

后缀自动机

它是什么

是一个自动机（直球）

是可以识别一个字符串所有后缀（或者说是识别一个字符串的子串）

它能用来干什么

它可以做到后缀数组能做到的一切事情。还可以做到AC自动机可以做到的一切事情，并且后缀自动机支持在线插入。

有时候还可以用来优化dp或者直接在后缀自动机上dp

它怎么写

在了解后缀自动机之前，我们可以了解一种叫后缀树的东西。我们把一个字符串的每一个后缀当成一个串，用这些串建出一棵trie树，这个trie树就是后缀树。由于字符串本质不同子串是 $O(n^2)$ 级别的，因此后缀树的节点数也是 $O(n^2)$ 级别的。

可以发现后缀树的每个节点对应一个后缀集合，每个点对应的后缀集合是这个点的字数中所有结束节点所表示的后缀的集合。我们可以发现有一个性质，对于只有一个儿子的费结束节点，他的后缀集合与他儿子的后缀集合相同，后缀自动机上只需要考虑所有后缀集合不同的节点，所以我们可以发现有意义的点只有结束节点和有一个以上儿子的节点。所以我们把所有非上述两种的点压缩到下面最近的一个上述两种点，我们就可以得到这个后缀自动机。

一个构造模版

```
struct SAM
{
    int trs[N<<1][26], fa[N<<1], len[N<<1];
    int cnt, last;
    void init(){cnt=last=1;}
    void insert(int x)
    {
        int p = last, np=++cnt, q, nq;
        last = np, len[np]=len[p]+1;
        end[np] = true;
        for(; p&&!trs[p][x]; p=fa[p]) trs[p][x] = np;
        if(!p) fa[np] = 1;
        else
        {
            q = trs[p][x];
            if(len[q]==len[p]+1) fa[np]=q;
            else

```

```
{
    fa[nq++cnt]=fa[q];
    len[nq]=len[p]+1;
    memcpy(trs[nq],trs[q],sizeof(trs[q]));
    fa[np] = fa[q] = nq;
    for(;p&&trs[p][x]==q;p=fa[p])trs[p][x]=nq;
}
}
}
} sam;
```

好题推荐

广义后缀自动机

From:
<https://wiki.cvbbacm.com/> - CVBB ACM Team

Permanent link:
https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:wangzai_milk:%E5%90%8E%E7%BC%80%E8%87%AA%E5%8A%A8%E6%9C%BA

Last update: 2020/05/16 00:10