



字符串学习笔记(2) 字典树与Border树



严格鸽
柚子厨/萝莉控/ACM银牌

已关注

42 人赞同了该文章

前面我们学到了如何处理字符串的 *next* 数组

严格鸽：字符串学习笔记(1) 基础概念与kmp前置

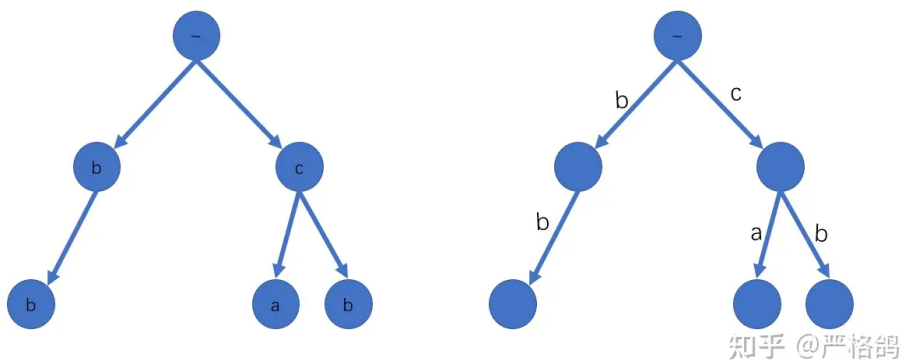
自动机的就是如何讲线性的字符串，扩展到树状的数据结构。

字典树

首先就是比较简单的字典树了。而且有很多教程都很好，所以这里就不做详细的解释了。

Pecco：算法学习笔记(43): 字典树

不过这里需要补充一个误区，在字典树中，保存字符信息的是边，而不是点。



不过感觉，两种理解方式都差不多~。

由于字典树以及是树上结构了，所以用一些树上的知识，比如树上dp，树刨什么的

严格鸽：(贪心/字典树) 代码源每日一题 Div1 删库

01字典树

Border树

我们回想下 $next[i]$ 数组的定义, $next[i]$ 是前缀 $pre[i]$ 的最大Border。

那么我们如何求出一个字符串 S 的全部Border呢?

我们可以先获得 S 的最大Border, 然后求最大Border的最大Border。根据Border的传递性

| S 的 Border 的 Border 也是 S 的 Border

那么 S 的最大Border的最大Border也是 S 的Border。

所以我们就是一个不断爬 $next[i]$ 的过程, 如果我们把 $next[i]$ 看成 $fa[i]$, 就是一颗树了。根节点为0。

题目链接

题意:

给一个长度为 n 的仅包含小写字母的字符串 S , 一个正整数 k , 求一个最长的字符串 T , 满足:

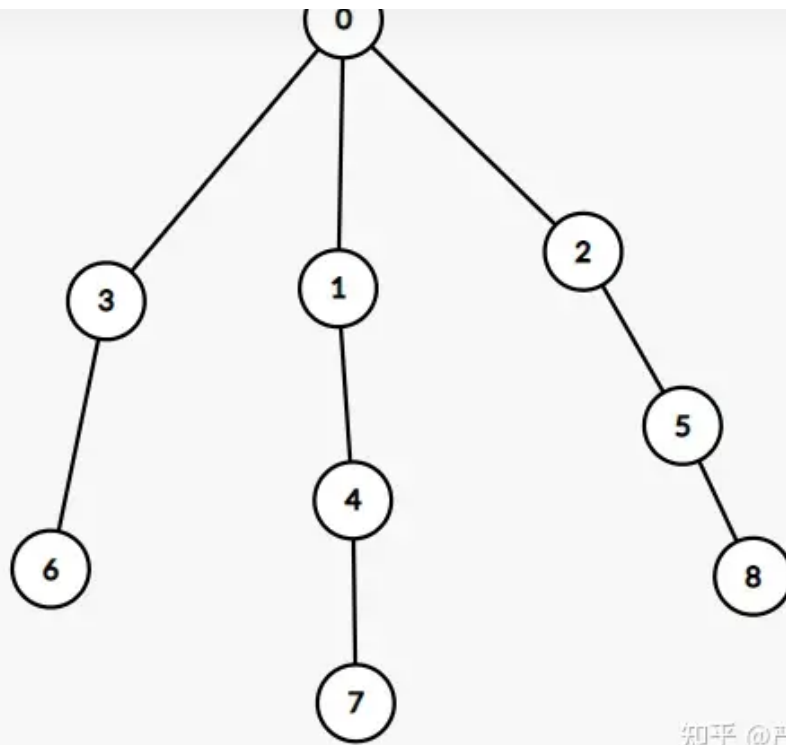
1. T 为 S 的前缀
2. T 为 S 的后缀
3. T 在 S 中至少出现 k 次

知乎 @严格鸽

首先 T 是一个Border。

我们求出 $S = abcabcb$ 的next数组并且建树

```
0 1
0 2
0 3
1 4
2 5
3 6
4 7
5 8
```



知乎 @严格鸽

从 $n = 8$ 出发，我们发现，**abcab** 是 S 的一个Border，**ab** 又是 **abcab** 的一个Border。

也就是说，**ab** 自身出现了一次，在 **abcab** 中作为后缀出现了一次，又在 **[abcab]cab** 中出现了一次。

所以 **ab** 出现了3次。

那么对于这个图来说，我们直接建树并且求出每个节点的子树大小，最后从 n 向上跑即可。

```

void dfs(int u) {
    sz[u] = 1;
    for (int v : g[u]) {
        dfs(v);
        sz[u] += sz[v];
    }
}

int main() {
    cin >> n >> k;
    cin >> s;
    s = "?" + s;
    get_next(s);
    for (int i = 1; i <= n; i++) {
        cout << nxt[i] << " " << i << endl;
        g[nxt[i]].push_back(i);
    }
    dfs(0);
    int ans = -1;
    for (int i = n; i >= 1; i = nxt[i]) {
        if (sz[i] >= k) ans = max(ans, i);
    }
    if (ans == -1) cout << -1 << endl;
    else {
        for (int i = 1; i <= ans; i++) cout << s[i];
        cout << endl;
    }
}

```

题意：

给一个长度为 n 的仅包含小写字母的字符串 S ，有 Q 次操作：

1. 修改操作：1 ch 表示向字符串末尾添加一个字符 ch
2. 查询操作：2 k ，求一个最长的字符串 T 满足： T 为 S 的前缀， T 为 S 的后缀，且 T 在 S 中至少出现 k 次

输入描述：

第一行一个整数 n, Q 分别表示初始字符串 S 长度，操作数

第二行为字符串 S

后面 Q 行，每行表示一个操作

$1 \leq n, q, k \leq 2e5$ ，保证 修改操作 中的 ch 为小写英文字母

知乎 @严格鸽

样例：

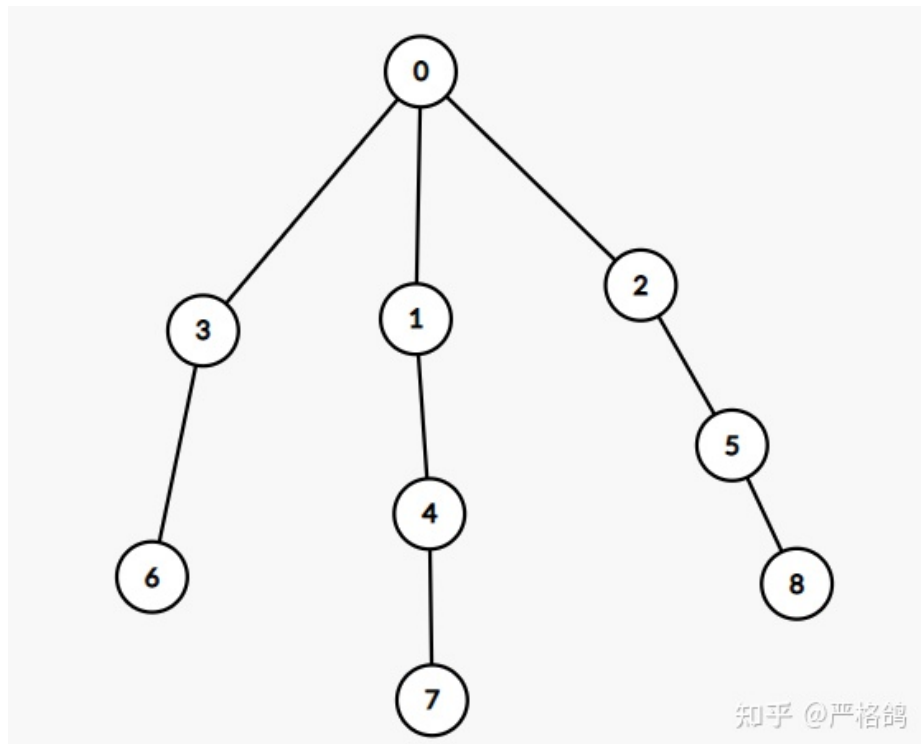
```
6 5
abcbabc
2 2
2 3
1 a
1 b
2 3
```

输出

```
3
-1
2
```

对于动态的问题，可以选择先离线下答案，最后的字符串 $S = abcbabcab$

我们对其建立Border树。



知乎 @严格鸽

就是上面的那个字符串

我们的任务还是统计子树大小，但是对于这个问题，我们需要动态的添加一个点，所以可以先跑一遍dfs序。

如果我们需要添加一个点，就只需要在**对应的dfs序的位置上加一**。

这样子树大小问题就转变成了**子树和的问题**，那么我们配合dfs序加上线段树/树状数组即可完成。

(过几天可能会出个dfs序的笔记，昨天的那个牛客的lca居然有dfs序的做法，好神奇啊qwq)

但是这样还没完，虽然我们处理出来了子树大小，但是我们还有个爬树的过程。

爬树的目的是找到深度最深的点，使得子树大小大于等于 k 。那么显然这个是个树链上的二分问题，我们可以使用树上倍增来解决。

严格鸽：(最小生成树/树上倍增) 代码源每日一题 Div1 【模板】最小瓶颈生成树 (数据加强版)

可以看到，字符串问题，在用上一些处理工具后，就变成了数据结构问题了。

回想一下，我们在kmp的时候是用到了 *next* 数组的，表示如果没有匹配上，我们应该如何跳跃。

而使用next数组建立的Border树还有个名字

<https://blog.csdn.net/article/details/>

失配树 (border树) _Jozky86的博客

2021年2月18日 — 4和6没有border关系举例子：当 $i = 3$ 时， $j = fa[i-1] = fa[2] = 0$ $s[j+1] = s[1] = a = s[3]$ 所以把 $i = 3$ 点的父亲设为 $j+1 = 1$ P5829 [模板]失配树。

<https://www.douaite.com/article/>

Border树 (失配树) - 豆奶特

参考资料1参考资料2前置知识：前缀函数，Border的简单认识约定：字符串的下标从 (0) 开始。

... Border树 (失配树) . 发布时间：2022-01-30 BORDER 失配 ...

知乎 @严格鸽

那个有个说法，ac自动机就是在字典树上进行kmp，也就是配合Border树。

AC 自动机 = Trie + Border 树

下一章不出意外就是自动机了qwq

ps：感觉牛客的字符串讲的很不错，至少让我听懂了qwq

另ps：第一期用了个比较正经的封面，但是感觉在一堆二次元里面好突兀的，所以还是二次元吧（

发布于 2022-07-26 09:57

字符串 ACM 竞赛 OI (信息学奥林匹克)



发一条带图评论吧

- 

永WA机

好耶，ygg要讲字符串系列了😄

2022-07-26 · IP 属地湖北

回复 1
- 

rankYu

好耶😄

2022-07-26 · IP 属地湖北

回复 1
- 

Acing

跟着ygg😄

2022-07-26 · IP 属地湖南

回复 1
- 

Space Walker

ygg给点串串练习题链接吧，每个专题都带几道练习题最好了😄😄

2022-08-05 · IP 属地浙江

回复 喜欢
- 

严格鸽 作者

好的，等我更完字符串内容救出习题qwq

2022-08-05 · IP 属地四川

回复 喜欢
- 

列奥那多是勇者

封面贴贴好馁

2022-07-30 · IP 属地浙江

回复 喜欢
- 

严格鸽 作者

贴贴😄

2022-07-30 · IP 属地四川

回复 喜欢
- 

Cgfyufsygsm 🏆

推荐 NOI2014 动物园

2022-07-26 · IP 属地贵州

回复 喜欢
- 

严格鸽 作者

好的qwq

2022-07-26 · IP 属地四川

回复 喜欢
- 

memset0

为啥感觉我学的时候都是叫fail树，是翻译的fail么（

2022-07-26 · IP 属地浙江

回复 喜欢
- 

严格鸽 作者

fail 匹配失败，失配树（

2022-07-26 · IP 属地四川

回复 喜欢



文章被以下专栏收录



字符串学习笔记

为了不在卡字符串的题目，勇敢勇敢我的朋友

推荐阅读



【c# .net】字典
(Dictionary)



8种Pythc
法，你知i

