

УДК 004.05

Мишин Артём Дмитриевич, студент,
Пензенский государственный университет
Mishin Artem Dmitrievich,
student, Penza State University

**АНАЛИЗ ЭФФЕКТИВНОСТИ ПОИСКА В СБАЛАНСИРОВАННЫХ
И НЕСБАЛАНСИРОВАННЫХ БИНАРНЫХ ДЕРЕВЬЯХ
ANALYSIS OF SEARCH EFFICIENCY IN BALANCED
AND UNBALANCED BINARY TREES**

Аннотация. В статье приводится сравнительный анализ производительности поиска в классическом бинарном дереве (BST) и AVL-дереве. Целью анализа является выявление зависимости времени поиска и высоты структуры от степени упорядоченности входных данных, что является критическим показателем эффективности алгоритмов обработки данных для древовидных структур.

Abstract. The article provides a comparative analysis of search performance in a classical binary tree (BST) and an AVL tree. The purpose of the analysis is to identify the dependence of the search time and the height of the structure on the degree of ordering of the input data, which is a critical indicator of the effectiveness of data processing algorithms for tree structures.

Ключевые слова: Бинарное дерево поиска, AVL-дерево, балансировка, высота дерева, временная сложность.

Keywords: Binary search tree, AVL tree, balancing, tree height, time complexity.

В современных информационных системах технология эффективного поиска данных покрывает широкий спектр задач – от индексации баз данных до реализации ассоциативных массивов. Фундаментальная трудность заключается в том, что при определенных условиях стандартные структуры данных могут деградировать, что сказывается на потере эффективности операции поиска.

Для бинарных деревьев поиска такой проблемой является потеря баланса, когда дерево превращается в линейный список. В этом случае высота дерева становится равной количеству элементов, что приводит к линейной сложности поиска $O(n)$ вместо логарифмической $O(\log n)$. Напротив, для самобалансирующихся AVL-деревьев эта проблема решается при каждой вставке нового элемента с помощью вращений, благодаря чему высота дерева всегда остается пропорциональной логарифму от количества узлов; этот процесс называется балансировкой [1, 2].

Именно высота дерева является фундаментальной характеристикой, определяющей эффективность поиска, поскольку максимальное количество сравнений при поиске элемента напрямую равно высоте дерева. Отсюда следует, что анализ эффективности поиска в деревьях различной степени упорядоченности является удобным инструментом для определения применимости конкретных алгоритмов в инженерной практике.

Данная статья посвящена анализу эффективности поиска в древовидных структурах данных и выявлению зависимости между высотой дерева, степенью упорядоченностью и эффективностью поиска.

Для проведения анализа были использованы следующие исходные данные: массивы ключей, размерность которых менялась от $N = 10000$ до $10N$. Значения элементов деревьев совпадали. Были рассмотрены два возможных случая: когда массивы заранее отсортированы в возрастающем порядке и когда не отсортированы. Исследовались два типа структур:



стандартное бинарное дерево поиска (BST) и самобалансирующееся AVL-дерево (AVL). Основными критерием эффективности выступали время поиска элемента, количество сравнений и высота дерева.

Для проведения анализа было разработано приложение на объектно-ориентированном языке программирования C# с использованием графического представления Windows Forms [3].

На рисунке 1 представлен анализ производительности поиска для деревьев, состоящих из 10000 элементов.

Неотсортированные деревья								
BST	Время:	7974	Найдено:	False	Сравнений:	14	Высота:	28
AVL	Время:	6565	Найдено:	False	Сравнений:	13	Высота:	16

Отсортированные деревья								
BST	Время:	3958	Найдено:	False	Сравнений:	10000	Высота:	10000
AVL	Время:	18	Найдено:	False	Сравнений:	14	Высота:	14

Рисунок 1. Анализ производительности поиска для деревьев из 10000 элементов

В ходе анализа было установлено, что высота классического бинарного дерева критически зависит от упорядоченности входных данных: когда на вход поступают отсортированные данные, дерево вырождается в линейный однонаправленный список. В таком случае поиск элементов крайне неэффективен и в худшем случае имеет сложность $O(n)$. Таким образом, для эффективного поиска в классическом бинарном дереве необходимо заранее учесть возможность вырождения в список, однако на практике это не всегда возможно предсказать, особенно в системах с непредсказуемым потоком входных данных. Следовательно, применение классического BST без дополнительных механизмов балансировки и эффективных алгоритмов поиска может быть рискованным.

AVL-дерево в 100% случаев показало лучший результат, чем классическое бинарное дерево. При этом результаты для отсортированного AVL-дерева лучше, чем для неотсортированного. Это объясняется тем, что в процессе вставки отсортированных данных балансировка выполняется на каждом шаге, формируя дерево с оптимальной высотой, близкой к теоретическому минимуму. В случае же неотсортированных данных дерево также остаётся сбалансированным, но его высота может быть незначительно выше из-за случайного характера распределения элементов

Таким образом, практическим путём было установлено, что AVL-деревья являются наиболее подходящей древовидной структурой с возможностью эффективного поиска для любой степени упорядоченности входных данных с незначительными корреляциями. Также было показано, что высота дерева является определяющим фактором эффективности поиска: чем меньше высота, тем меньше требуется времени и сравнений для нахождения элементов.



Список литературы:

1. Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы построения и анализ данных. / Пер. с англ. канд. техн. наук И.В. Красикова, Н.А. Ореховой, В.Н. Романова – изд. «Вильямс», 2011 – 1296 с.
2. Вирт Н. Алгоритмы и структуры данных. / Пер. с англ. Ткачев Ф. В. – М.: ДМК Пресс, 2010.– 272 с.: ил.
3. Васильева А. Н. Программирование на C# для начинающих. / Васильева А. Н. – Москва: изд-во Эксмо, 2022. – 592 с.

