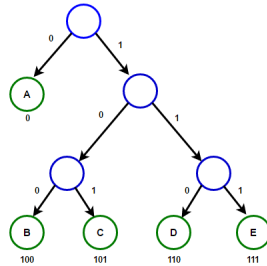


Описание структуры дерева кодирования

Введение

Дерево кодирования представляет собой бинарное дерево Хаффмана. Оно подчиняется префиксному правилу — никакое кодовое слово не является началом другого. Это позволяет однозначно декодировать последовательность кодовых слов. Каждый лист дерева является некоторым уникальным символом (байтом), которому соответствует некоторая бинарная последовательность, составленная из пути от корня дерева к этому листу. Все левые ветви обозначены нулем, а правые — единицей. Чем чаще попадаетея символ, тем короче его бинарное представление.



Структура класса Node

Класс Node хранится в файле `src/Node.java`. Он представляет собой узлы дерева, не являющиеся листьями. Реализует интерфейс `Comparable`, позволяя сравнивать узлы между собой. Имеет ссылки на левого и правого детей (которые всегда существуют), а также целочисленное поле `weight` — суммарный вес поддерева.

Структура класса Leaf

Класс Leaf хранится в файле `src/Leaf.java`. Он наследуется от класса Node, и хранит `symbol` — символ (байт) и `prefix` — его бинарное представление.

Построение дерева

Дерево строится в методе `getTree()` класса `Haffman`, используя очередь с приоритетом, уже заполненной листьями (экземплярами класса Leaf), где узлы сравниваются по своим весам (частотности в тексте). Дерево строится до тех пор, пока не опустошиться вся очередь.