

Лабораторная работа №2 по курсу дискретного анализа: Сбалансированные деревья

Выполнил студент группы М8О-208Б-20 Борисов Ян.

Условие

1. Необходимо создать программную библиотеку, реализующую указанную структуру данных, на основе которой разработать программу-словарь. В словаре каждому ключу, представляющему из себя регистронезависимую последовательность букв английского алфавита длиной не более 256 символов, поставлен в соответствие некоторый номер, от 0 до $2^{64}-1$. Разным словам может быть поставлен в соответствие один и тот же номер.
2. Вариант задания: В-дерево.
3. Операции: вставка, поиск, удаление элемента, сохранение словаря в файл, загрузка словаря из файла.

Метод решения

Реализовал классы для элемента (TItem), узла дерева (TNode) и В-дерева (TBtree). При этом для реализации В-дерева используются методы класса узла, а в узле хранятся объекты класса элемента.

Описание программы

Программа состоит из одного файла, в котором написана реализация трех классов и функция main. В данных классах реализованы функции для вставки пары в дерево, поиска, удаления, сохранения дерева в файл и загрузки дерева из файла.

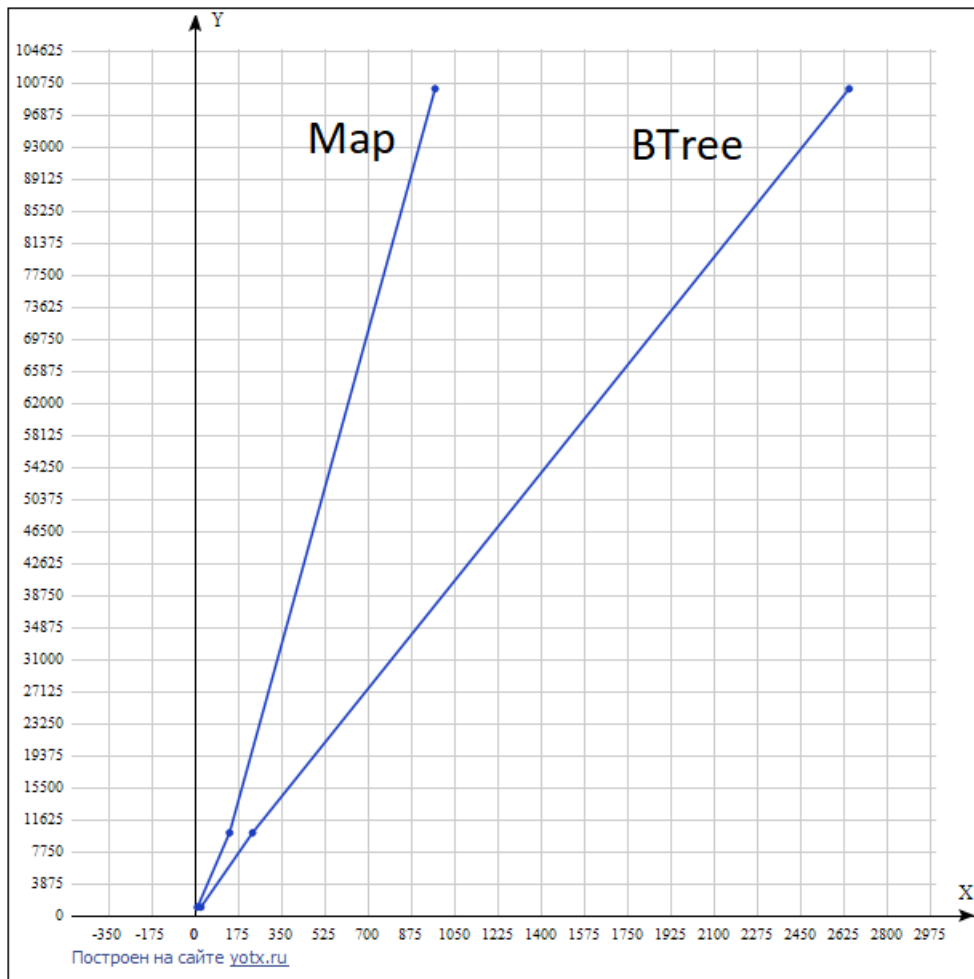
Дневник отладки

1. Сначала я загрузил программу на чекер без Save и Load. Программа прошла 7 тестов.
2. Затем я реализовал Save, Load, но при загрузке на чекер программа прошла только до 13 теста.
3. Выяснилось, что в функциях Save и Load была допущена ошибка при записи узлов, проблема устранена.

Тест производительности

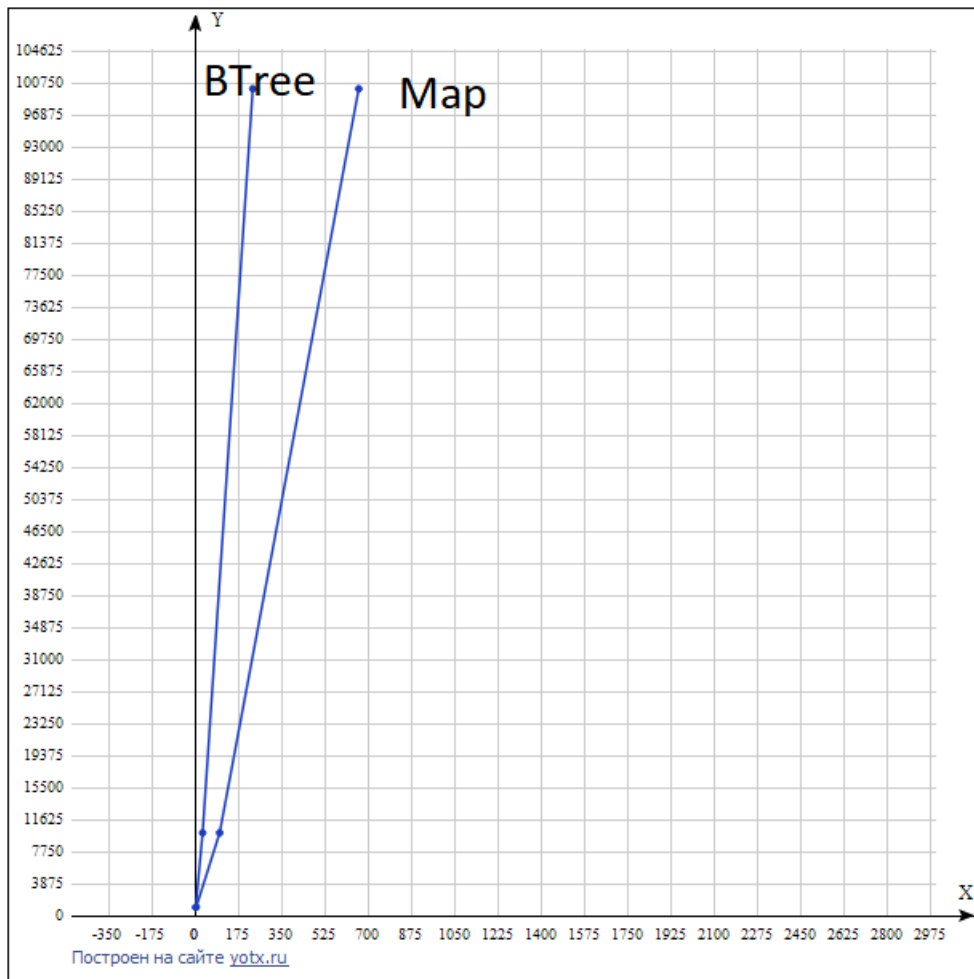
Во всех графиках по оси Y отложено время выполнения (в миллисекундах), по оси X – количество команд.

Вставка



Кол-во команд	std::map	TBTree
1000	10	24
10000	140	233
100000	972	2648

Поиск



Кол-во команд	std::map	TBTree
1000	5	3
10000	100	31
100000	663	234

Удаление



Кол-во команд	std::map	TBTree
1000	4	11
10000	76	130
100000	525	1000

Из графиков можно сделать вывод, что операция поиска выполняется в B-дереве быстрее чем в Map. При этом вставка и удаление немного медленнее.

Недочёты

К сожалению, пришлось совместить всю разбитую по файлам структуру программы в один файл, чтобы получить ОК на Яндекс Контест, из-за этого структура программы

стала менее читаемой и понятной.

Выводы

В-дерево – очень полезная структура для хранения данных, так как операции вставки, поиска и удаления элементов в среднем работают быстрее, чем в несбалансированном бинарном дереве поиска.

Структура В-дерева применяется для организации индексов во многих современных СУБД.

Однако данные операции являются сложными для реализации (в частности операция удаления), поэтому стоит соизмерять пользу от использования данного дерева с временем, затраченным на его реализацию.