: метод грубой силы

```
public class ThreeSum
{
    public static int count(int[] a)
    {
        int N = a.length;
        int count = 0;
        for (int i = 0; i < N; i++)
            for (int j = i+1; j < N; j++)
                for (int k = j+1; k < N; k++)
                    if (a[i] + a[j] + a[k] == 0)
                        count++;
        return count;
    }

    public static void main(String[] args)
    {
        int[] a = In.readInts(args[0]);
        StdOut.println(count(a));
    }
}
```

← check each triple
← for simplicity, ignore integer overflow

Q. How to time a program?

A. Automatic.

```
public class Stopwatch (part of stdlib.jar)
```

```
    Stopwatch() create a new stopwatch
```

```
    double elapsedTime() time since creation (in seconds)
```

```
public static void main(String[] args)
{
    int[] a = In.readInts(args[0]);
    Stopwatch stopwatch = new Stopwatch();
    StdOut.println(ThreeSum.count(a));
    double time = stopwatch.elapsedTime();
}
```

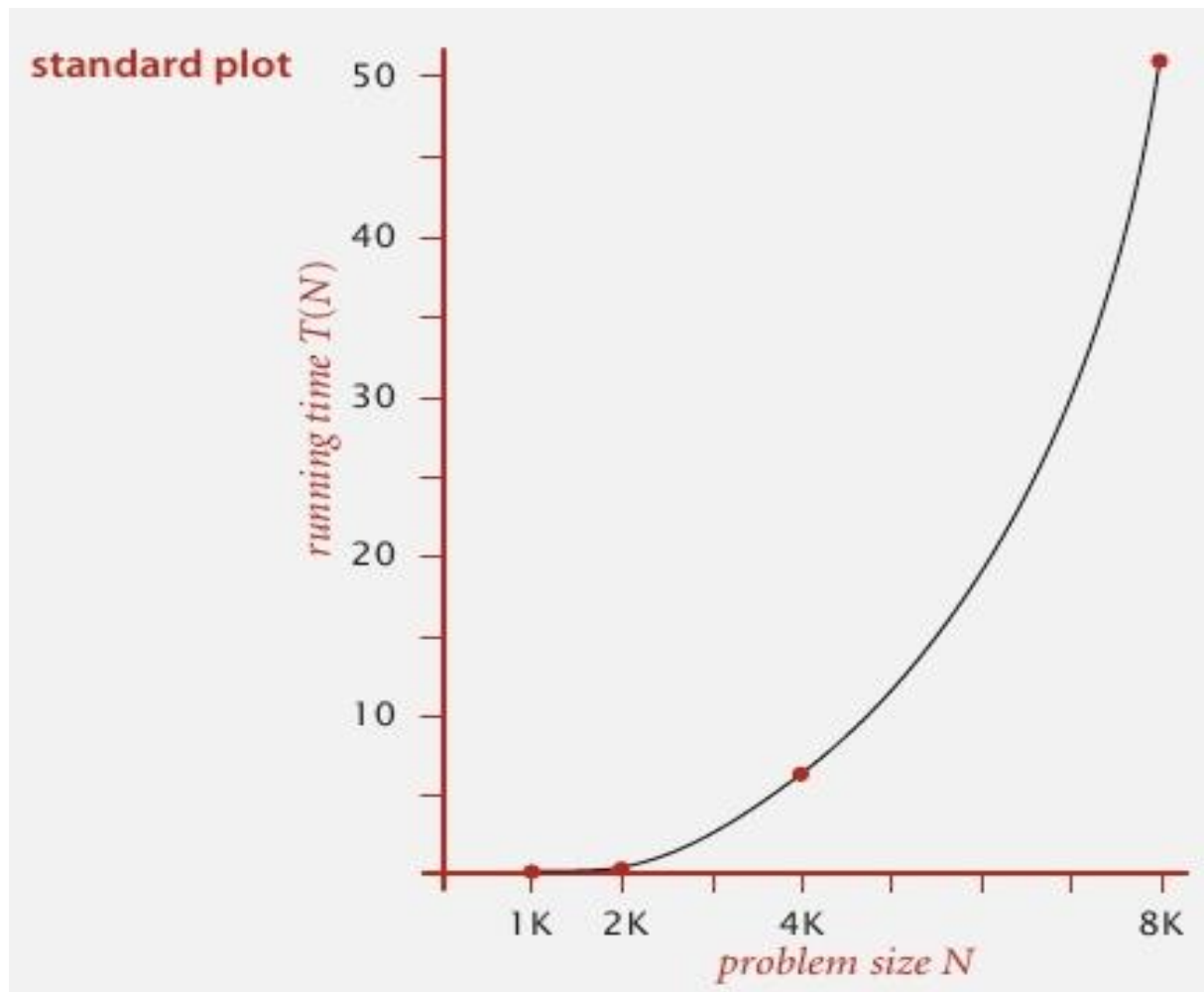
Эмпирический анализ

- Запуск программы с различными входными данными и измерение времени выполнения

N	time (seconds) †
250	0.0
500	0.0
1,000	0.1
2,000	0.8
4,000	6.4
8,000	51.1
16,000	?

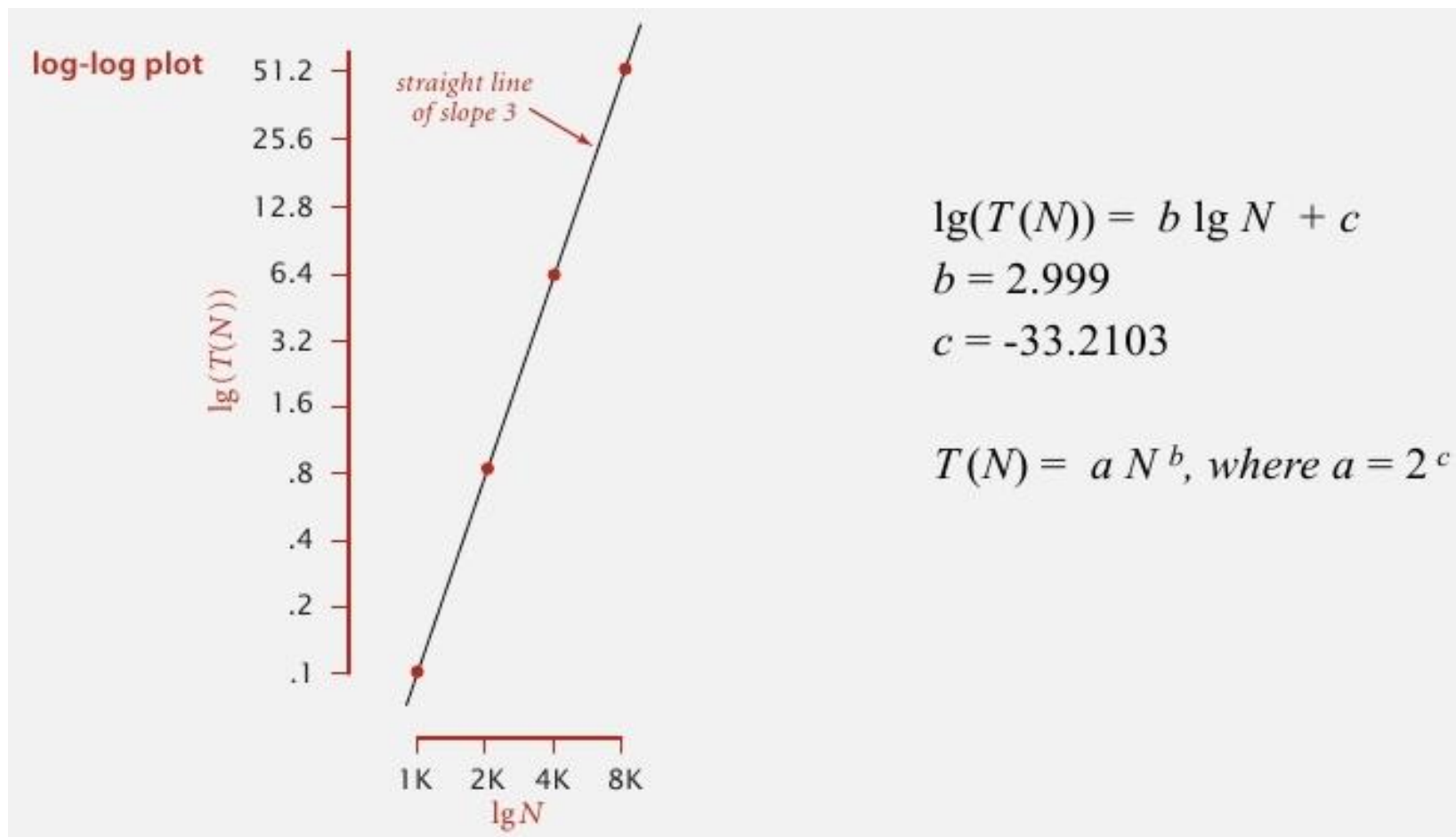
Анализ данных

- Шкала с линейным масштабом



Свойства быстрого поиска

- Логарифмическая шкала



- Регрессия. Провести прямую линию через точки:
 aN^b

Предсказание и проверка

- **Гипотеза.** Время выполнения $1,006 * 10^{-10} * N^{2,999}$ секунд
- **Предсказание.**
 - 51,0 секунда для $N = 8000$
 - 408,1 секунда для $N = 16000$
- **Наблюдение.**

N	time (seconds) †
8,000	51.1
8,000	51.0
8,000	51.1
16,000	410.8

validates hypothesis!

Удвоение гипотезы

- Запустить программу с удвоенным размером ВХОДНЫХ ДАННЫХ

N	time (seconds) †	ratio	lg ratio
250	0.0		-
500	0.0	4.8	2.3
1,000	0.1	6.9	2.8
2,000	0.8	7.7	2.9
4,000	6.4	8.0	3.0
8,000	51.1	8.0	3.0

↑
seems to converge to a constant $b \approx 3$

- **Гипотеза.** Время выполнения aN^b , где $b = \log_2$ (ratio)

Удвоение гипотезы

- Вопрос: Как оценить a (если мы знаем b)?
- Ответ: Запустить программу (на большом наборе данных N) и найти a .

N	time (seconds) †
8,000	51.1
8,000	51.0
8,000	51.1

$51.1 = a \times 8000^3$

$\Rightarrow a = 0.998 \times 10^{-10}$

- **Гипотеза.** Время выполнения $0,998 * 10^{-10} * N^3$

Экспериментальная алгоритмика

System independent effects.

- Algorithm.
 - Input data.
- } determines exponent b
in power law

System dependent effects.

- Hardware: CPU, memory, cache, ...
- Software: compiler, interpreter, garbage collector, ...
- System: operating system, network, other apps, ...

} determines constant a
in power law