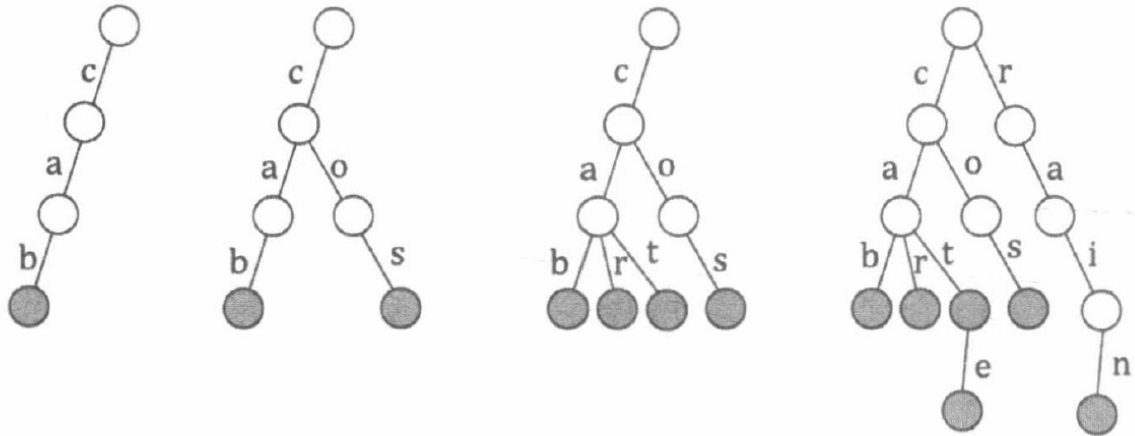


字典树Trie

Trie特点

- 字典树Trie，树的每条边上恰好对应一个字符，每个顶点代表从根到该节点的路径所对应的字符串。
- 任意一个节点所代表的字符串，都是实际字符串集合中某些串的前缀。
- 特别地，根节点表示空串。
- 因此，不难发现，Trie很好的利用了串的公共前缀，节约了存储空间。
- 若将字符集看作是小写的英文字母，则Trie也可以看作是一个26叉树。



在空 Trie 中
插入 cab

继续插入 cos

继续插入 car 与 cat

继续插入
cate 与 rain

常用操作

```
1 void insert(const char* str){
2     int p = 0;
3     for(int i = 0; str[i]; i++){
4         int u = str[i] - 'a';
5         if(!son[p][u]) son[p][u] = ++idx;
6         p = son[p][u];
7     }
8     cnt[p]++;
9 }
```

```
1 int query(const char* str){
2     int p = 0;
3     for(int i = 0; str[i]; i++){
4         int u = str[i] - 'a';
5         if(!son[p][u]) return 0;
6         p = son[p][u];
7     }
8     return cnt[p];
9 }
```

快乐刷题

- [P52 前缀统计](#)
- [P444 Phone List](#)
- [P69 The XOR Largest Pair](#)