

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по лабораторной работе №5
по дисциплине «Алгоритмы и структуры данных»
Тема: Деревья поиска

Студент гр. 9303

Ефимов М.Ю

Преподаватель

Филатов Ар.Ю

Санкт-Петербург

2020

Описание алгоритма

Вариант 8.

БДП: случайное* БДП с рандомизированной вставкой; действие: 1+2а

1) По заданной последовательности элементов Elem построить структуру данных определённого типа – БДП или хеш-таблицу;

2а) Для построенной структуры данных проверить, входит ли в неё элемент e типа Elem, и если входит, то добавить элемент e в структуру данных. Предусмотреть возможность повторного выполнения с другим элементом.

Основные теоретические положения.

Как известно, можно подобрать такую последовательность операций с бинарным деревом поиска в наивной реализации, что его глубина будет пропорциональна количеству ключей, а следовательно операции будут выполняться за $O(n)$. Поэтому, если поддерживать инвариант "случайности" в дереве, то можно добиться того, что математическое ожидание глубины дерева будет небольшим. Дадим рекурсивное определение рандомизированного бинарного дерева поиска (RBST).

Определение:

Пусть T — бинарное дерево поиска. Тогда

- Если T пусто, то оно является рандомизированным бинарным деревом поиска.
- Если T непусто (содержит n вершин, $n > 0$), то T — рандомизированное бинарное дерево поиска тогда и только тогда, когда его левое и правое поддеревья (L и R) оба являются RBST, а также выполняется соотношение $P[\text{size}(L)=i]=1/n, i=1..n$.

Из определения следует, что каждый ключ в RBST размера n может оказаться корнем с вероятностью $1/n$.

Идея RBST состоит в том, что хранимое дерево постоянно является рандомизированным бинарным деревом поиска. Далее подробно будет описана реализация операций над RBST, которая позволит добиться этой цели. Заметим лишь, что хранение RBST в памяти ничем не отличается от хранения обычного дерева поиска: хранится указатель на корень; в каждой вершине хранятся указатели на её сыновей.

Похожие идеи используются в декартовом дереве, поэтому во многих русскоязычных ресурсах термин рандомизированное бинарное дерево поиска используется как синонимическое название декартового дерева и декартового дерева по неявному ключу.

Выполнение работы.

В ходе выполнения работы была написана шаблонная структура Tree содержащая поля key для хранения данных, size – количество потомков дерева и left, right – ссылки на потомков. Была реализованная функция insert и insertroot для вставки и вставки в корень. Были реализованы функции для поворота left rotateright и rotateleft. Были Считывание может происходить как с клавиатуры так и с консоли. Конечный результат и промежуточные значения выводятся на консоль и дублируются в файл. Пример входных данных и результат работы приведен в файлах input.txt и result.txt.

Выводы.

В ходе выполнения работы была изучена структура данных типа случайное бинарное дерево поиска со случайной вставкой . Написана программа, в которой была реализована данная структура данных , и написаны функции для работы с ней.

