

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский химико-технологический университет имени Д.И.
Менделеева»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №1

Выполнил студент группы КС-36: Золотухин А.А.

Ссылка на репозиторий: [https://github.com/
MUCTR-IKT-CPP/
ZolotukhinAA_36_ALG](https://github.com/MUCTR-IKT-CPP/ZolotukhinAA_36_ALG)

Принял: Крашенников Роман Сергеевич

Дата сдачи: 17.02.2025

Москва
2025

Оглавление

Описание задачи	1
Описание метода/модели	2
Выполнение задачи	3
Выводы	3

Описание задачи

В лабораторной работе предлагается изучить способ анализа алгоритма, связанный со временем. Рассмотреть для выбранного алгоритма сортировки наилучшее, наихудшее и среднее время и соотнести его с известным для алгоритма показателем эффективности O -большое.

Допускается реализация задания на любом языке программирования, кроме лиспоподобных. Преподаватель может не знать конкретного языка реализации, поэтому вы должны быть способны объяснить алгоритм и нарисовать его без демонстрации непосредственно вашего кода.

Задание:

- Реализовать метод сортировки: **Сортировка вставками**;
- Реализовать проведения тестирования алгоритма сериями расчетов для измерения параметров. За один расчёт выполняются следующие операции:
 - Генерируется массив случайных значений;
 - Запоминается время начала расчета алгоритма сортировки;
 - Выполняется алгоритм сортировки
 - * Во время выполнения измерить количество повторных проходов по массиву.
 - * Во время выполнения измерить количество выполнения операций обмена значений.
 - Вычисляется время, затраченное на сортировку: текущее время - время начала;
 - Сохраняется время для одной попытки. После этого расчёт повторяется до окончания серии.
 - * Алгоритм вычисляется 8 сериями по 20 раз за серию;
 - * Алгоритм в каждой серии вычисляется для массива размером M . (1000, 2000, 4000, 8000, 16000, 32000, 64000, 128000);
 - * Массив заполняется значениями чисел с плавающей точкой в интервале от -1 до 1;
 - * Для серии запоминаются все времена, которые были замерены.
- По полученным данным времени построить графики зависимости времени от числа элементов в массиве:
 - Совмещенный график наихудшего времени выполнения сортировки и сложности алгоритма, указанной в нотации O большое;

Для построения графика вычисляется O большое для каждого размера массива. При этом при вычислении функции $O(c * g(N))$ подбирается такая константа c , чтобы при значении >1000 график $O(N)$ был выше графика наихудшего случая, но второй график на его фоне не превращался в прямую линию.

- Совмещенный график среднего, наихудшего и наилучшего времени исполнения;
 - График среднего количества обмена значений;
 - График повторных обходов массива.
- По результатам расчётов оформляется отчёт по предоставленной форме, в отчете:
 - Приводится описание алгоритма;
 - Приводится описание выполнения задачи (описание кода и специфических элементов реализации);
 - Приводятся выводы (Графики и их анализ).

Описание метода/модели

Сортировка вставками - это простой алгоритм сортировки, который работает путём итеративной вставки каждого элемента несортированного списка в его правильное положение в отсортированной части списка. Это похоже на сортировку игральных карт в ваших руках. Вы разделяете карты на две группы: отсортированные карты и несортированные карты. Затем вы выбираете карточку из несортированной группы и помещаете её в нужное место в отсортированной группе.

Ход алгоритма:

1. Начинаем со второго элемента массива, поскольку предполагается, что первый элемент в массиве должен быть отсортирован;
2. Сравниваем второй элемент с первым и проверяем, не меньше ли второй элемент, затем меняем их местами;
3. Переходим к третьему элементу, сравниваем его с первыми двумя элементами и устанавливаем в правильное положение;
4. Повторяем до тех пор, пока не будет отсортирован весь массив.

Анализ сложности сортировки вставками:

- **Лучший вариант:** $O(N)$, если список уже отсортирован;
- **Средний вариант:** $O(N^2)$, если список упорядочен случайным образом;

- **Наихудший вариант:** $O(N^2)$, если список находится в обратном порядке, где N - количество элементов в списке.

Algorithm 1 Реализация алгоритма сортировки вставками

```
 $i \leftarrow 2$ 
while  $i \leq n - 1$  do
     $j \leftarrow i - 1$ 
    while  $j \geq 0$  AND  $A[j] > A[j + 1]$  do
         $swap(A[j], A[j + 1])$ 
         $j \leftarrow j - 1$ 
```

Преимущества:

- Эффективно справляется с небольшими и почти отсортированными массивами;
- Экономит место;
- Простой и легкорееализуемый.

Недостатки:

- Неэффективно справляется с большими массивами.

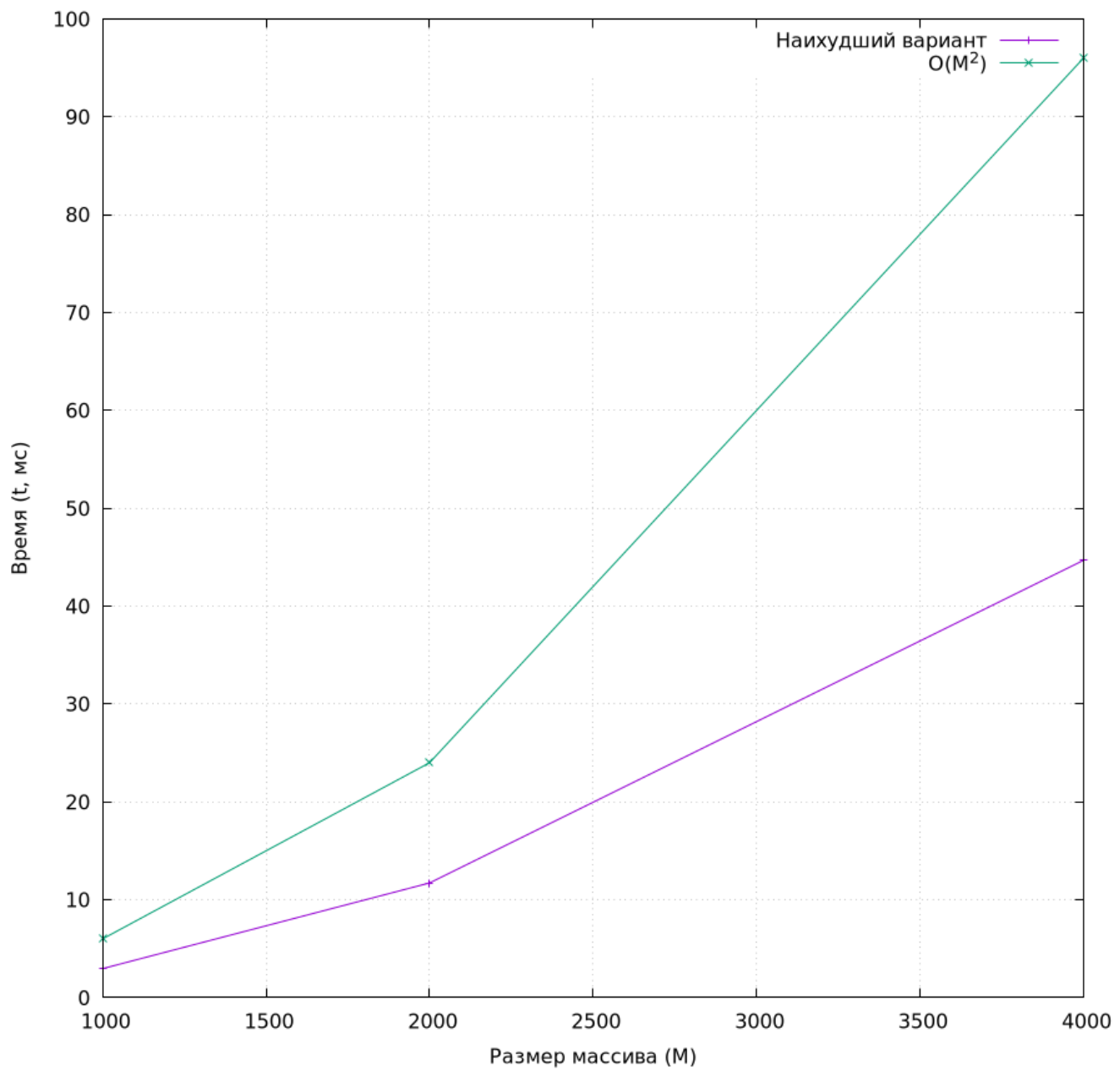
Выполнение задачи

Алгоритм сортировки вставками реализован на языке *C++*. Построение графиков проводить с помощью программы *GNUplot*.

```
1  #include <iostream>
2  int main() {
3      std::cout << "Hello" << std::endl;
4      return 0;
5  }
6
```

Выводы

Расчётные кривые наихудшего времени выполнения сортировки и сложности алгоритма



Расчётные кривые среднего, наилучшего и наихудшего времени исполнения

