

Skillbox

Алгоритм бинарного поиска

Java-разработчик с нуля

Поиск элемента перебором

Васильев
Андреев
Севостьянов
Матвеев
Константинов
Кандауров
Ярославцев
Боровой
Арбатов

```
for(String name : names)
{
    if(name.equals(query)) {
        return true;
    }
}
return false;
```

$O(n)$

Skillbox

Java-разработчик с нуля

Бинарный поиск

Андреев
Арбатов
Боровой
Васильев
Кандауров
Константинов
Матвеев
Севостьянов
Ярославцев

Матвеев

Skillbox

Java-разработчик с нуля

Бинарный поиск

Андреев
Арбатов
Боровой
Васильев
Кандауров
Константинов
Матвеев
Севостьянов
Ярославцев

Матвеев

Skillbox

Java-разработчик с нуля

Бинарный поиск

Андреев
Арбатов
Боровой
Васильев
Кандауров
Константинов
Матвеев
Севостьянов
Ярославцев

Матвеев

Skillbox

Java-разработчик с нуля

Бинарный поиск

Андреев
Арбатов
Боровой
Васильев
Кандауров
Константинов
Матвеев
Севостьянов
Ярославцев

Матвеев

Skillbox

Java-разработчик с нуля

Бинарный поиск

Андреев
Арбатов
Боровой
Васильев
Кандауров
Константинов
Лисенков
Севостьянов
Ярославцев

Матвеев

Skillbox

Java-разработчик с нуля

Бинарный поиск - временная сложность

1 000 000

Skillbox

Java-разработчик с нуля

Бинарный поиск - временная сложность

1 000 000
500 000

Skillbox

Java-разработчик с нуля

Бинарный поиск - временная сложность

1 000 000
500 000
250 000
125 000
62 500
31 250
15 625
7 812
3 906
1 953

976
488
244
122
61
30
15
7
3
1

Бинарный поиск - временная сложность

$$\frac{1\ 000\ 000}{2}$$

Skillbox

Java-разработчик с нуля

Бинарный поиск - временная сложность

$$\frac{1\ 000\ 000}{2^2}$$

Skillbox

Java-разработчик с нуля

Бинарный поиск - временная сложность

$$\frac{1\ 000\ 000}{2^3}$$

Skillbox

Java-разработчик с нуля

Бинарный поиск - временная сложность

$$\frac{1\ 000\ 000}{2^{19}}$$

Skillbox

Java-разработчик с нуля

Бинарный поиск - временная сложность

$$\frac{1\,000\,000}{2^{19}}$$

$$19 = \log_2 10\,000\,000$$

Skillbox

Java-разработчик с нуля

Бинарный поиск - временная сложность

$$\frac{1\,000\,000}{2^{19}}$$

$$19 = \log_2 10\,000\,000 = \frac{\ln 10\,000\,000}{\ln 2}$$

Skillbox

Java-разработчик с нуля

Бинарный поиск - временная сложность

$$\frac{1\,000\,000}{2^{19}}$$

$$19 = \log_2 1000000 = \frac{\ln 1000000}{\ln 2}$$

$O(\log n)$

Skillbox

Java-разработчик с нуля

Бинарный поиск - временная сложность

Количество элементов	Количество операций
10	3
100	6
1 000	9
10 000	13
100 000	16
1 000 000	19

Бинарный поиск - временная сложность

Количество элементов	Количество операций
10	3
100	6
1 000	9
10 000	13
100 000	16
1 000 000	19

$O(\log n)$

Skillbox
Java-разработчик с нуля

```
private int search(String query, int from, int to)
{
    int middle = (from + to) / 2;
    int comparison = query.compareTo(list.get(middle));
    if(comparison == 0) {
        return middle;
    }
    if(comparison > 0) {
        return search(query, middle, to);
    }
    if(comparison < 0) {
        return search(query, from, middle);
    }
}
```

Бинарный поиск в Java

`Collections.binarySearch()`

Skillbox

Java-разработчик с нуля