**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра САПР**

**ОТЧЁТ**

**по лабораторной работе №1**

**по дисциплине «Алгоритмы и структуры данных»**

**Тема: Самобалансирующиеся двоичные деревья**

Студент гр. 0322 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Руссу В.А.

Преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Пестерев Д.О.

Санкт-Петербург

2023

# Задача

Реализовать двоичное дерево поиска, красно-черное дерево и АВЛ-дерево. Сравнить длины деревьев на случайном наборе входных данных, распределенных равномерно и экспоненциально. Сравнить временные затраты на балансировку для красно-черного и АВЛ-дерева. Отчет должен содержать графики.

# Описание реализуемых классов

RedBlackTree – в данном классе реализованы основные методы для красно-черного дерева (вставка, поиск, удаление, вывод и подсчет высоты дерева).

NodeBST – данный класс содержит в себе поля, которые хранят информацию о значении и двух указателях на левое и правое поддерево. Для данного дерева реализованы методы вставки, удаления и подсчет высоты.

NodeAVL - данный класс содержит в себе поля, которые хранят информацию о значении, высоте дерева и двух указателях на левое и правое поддерево. Для данного дерева реализованы методы вставки, удаления, вывода, а также вспомогательные функции для балансировки дерева.

# Сравнение длин деревьев

Сравним длины деревьев на случайном наборе входных данных, распределенных равномерно и экспоненциально.

График при равномерном распределении входных данных (Рисунок 1).

Координата x – количество введённых данных, координата y – длина дерева.

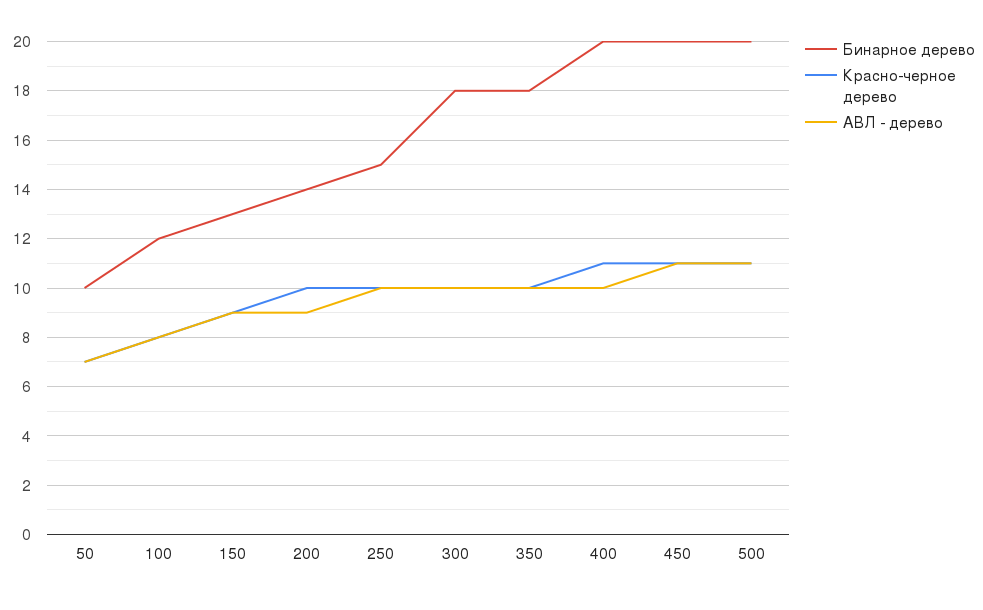


Рисунок 1

График при экспоненциальном распределении входных данных (Рисунок 2).

Координата x – количество введённых данных, координата y – длина дерева.

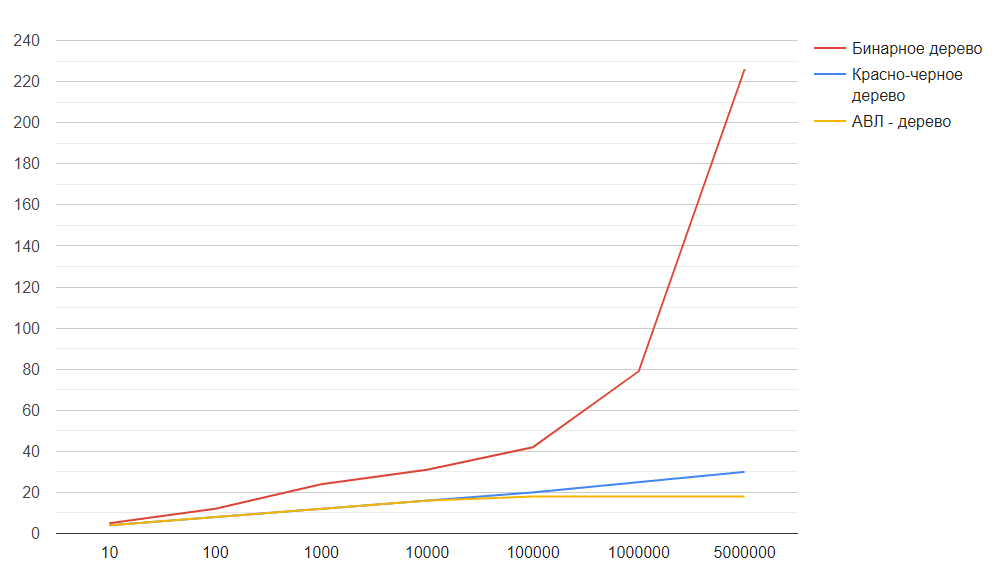


Рисунок 2

# Сравнение временных затрат на балансировку

На рисунке 3 представлен график временных затрат на балансировку по отношению к количеству веденных данных. На рисунке 4 показан пример работы программы.

Координата x – количество введённых данных, координата y – время, выраженное в наносекундах.  
По графику можно заметить, что AVL дерево тратит больше времени на балансировку.

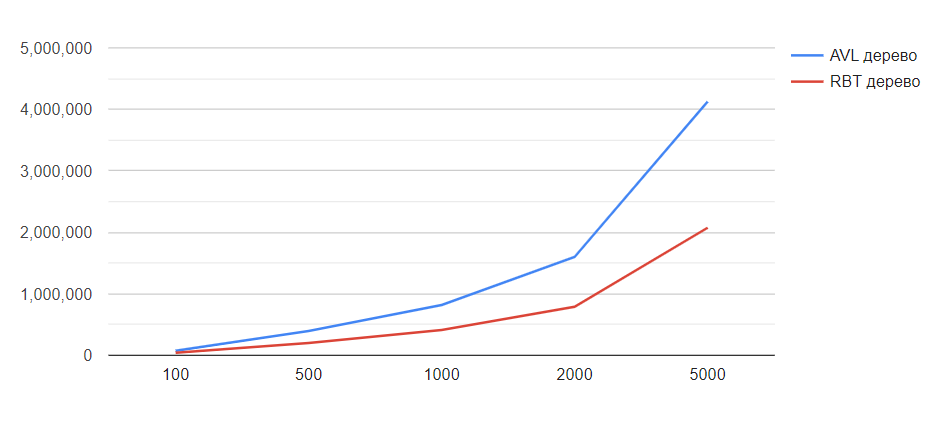


Рисунок 3

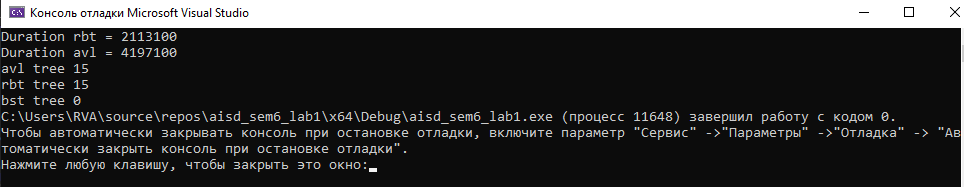


Рисунок 4

# Заключение

В ходе выполнения данной лабораторной работы были реализованы три дерева: двоичное дерево поиска, красно-черное дерево и АВЛ-дерево.

Ссылка   
https://github.com/RVA-t/aisd\_sem6\_lab1