

Лабораторная работа № 2 по курсу дискретного анализа: Словарь

Выполнил студент группы 08-207 МАИ *Лебедев Иван*.

Условие

1. Необходимо создать программную библиотеку, реализующую указанную структуру данных, на основе которой разработать программу-словарь. В словаре каждому ключу, представляющему из себя регистро-независимую 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. AVL-дерево.

Метод решения

В цикле считываются данные, проверяется первый символ, если он равен «+», то считывается пара «ключ значение» и вставляется в дерево. Если такой ключ уже существует выводится «Exist». При «-» считывается ключ и удаляется из дерева, если такого ключа в дереве нет, выводится «NoSuchWord». При «!» считывается следующее слово, если это «Save», то считывается имя файла и происходит сериализация существующего дерева. Если «Load» то считывается имя файла, создается новое дерево и происходит десериализация данных из файла. При неудаче работа продолжается с исходным деревом. Если первый символ введенного слова не совпадает ни с одним из вышеперечисленных, то выполняется поиск этого слова в дереве.

Описание программы

1. *Заголовочные файлы:*
TNode.h(Объявление узла AVL-дерева)
TAVLtree.h(Объявление функций AVL-дерева).
2. *Cpp файлы:*
main.cpp(Считывание и вывод данных, управление программой)
TAVLtree.cpp (Реализация функций дерева)
TNode.cpp(Реализация функций узла).

Дневник отладки

Программа не проходила тест из-за того, что в некоторых случаях строка не переводилась в нижний регистр.

Выводы

AVL-дерево — сбалансированное по высоте двоичное дерево поиска. В отличие от обычного бинарного дерева поиска, в AVL-дереве все операции в наихудшем случае выполняются за $\log n$. в тоже время на достаточно простое для понимания и реализации.