

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

## **ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №7**

Выполнил студент группы.....КС-36 Жижин Алексей Павлович

Ссылка на репозиторий: [https://github.com/MUCTR-IKT-CPP/APZhizhin\\_36\\_2SEM](https://github.com/MUCTR-IKT-CPP/APZhizhin_36_2SEM)

Приняли: .....Пысин Максим Дмитриевич  
.....Краснов Дмитрий Олегович

Дата сдачи: 25.04.2022

---

### **Оглавление**

Описание задачи.....	2
Описание метода/модели.....	2
Заключение. ....	4

## Описание задачи.

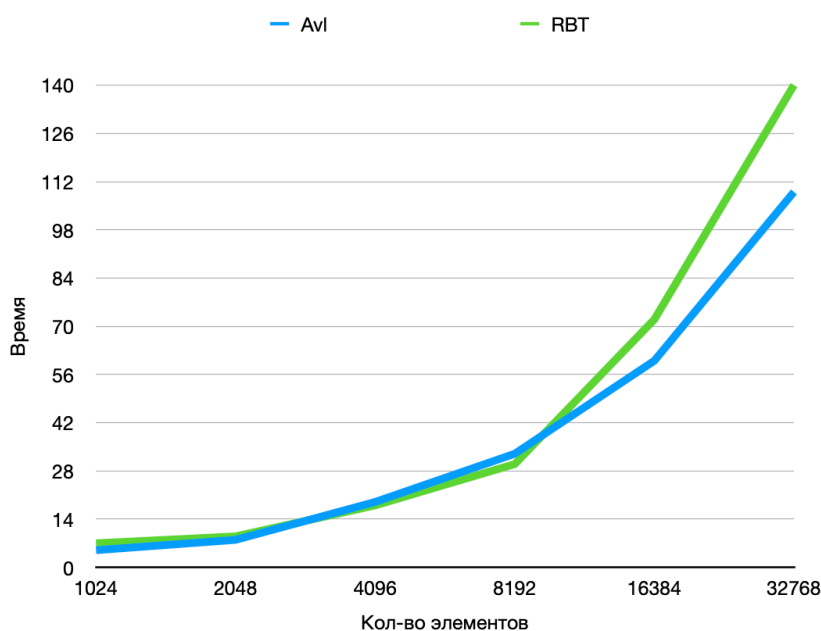
Реализация рандомизированного дерева и сравнение с avl-деревом

## Описание метода/модели.

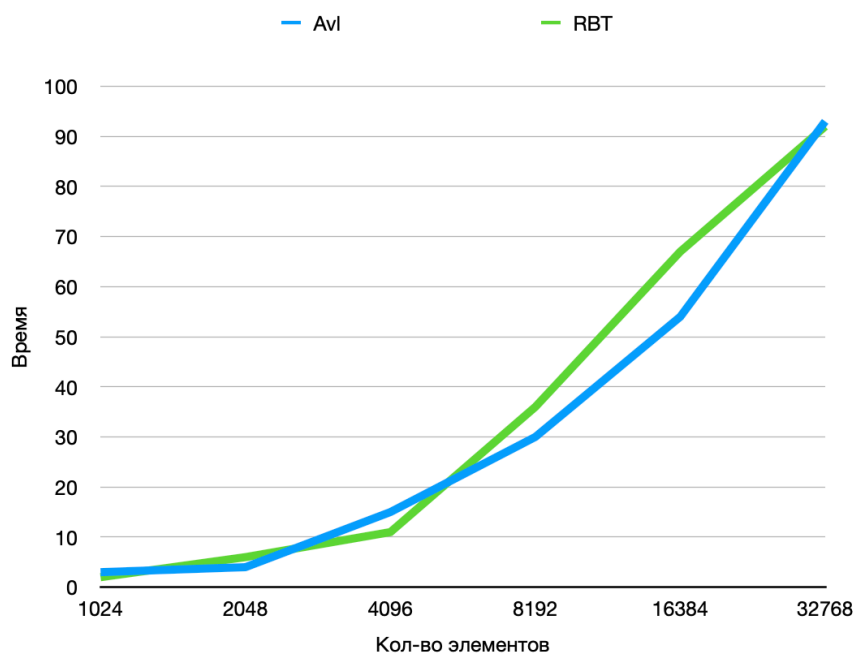
Рандомизированное дерево- бинарное дерево, в котором нет отдельного блока балансировки. В этом дереве происходит генерация вероятности, того будет ли вставлен элемент в текущий узел или берем следующий элемент(правый или левый). Т.е алгоритм подразумевает, что на каждой вставке, наше дерево будет генерировать случайную вероятность и менять (при нужном результате) текущий корневой узел. В данном дереве алгоритмическая сложность лежит в пределе от  $O(\log N)$  до  $O(N)$ , в данном случае алгоритмическая сложность зависит от того, какая вероятность у нас выпадет. Конечно сложность  $O(N)$  мало вероятна, но сказать, что сложность  $O(\log N)$  присутствует всегда, тоже нельзя.

АВЛ-дерево — это прежде всего двоичное дерево поиска, ключи которого удовлетворяют стандартному свойству: ключ любого узла дерева не меньше любого ключа в левом под дереве данного узла и не больше любого ключа в правом под дереве этого узла. Это значит, что для поиска нужного ключа в АВЛ-дереве можно использовать стандартный алгоритм. Алгоритмическая сложность  $O(\log N)$

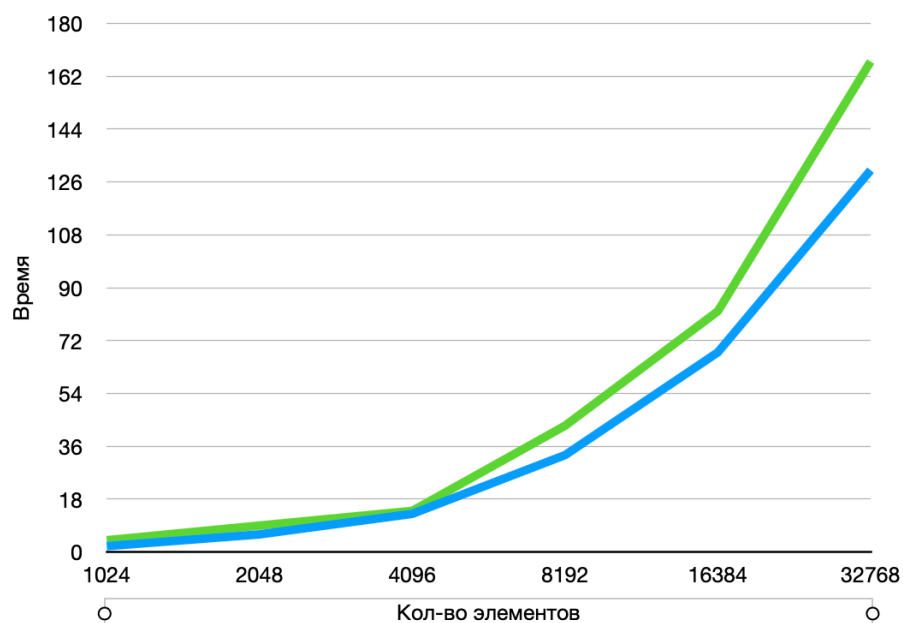
Эффективность работы деревьев показывают тесты:



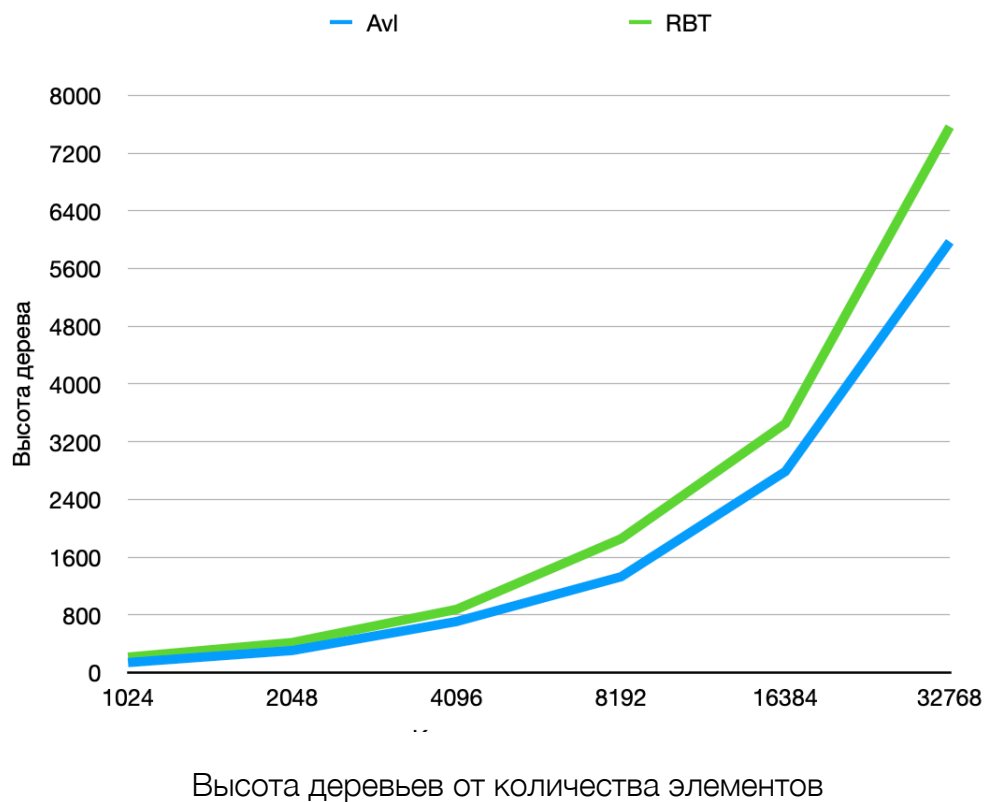
Среднее время вставки элемента в деревья



Среднее время удаления элемента из деревьев



Среднее время поиска элемента в деревьях



### Заключение.

Проводя сравнение между avl- деревом и рандомизированным бинарным деревом, я пришел к выводу, что avl- дерево лучше. Во-первых, у него есть балансировка, которая гарантирует поиск элементов за  $O(\log N)$ , что дает выигрыш в соревновании. Так как рандомизированное дерево выдает случайность, это не дает точной оценки сложности, поэтому мы равняемся на относительный результат от  $O(\log N)$  до  $O(N)$ . Уо у рандомизированного действия нет дополнительного блока проверки, что положительно складывается на выделении ресурсов(их требуется меньше).