

前缀树原理和代码详解

前置知识：

知道什么是树结构，比如二叉树。讲解008-数据结构分类，讲解017-二叉树基本概念

知道为什么推荐静态数组的方式实现各种结构。讲解019-处理输入和输出-推荐静态空间的实现

知道哈希表怎么用。讲解026-哈希表的使用

前缀树又叫字典树，英文名*trie*：

每个样本 都从头节点开始 根据 前缀字符或者前缀数字 建出来的一棵大树，就是前缀树

没有路就新建节点；已经有路了，就复用节点

前缀树的使用场景：需要根据前缀信息来查询的场景

前缀树的优点：根据前缀信息选择树上的分支，可以节省大量的时间

前缀树的缺点：比较浪费空间，和总字符数量有关，字符的种类有关

前缀树的定制：*pass*、*end*等信息

这节课是前缀树的原理和代码讲解。下节课是前缀树相关题目讲解，展示前缀树在解题时的常见用法。

前缀树原理和代码详解

实现前缀树 *Trie* 类：

1. *Trie()* 初始化前缀树对象。
2. *void insert(String word)* 将字符串 *word* 插入前缀树中。
3. *int search(String word)* 返回前缀树中字符串 *word* 的实例个数。
4. *int prefixNumber(String prefix)* 返回前缀树中以 *prefix* 为前缀的字符串个数
5. *void delete(String word)* 从前缀树中移除字符串 *word* 。

分别用 类描述的方式、静态数组的方式 实现

前缀树原理和代码详解

前缀树的实现方式：

- 1) 类描述的实现方式（动态结构）。不推荐！虽然最常用。
 1. 路的可能性范围较小，用 固定数组 实现路
 2. 路的可能性范围较大，用 哈希表 实现路
- 2) 静态数组的实现方式。推荐！不仅笔试，就连比赛也能保证使用。
 1. 一切都是静态数组来实现，提交准备好够用的空间
 2. 如果路的可能性范围较大，就用每一位的信息建树。下节课前缀树的题目里展示