Государственное образовательное учреждение высшего профессионального образования

«Московский государственный технический университет имени Н.Э. Баумана»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ Информатика и системы управления

КАФЕДРА Программное обеспечение ЭВМ и информационные технологии

Обзор существующих алгоритмов поиска дубликатов в наборе данных

Студент: Коротыч Михаил Дмитриевич

Руководитель: Шикуть А. В.

Москва, 2022

Цель и задачи

Цель работы – обзор существующих методов поиска дубликатов в наборе данных и определение оптимального среди них.

Задачи работы:

1) рассмотреть проблему дубликации данных;
2) описать существующие алгоритмы поиска дубликатов;
3) провести всесторонний анализ таких алгоритмов;
4) определить оптимальный среди рассмотренных.

Определение дубликата. Проблема дубликации

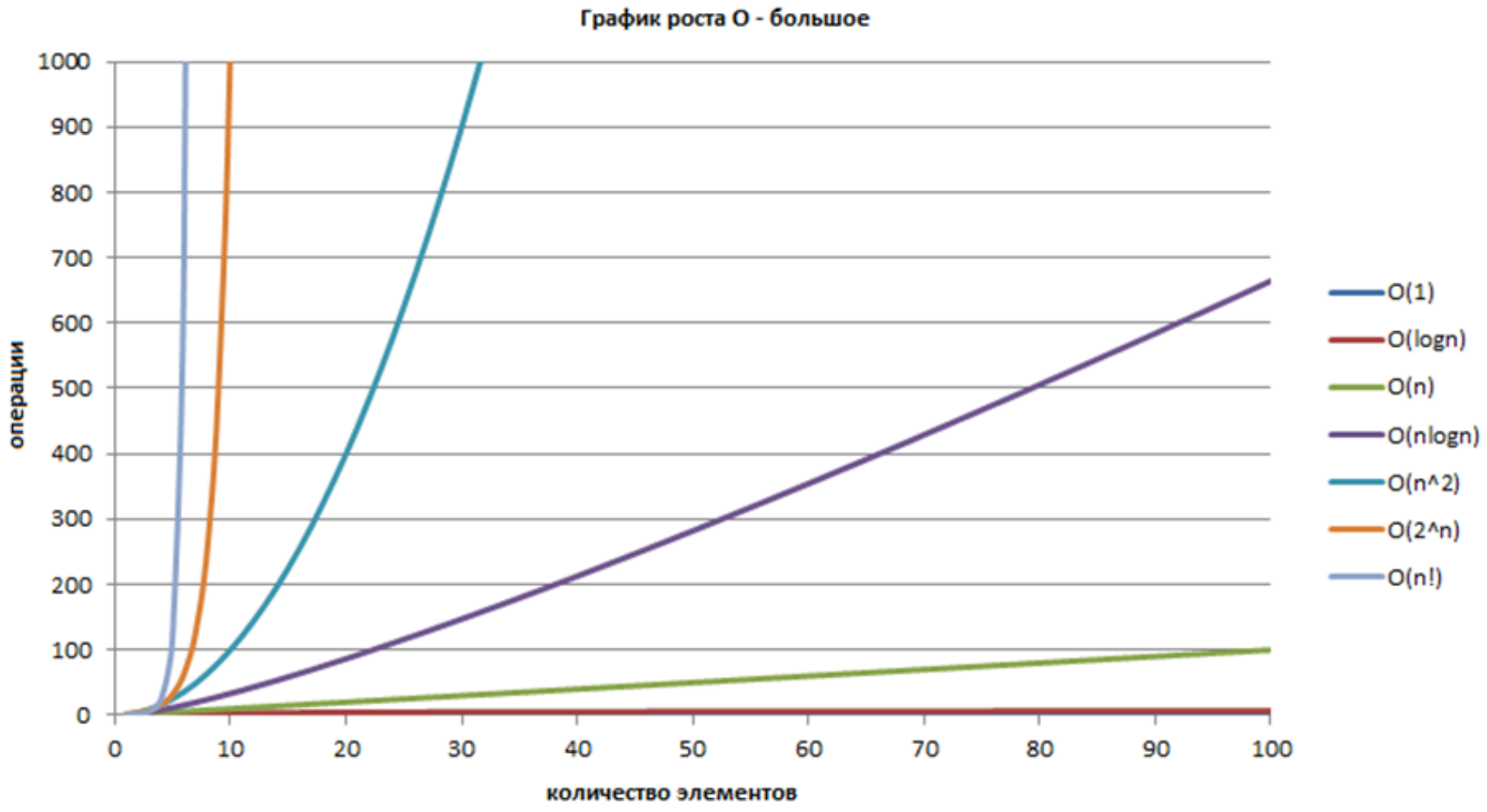
Дубликат - объект, который содержится в наборе данных два и более раз.

Проблема дубликатов – дубликация данных зачастую нарушает целостность самих данных и отношений между ними.

	department	fio	dt	status
1	ИТ	Иванов Иван Иванович	2020-01-15	Больничный
2	ИТ	Иванов Иван Иванович	2020-01-16	На работе
3	ИТ	Иванов Иван Иванович	2020-01-18	На работе
4	ИТ	Иванов Иван Иванович	2020-01-19	Оплачиваемый отпуск
5	ИТ	Иванов Иван Иванович	2020-01-20	Оплачиваемый отпуск
6	Бухгалтерия	Петрова Ирина Ивановна	2020-01-15	Оплачиваемый отпуск
7	Бухгалтерия	Петрова Ирина Ивановна	2020-01-16	На работе
8	Бухгалтерия	Петрова Ирина Ивановна	2020-01-17	На работе
9	Бухгалтерия	Петрова Ирина Ивановна	2020-01-18	На работе
10	Бухгалтерия	Петрова Ирина Ивановна	2020-01-19	Оплачиваемый отпуск
11	Бухгалтерия	Петрова Ирина Ивановна	2020-01-20	Оплачиваемый отпуск
12	ИТ	Иванов Иван Иванович	2020-01-16	На работе

Нотация «О большое»

Нотация «О большое» — это математическая нотация, которая описывает ограничивающее поведение функции, когда аргумент стремится к определенному значению или бесконечности.



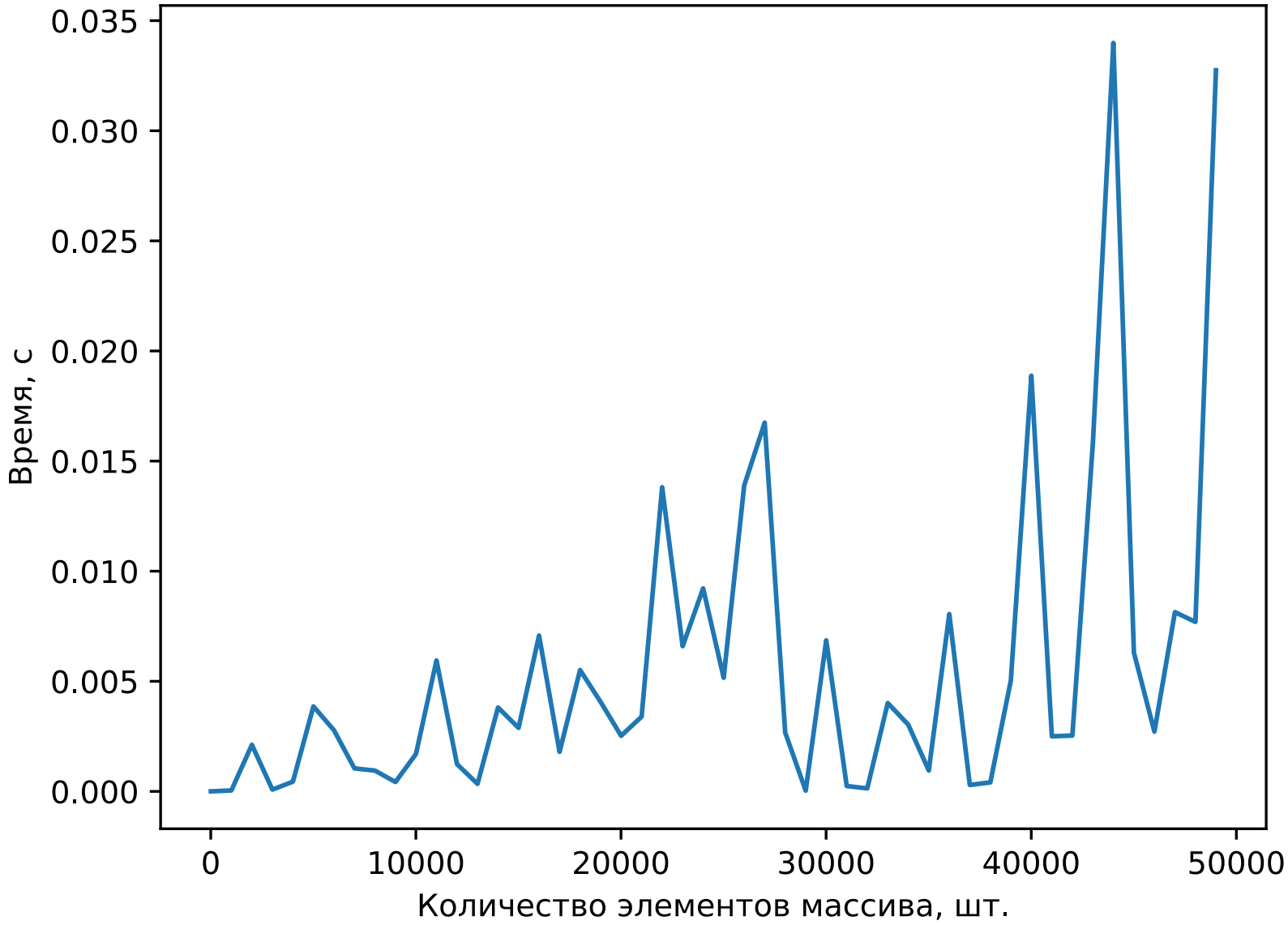
Критерии оптимальности

Алгоритм является оптимальным, если:

- любой другой алгоритм, решающий поставленную задачу, работает не быстрее рассматриваемого;
- он не изменяет каким-либо образом входные данные;

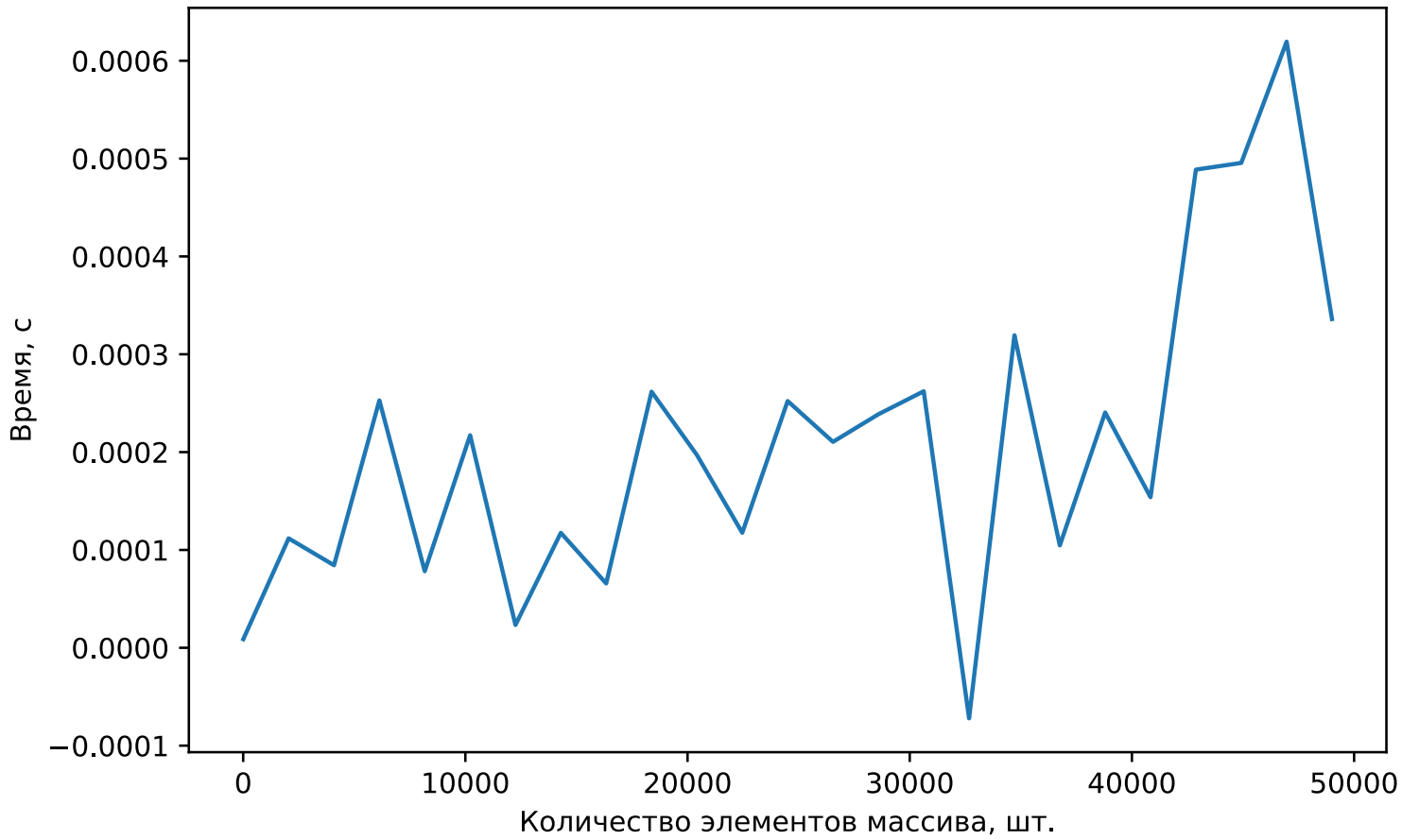
Brute force

- Решение «в лоб»
- Простое сравнение пар элементов
- Вычислительная сложность: $O(n^2)$
- Пространственная сложность: $O(1)$



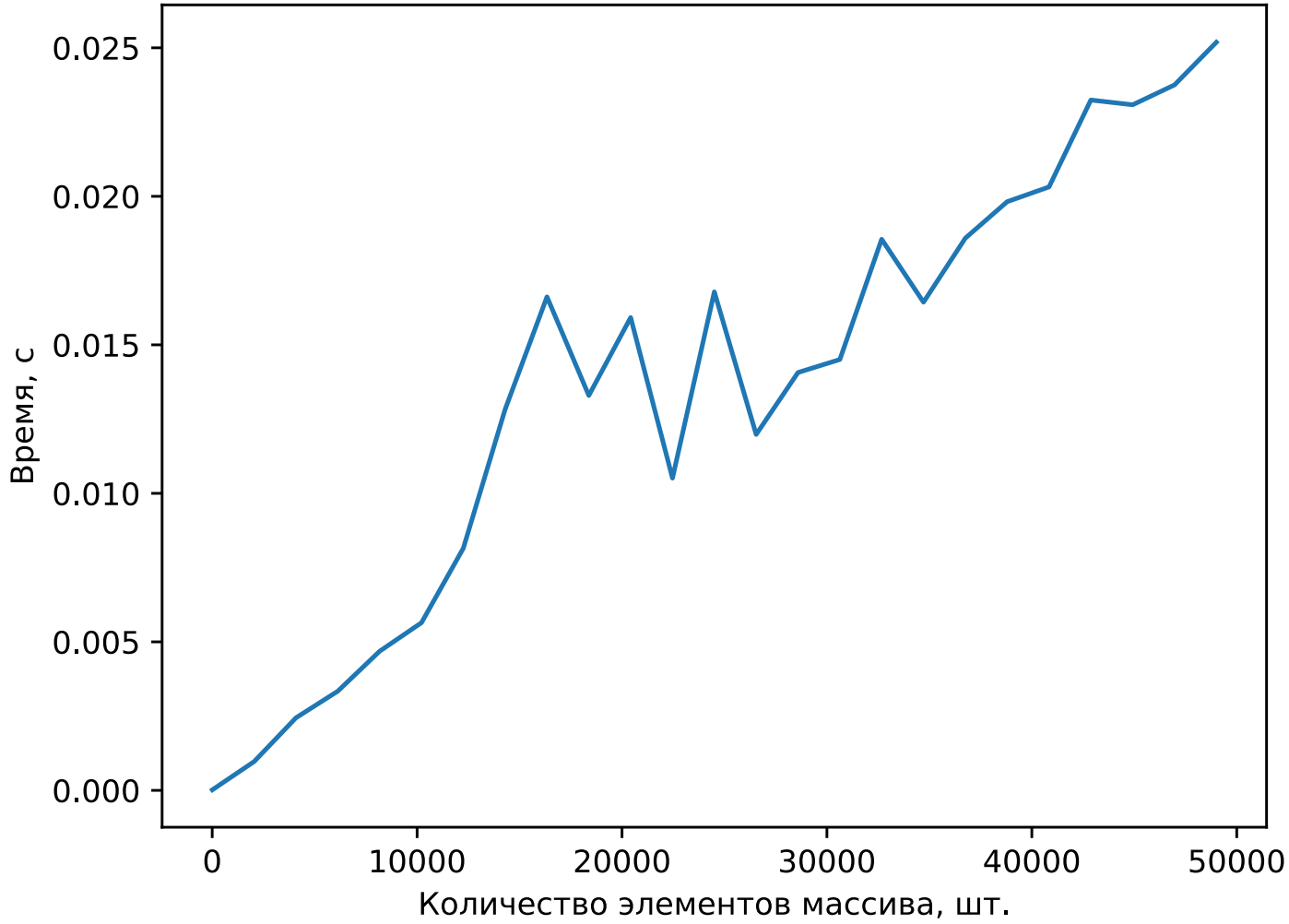
Подсчёт итераций

- Метод заключается в подсчёте итераций каждого целочисленного элемента
- Хорошо подходит для хеш-таблиц
- Временная сложность: $O(n)$
- Пространственная сложность: $O(n)$



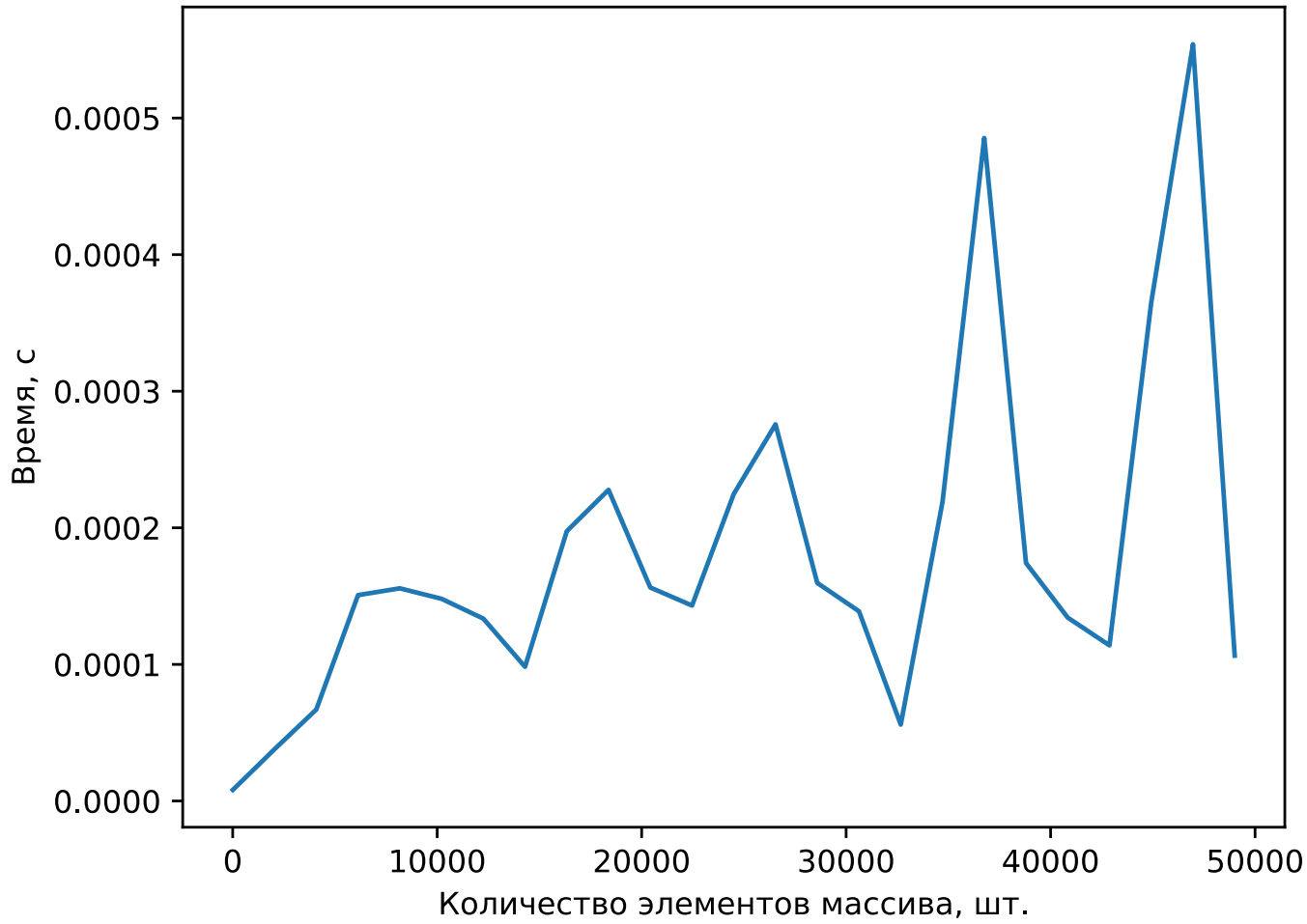
Метод предварительной сортровки

- Использует метод предварительной сортировки Timsort
- Временная сложность: $O(n * \log n)$
- Пространственная сложность: $O(1)$



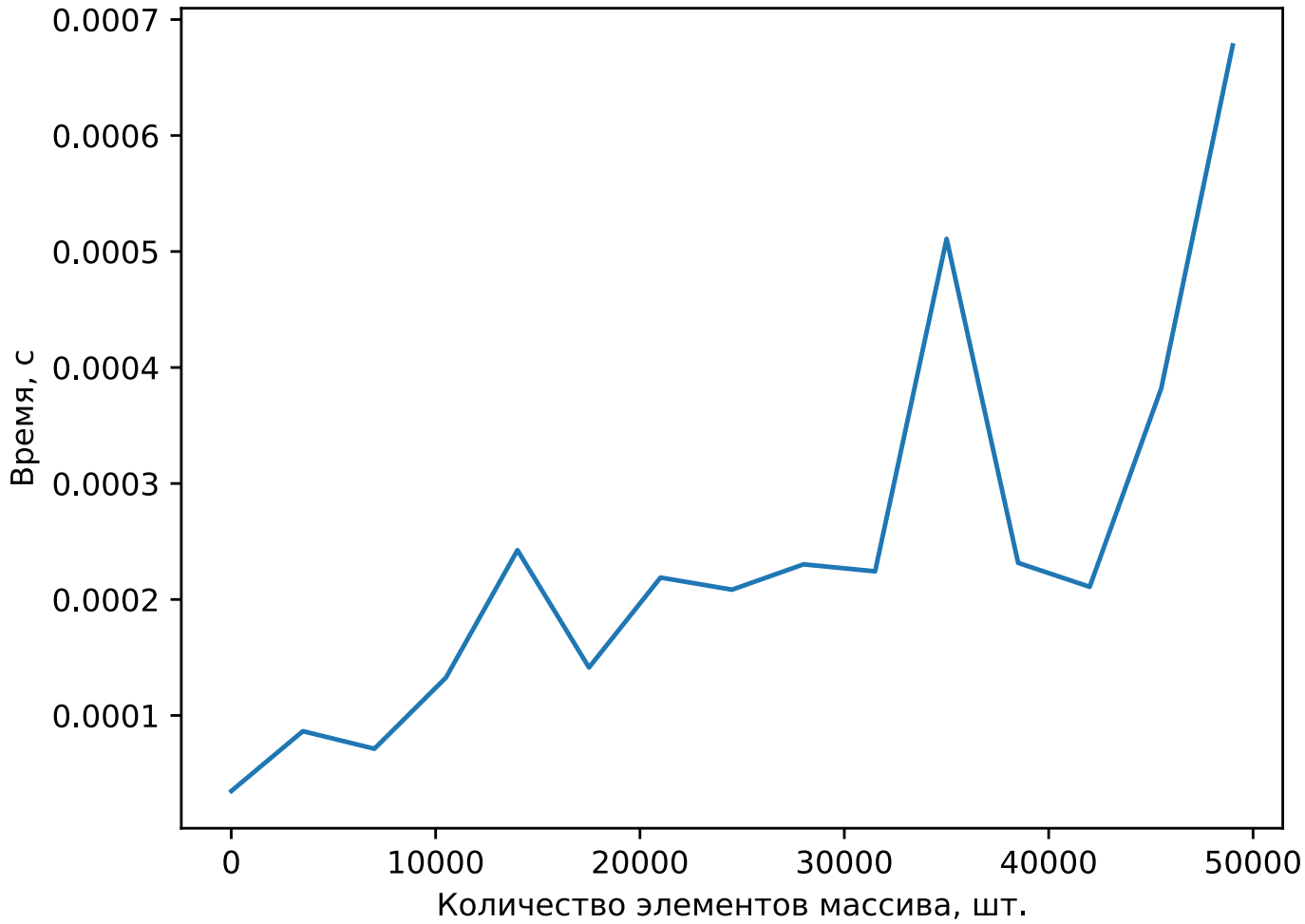
Метод подсчёта суммы элементов

- Результатом работы алгоритмы будет значение дублированного элемента
- Корректно работает только с одним дубликатом
- Рекурсия препятствует реализации
- Временная сложность: $O(n)$
- Пространственная сложность: $O(1)$



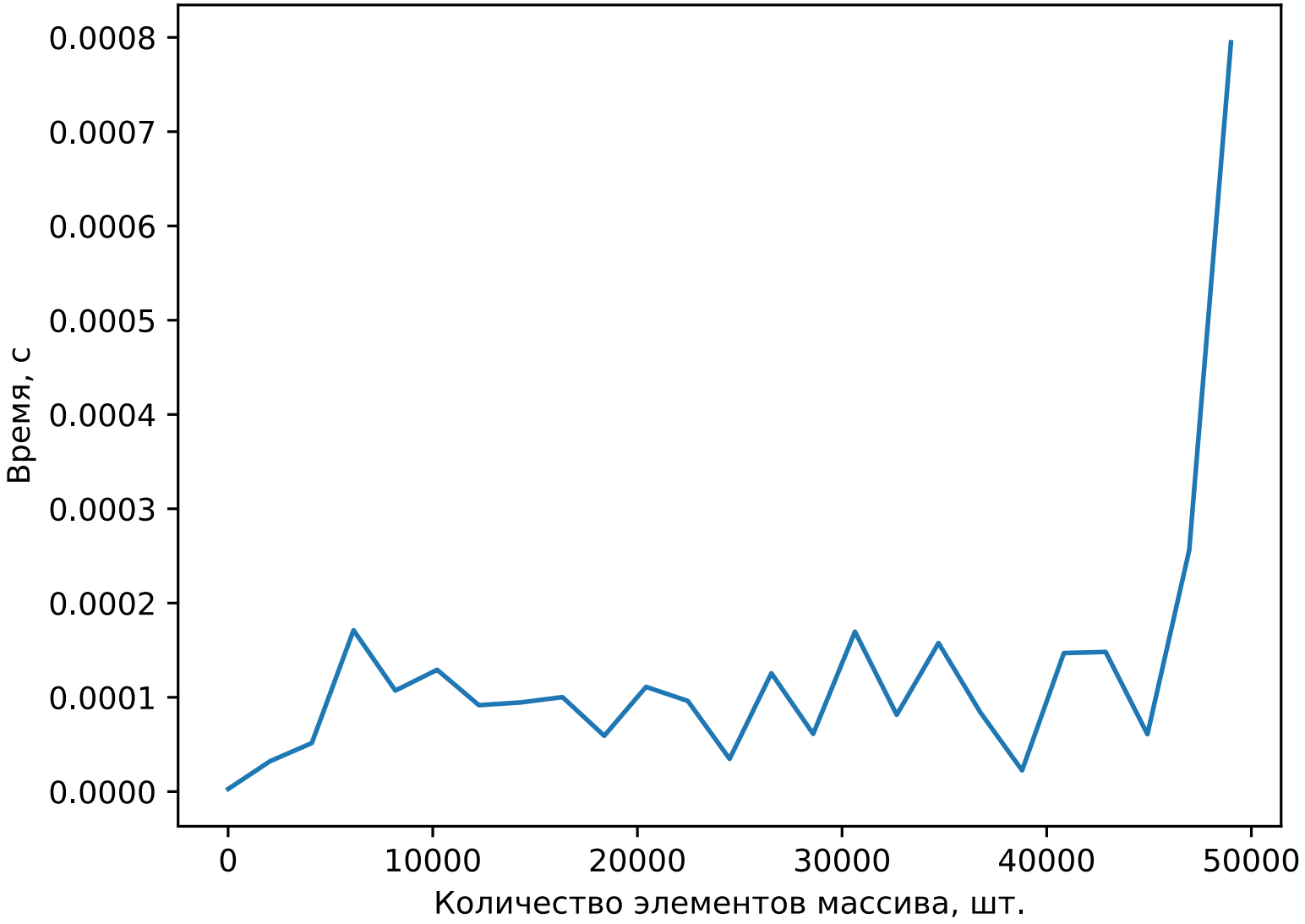
Метод маркера

- Использует концепцию массива как связного списка
- Можно искать несколько дубликатов одного и того же значения элемента или несколько дубликатов разных значений
- Временная сложность: $O(n)$
- Пространственная сложность: $O(1)$



Метод «бегуна»

- Также как и метод маркера использует связный список
- За основу реализации можно взять алгоритм Флойда
- Временная сложность: $O(n)$
- Пространственная сложность: $O(1)$



Сравнение характеристик

- Самый оптимальный – метод бегуна.
- Имеет $O(n)$ и $O(1)$ сложности.
- Не изменяет входной набор данных
- Позволяет найти несколько дубликатов

Алгоритм	Пространственная сложность	Временная сложность
Brute Force	$O(1)$	$O(n^2)$
Метод подсчёта итераций	$O(n)$	$O(n)$
Метод с предварительной сортировкой	$O(1)$	$O(n * \log n)$
Метод подсчёта суммы элементов	$O(1)$	$O(n)$
Метод маркера	$O(1)$	$O(n)$
Метод бегуна	$O(1)$	$O(n)$

Заключение

1. Проведён анализ предметной области.
2. Рассмотрена проблема дубликации данных.
3. Описаны существующие алгоритмы поиска дубликатов.
4. Проведён всесторонний анализ таких алгоритмов.
5. Определён оптимальный алгоритм среди представленных — алгоритм бегуна. Он обладает $O(1)$ пространственной сложностью и $O(n)$ временной сложностью и не изменяет входные данные.

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ Н.Э. БАУМАНА (НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ) МОСКВА, 2022 ГОД