

Лабораторная работа № 2 по курсу дискретного анализа: Словарь

Выполнил студент группы М8О-208Б МАИ *Захаров Игорь*.

Условие

Кратко описывается задача:

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 264 1. Разным словам может быть поставлен в соответствие один и тот же номер.
2. Структура данных: PATRICIA.

Метод решения

В данной лабораторной работе мне нужно разработать словарь на предложенной структуре данных – PATRICIA. PATRICIA – одно из видов trie-деревьев, придуманное с целью ликвидации недостатков обычного trie-дерева: создание дополнительных узлов из-за однонаправленного ветвления и хранение двух различных типов узлов, которое приводит к усложнениям при работе с деревом. В PATRICIA мы избегаем однонаправленного ветвления благодаря тому, что в каждом узле содержится бит, который должен проверяться с целью выбора пути из этого узла. Внешние узлы и ключаются при помощи того, что данные хранятся во внутренних узлах, а связи с внешними узлами заменяются связями, которые указывают в обратном направлении вверх на требуемый узел. Эти два изменения позволяют представлять trie-деревья как бинарные деревья, состоящие из узлов с ключом и двумя связями, а также дополнительным полем под бит. PATRICIA позволяет выполнять поиск N ключей в дереве, содержащем всего N узлов. Для выполнения одного поиска требуется всего лишь около $\lg N$ сравнений рядов и одного сравнения полного ключа.

Данное дерево решено было реализовывать на основе двух структур `TPatriciaTrieNode` и `TPatriciaTrie`. В первой хранится только лишь корень дерева и количество элементов в нем, во второй – ключ, значение, бит различия, два указателя на следующие узлы.

Дневник отладки

По ходу выполнения лабораторной работы столкнулся с проблемами: Было множество проблем со структурой, памятью, динамикой, ссылками и тд.

Выводы

PATRICIA удобна тем, что позволяет идентифицировать разряды, которые отличают ключи, после чего встраивать их в структуру данных без избыточных узлов. Это обеспечивает быстрое попадание от любого искомого ключа к единственному ключу в структуре, который мог бы быть равен искомому. Если сравнивать PATRICIA-деревья со стандартными trie-деревьями, то первые содержат в два раза меньше узлов и являются почти идеально сбалансированными. Вставка и поиск случайного ключа в дереве требует в среднем приблизительно $\lg N$ битовых сравнений. Количество сравнений никогда не превысит длины ключа. Для PATRICIA не существует никаких ограничений на длину искомого ключа, к тому же оно также эффективно и для ключей переменной длины. Сложность поиска, вставки и удаления в PATRICIA составляет $O(h)$, где h – высота дерева. Сложности данных операций в моей программе близки к этому. В открытых источниках информации о PATRICIA представлено не так много, как, например, о структурах других вариантов данной лабораторной работы или о классическом trie-дереве. Мне показалось это странным, возможно это не самый удобный или не самый эффективный вариант реализации префиксных деревьев. В любом случае, идея, заложенная в основе PATRICIA-деревьев, показалась мне необычной и было приятно ознакомиться с ней глубже. Во время поиска материала я узнал о такой интересной вариации PATRICIA-дерева как Merkle Patricia Trie, которую использовали при создании Эфириума.