
Разработка и реализация алгоритма
создания и балансировки двоичного дерева
поиска со взвешенными узлами

Подготовила Колосова Ирина, гр. 4940

Цель работы

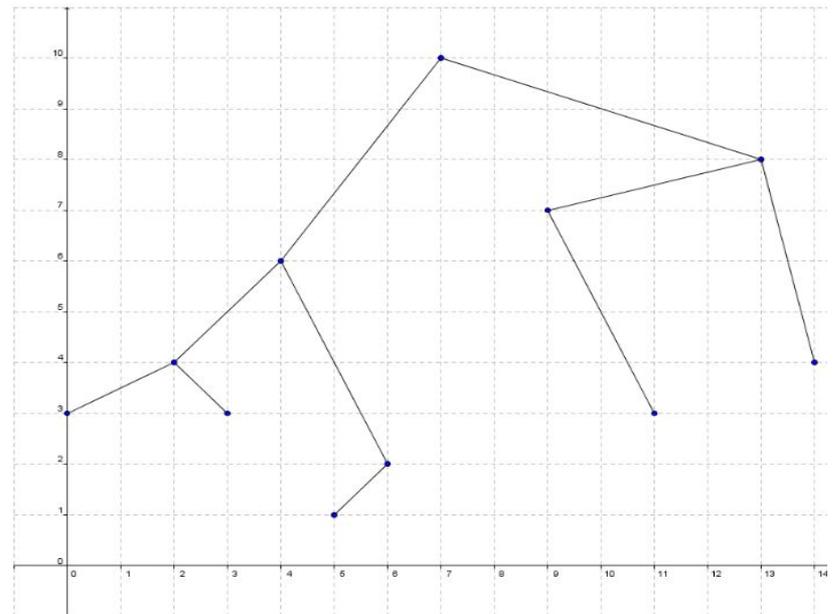
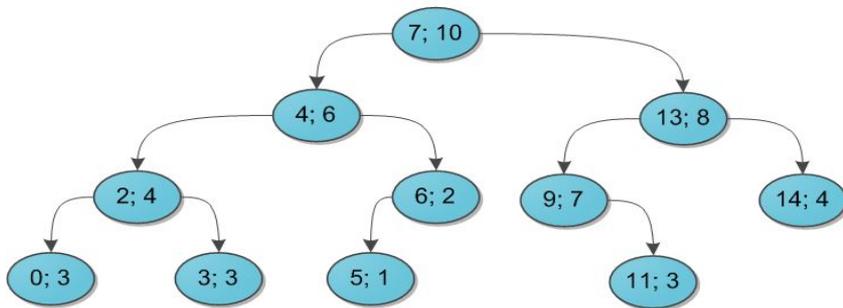
- Разработка структуры:
 - минимальные затраты памяти;
 - быстрый поиск;
 - приоритезированный доступ.
 - Реализация структуры.
 - Анализ эффективности.
 - Визуализация.
-

Балансировка двоичных деревьев поиска

- По высоте
 - По весу
 - По количеству узлов
-

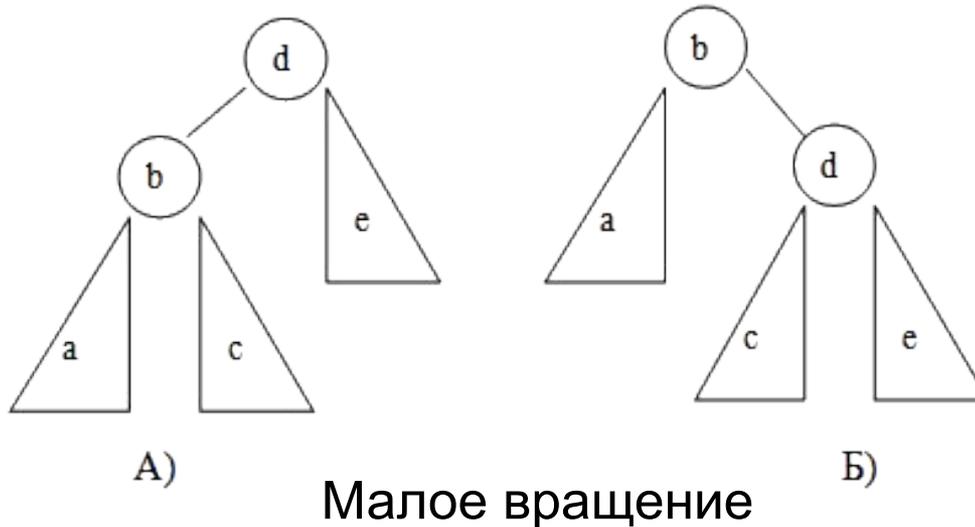
Декартово дерево (Treap)

Декартово дерево - хранит пары (X, Y) в виде бинарного дерева таким образом, что оно является деревом поиска по X и кучей по Y .



Структура данных ТКОЛ

Разработанная структура, названная ТКОЛ – двоичное дерево поиска, балансируемое по весу.



- А) Структура дерева до вращения
Б) Структура дерева после вращения

Критерий вращения:

$$|P_a + P_b + P_c - P_e| > |P_c + P_d + P_e - P_a| ,$$

где P_i – вес i -го узла или поддеревя.

Теоретический анализ

Количество переходов по дереву до случайного элемента составляет

$$T(N, P_{left}) = \frac{P_{left}}{P} T(N_{left}, P_{left}) + \left(1 - \frac{P_{left}}{P}\right) T(N - N_{left} - 1, P_{left})$$

где:

P_{left} - вес левого поддерева;

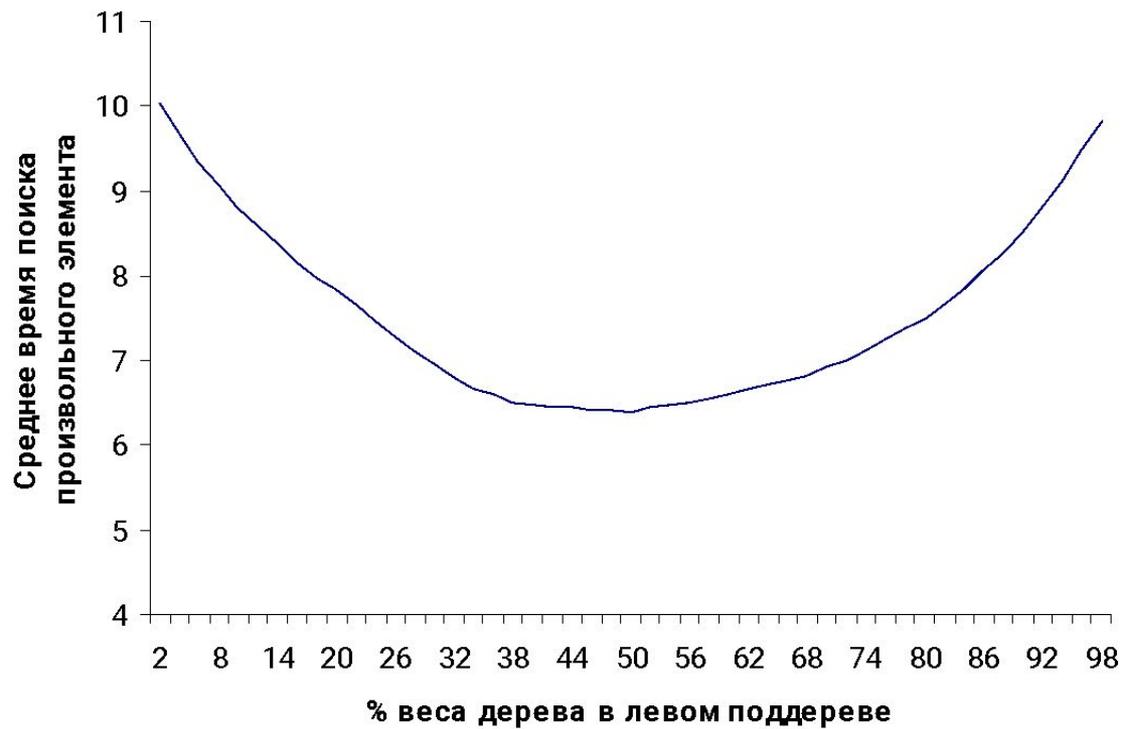
P - вес всего дерева;

N_{left} - количество узлов в левом поддерева;

N - суммарное количество узлов дерева.

Теоретический анализ

Влияние весовой сбалансированности дерева на среднее время поиска



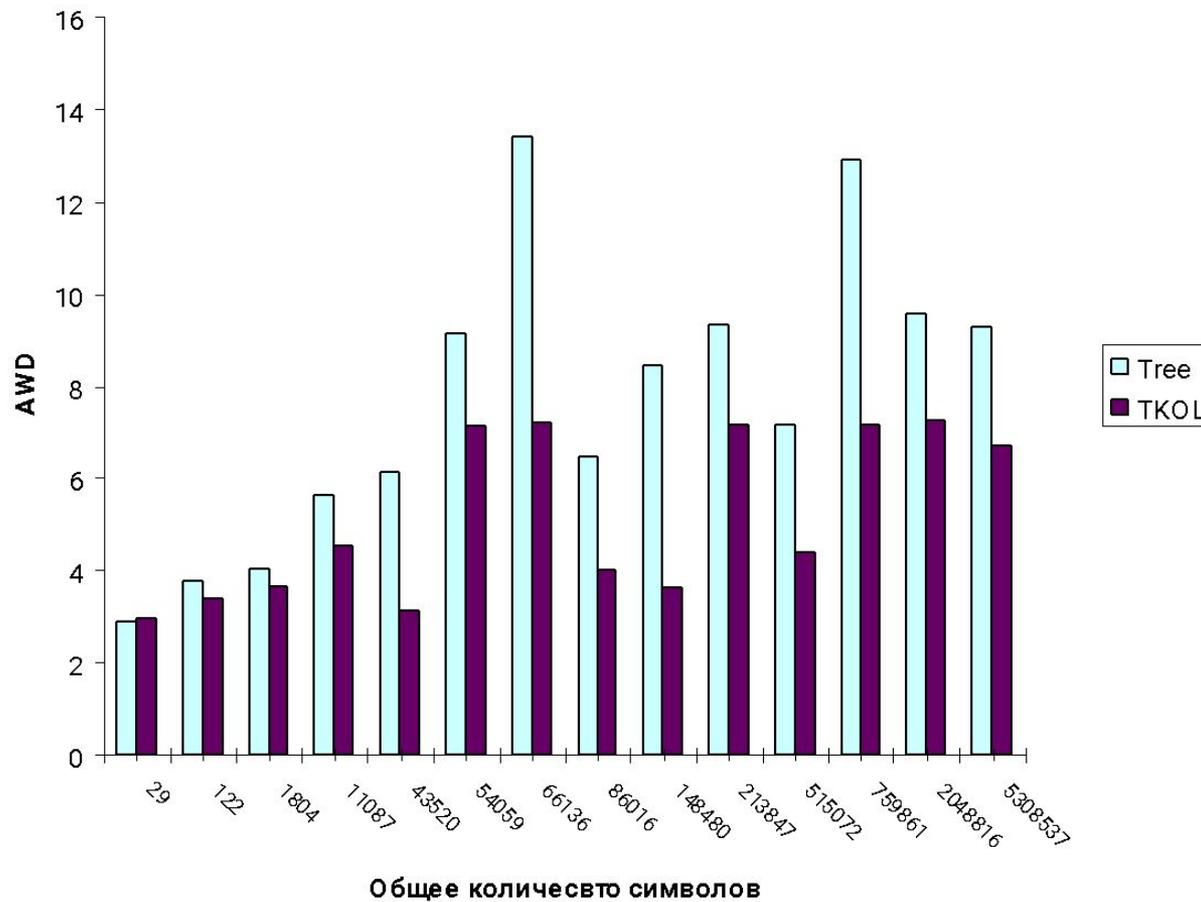
Средневзвешенный путь

$$AWD = \frac{\sum P_i \cdot d}{\sum P_i}$$

где P_i — собственный вес узла;

d — длина пути от корня до узла,
увеличенная на единицу.

Сравнение эффективности структуры TKOL и несбалансированного BST



Сравнение эффективности структуры ТКОЛ и декартового дерева

