

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 №3

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: 10.03.2025

Москва
2025

Оглавление

Описание задачи	1
Описание метода/модели	2
Выполнение задачи	3
Выводы	6

Описание задачи

Написать свою реализацию двусвязного списка:

- добавление элемента в начало, в конец, в произвольное место;
- удаление элемента из списка.

В рамках лабораторной работы необходимо изучить и реализовать двусвязный список. Структура должна:

- использовать шаблонный подход, обеспечивая работу контейнера с произвольными данными;
- реализовывать своё итератор, предоставляющий стандартный для языка механизм работы с ним (для *C++* это операции *++* и *!=*);
- обеспечивать работу стандартных библиотек и конструкции *for each*, если она есть в языке, если их нет, то реализовать собственную функцию, использующую итератор;
- обеспечивать проверку на пустоту и подсчёт количества элементов.

Для демонстрации работы структуры необходимо создать набор тестов (под тестом понимается функция, которая создаёт структуру, проводит операцию или операции над структурой и удаляет структуру):

- заполнение контейнера 1000 целыми числами в диапазоне от -1000 до 1000 и подсчёт их суммы, среднего, минимального и максимального;
- провести проверку работы операций вставки и изъятия элементов на коллекции из 10 строковых элементов;
- заполнение контейнера 100 структур, содержащих фамилию, имя, отчество и дату рождения (от 01.01.1980 до 01.01.2020). Значения каждого поля генерируются случайно из набора заранее заданных. После заполнения необходимо найти всех людей младше 20 лет и старше 30 и создать новые структуры, содержащие результат фильтрации, проверить выполнение на правильность подсчётом количества элементов, не подходящих под условие в новых структурах.

Тесты:

1. перемешать все элементы;
2. выполнить серию тестирования сортировки из первой лабораторной на реализованном списке и сравнить производительность с полученной на массиве.

Описание метода/модели

Двусвязный список - это двунаправленный список, в котором каждый узел имеет два указателя: на следующий и предыдущий узлы, которые ссылаются на следующий и предыдущий узлы соответственно. В отличие от односвязанного списка, в котором каждый узел указывает только на следующий узел, в двусвязном списке есть дополнительный предыдущий указатель, который позволяет перемещаться как вперёд, так и назад.

Каждый узел двусвязного списка состоит из трёх полей:

- *data* - значение, хранящееся в узле;
- *next* - ссылка на следующий узел в списке;
- *prev* - ссылка на предыдущий узел в списке.

Анализ сложности основных операций над двусвязным списком:

- Вставка в начало: $O(1)$;
- Вставка в конец: $O(1)$;
- Вставка в определенный узел: $O(n)$;
- Удаление в начале: $O(1)$;
- Удаление в конце: $O(1)$;
- Удаление в определенном узле: $O(n)$;

Преимущества:

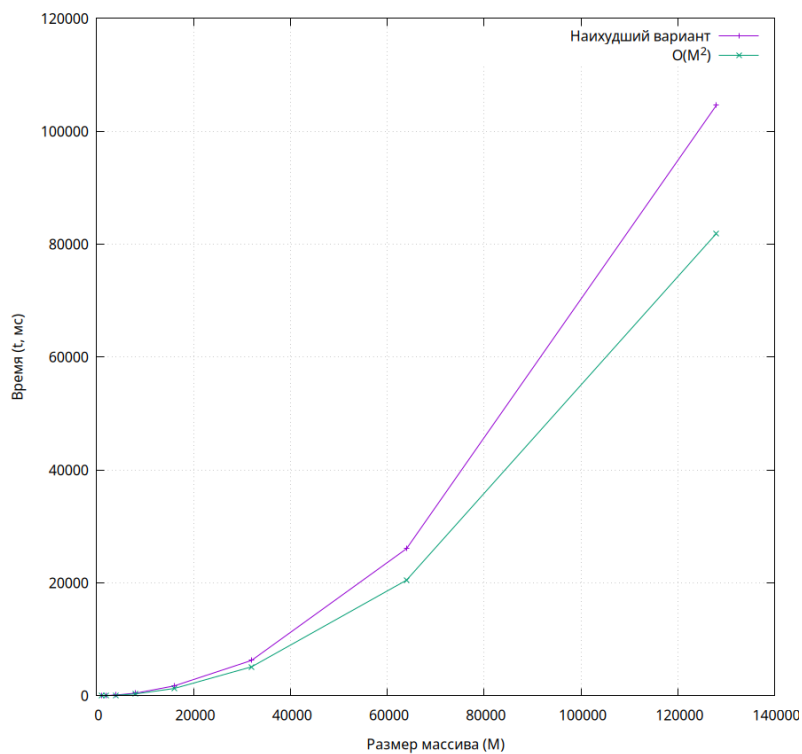
- Позволяет перемещаться как вперёд, так и назад;
- Удаление узла выполняется более эффективно и просто, поскольку у него есть указатель на предыдущий узел;
- Является динамическим по своей природе, поэтому он может увеличиваться и уменьшаться в размерах.

Недостатки:

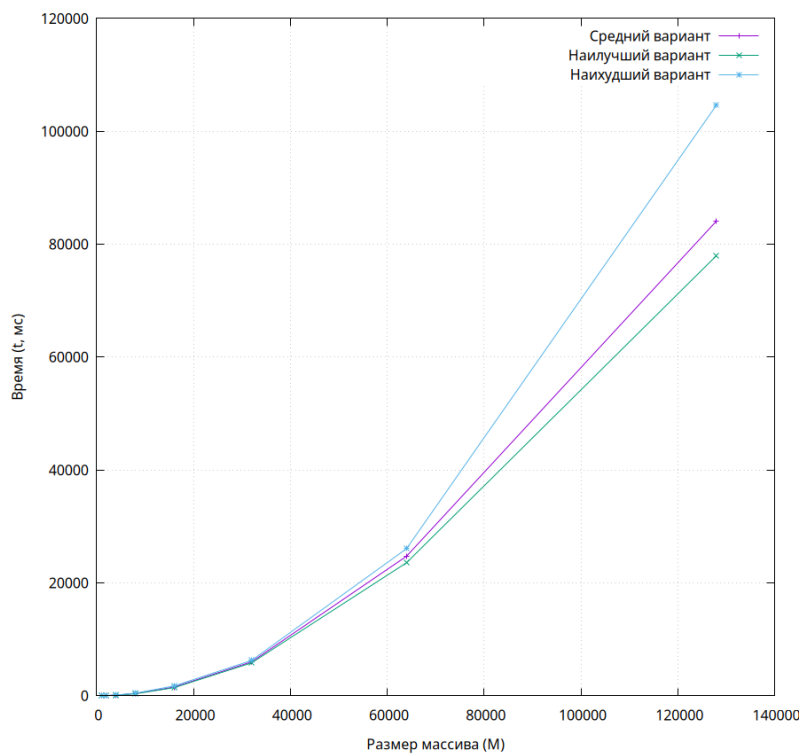
- Для каждого узла требуется больше памяти, чем для массивов, из-за дополнительного хранилища, используемого для указателей;
- Его сложнее реализовать и поддерживать по сравнению с односвязным списком;
- Нужно пройти от головного узла к определённому узлу для вставки и удаления в определённых местах.

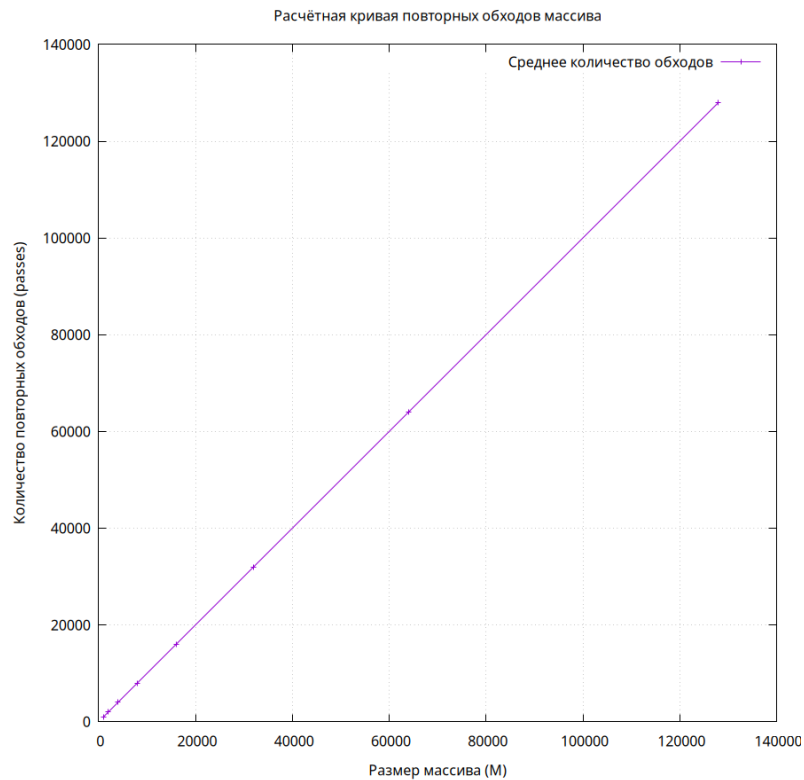
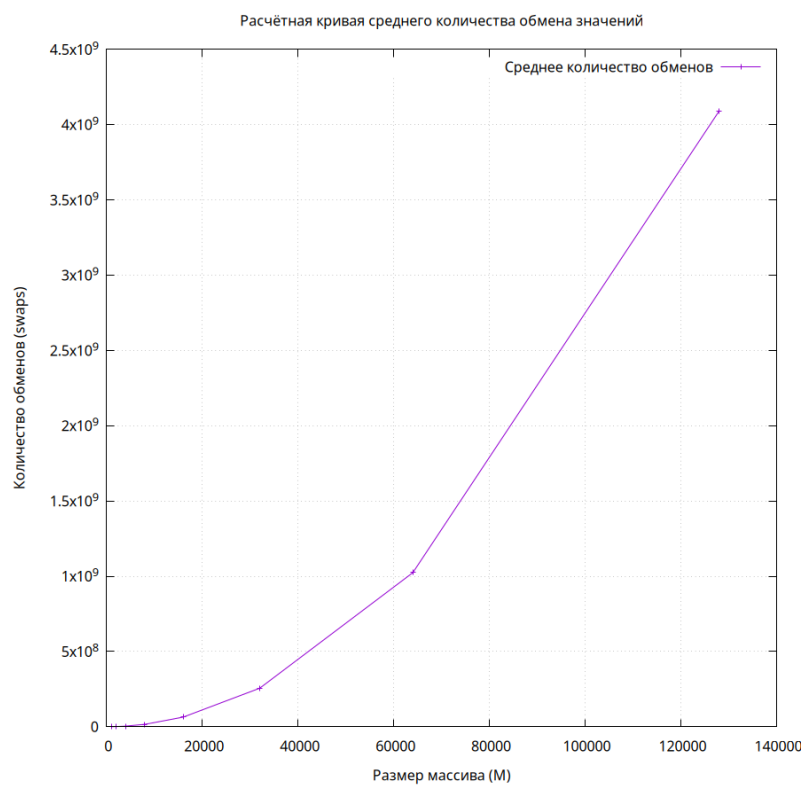
Выполнение задачи

Расчётные кривые наихудшего времени выполнения сортировки и сложности алгоритма



Расчётные кривые среднего, наилучшего и наихудшего времени исполнения





Выводы