

瞬间~永恒

博客

微博

相册

收藏

留言

关于我



暴风雪

浏览: 6373 次

性别:

来自: 济南

我在线

最近访客 [更多访客>>](#)



笔良文昌



eclipse-my



johnamo



cttnbcj

文章分类

- 全部博客 (93)
- 图论专区 (31)
- 学习资料 (6)
- Codeforces && TopCoder (3)
- 数据结构&&字符串 (47)
- DP&&搜索 (6)
- 模拟&&水题 (2)
- usaco (1)

社区版块

- 我的资讯 (0)
- 我的论坛 (0)
- 我解决的问题 (0)

存档分类

- 2012-04 (2)

[后缀数组]poj 3581: Sequence

博客分类: [数据结构&&字符串](#)

后缀数组

poj 3581

字符串

acm

数据结构

大致题意:

给出n个数,把这个数列分为三段,再把三段反转后连接在一起成为一个新串,求字典序最小的新串。

大致思路:

由于需要翻转,所以在输入时就按照反序输入。比如样例输入是5 10 1 2 3 4。我们从后向前读入就变为5 4 3 2 1 10。对这列数求出后缀数组。在大于2的后最中找到最小的后缀并输出。对于剩下的前缀s,我们把s串接到自己后面,也就是ss。再对这个串求出后缀数组,然后再把s中最小的前缀输出。最后把剩下的串输出。

对于第二步为什么要复制剩余串接在后面,用下面案例说明

6

10 1 2 2 3 4

第一步翻转后得到

4 3 2 2 1 10

求出后缀数组后得到最小的后缀便是: 1 10, 将其输出

剩下的串是 4 3 2 2.

我们如果直接从剩余串中找到最小后缀的话会产生以下结果。

最小后缀是 2, 输出。

输出剩余串 4 3 2。

最后得到1 10 2 4 3 2 很明显是wrong的

我们把剩余的串复制到剩余串的后面。

对 4 3 2 2 4 3 2 2 求出后缀数组。

得到前四个字符的最小的后缀是 2 2, 输出。

输出剩余串 4 3.

得到1 10 2 2 4 3

注意这里面数串中数字的范围没有给出,所以原来倍增算法模版中的桶排序需要改成快排。

Cpp代码

```
1. #include<iostream>
```

▪ [2012-03](#) (29)

▪ [2012-02](#) (39)

▪ [更多存档...](#)

评论排行榜

▪ [\[后缀数组\]poj 3581: Sequence](#)

▪ [\[DP+记忆化搜索\]hdoj 1224: Free DIY Tour](#)

▪ [\[堆排序\]poj 2823: Sliding Window](#)

▪ [\[模拟\]hdoj 4241: Xiangqi](#)

▪ [\[KMP\]hdoj 2087: 剪花布条](#)

最新评论

[暴风雪](#): 笔良文昌 写道还是不怎么明白什么是 最小覆盖子串。解释明白了? ...

[\[KMP\]poj 2185: Milking Grid](#)

[暴风雪](#): 笔良文昌 写道 next数组的意义其实就是 模式串 ...

[\[KMP\]hdoj 3336: Count the string](#)

[暴风雪](#): 笔良文昌 写道字哥, 你想多了。其实我们每次成功匹配后, 把赋值 ...

[\[KMP\]hdoj 2087: 剪花布条](#)

[笔良文昌](#): 还是不怎么明白什么是 最小覆盖子串。

[\[KMP\]poj 2185: Milking Grid](#)

[笔良文昌](#): next数组的意义其实就是 模式串 pat[0. ...

[\[KMP\]hdoj 3336: Count the string](#)

```
2. #include<cstdio>
3. #include<cstring>
4. #include<cmath>
5. #include<algorithm>
6. using namespace std;
7. const int Max = 500005;
8. const int inf=1<<30;
9. int n, num[Max];
10. int sa[Max], rankk[Max], height[Max];
11. int wa[Max], wb[Max], wv[Max], wd[Max];
12.
13.
14. int sa1[Max],rankk1[Max],c[Max],h[Max];
15.
16. int poww,count;
17.
18. int cmp(const void *a,const void *b){
19.     int x=(int*)a;
20.     int y=(int*)b;
21.     if(rankk[x]!=rankk[y])
22.         return(1);
23.     else
24.         if(rankk[x+poww]!=rankk[y+poww])
25.             return(1);
26.         else
27.             return(0);
28. }
29.
30. int cmp1(const void *a,const void *b){
31.     return num[*((int *)a)]-num[*((int *)b)];
32. }
33.
34. void da(int *m,int n){
35.     int i,j,k;
36.     for(i=0;i<n;i++)sa[i]=i;
37.     qsort(sa,n,sizeof(int),cmp1);
38.     for(i=0,j=0;i<n;i++){
39.         if(i>0&&m[sa[i]]!=m[sa[i-1]])
40.             j++;
41.         rankk[sa[i]]=j;
42.     }
43.
44.     for(poww=1;poww<n;poww*=2){
45.         for(i=0;i<n;i++)c[i]=0;
46.         for(i=0;i<n;i++){
47.             if(sa[i]+poww<n)c[rankk[sa[i]+poww]]++;
48.             else c[rankk[sa[i]]]++;
49.             for(i=1;i<n;i++)c[i]+=c[i-1];
50.             for(i=n-1;i>=0;i--)
51.                 if(sa[i]+poww<n)sa1[--c[rankk[sa[i]+poww]]]=sa[i];
52.                 else sa1[--c[rankk[sa[i]]]]=sa[i];
53.             for(i=0;i<n;i++)c[i]=0;
54.             for(i=0;i<n;i++)c[rankk[sa1[i]]]++;
55.             for(i=1;i<n;i++)c[i]+=c[i-1];
56.             for(i=n-1;i>=0;i--)sa[--c[rankk[sa1[i]]]]=sa1[i];
57.             for(i=0,j=0;i<n;i++){
58.                 if(i>0&&cmp(&sa[i],&sa[i-1])!=0)j++;
59.                 rankk1[sa[i]]=j;
60.             }
61.             for(i=0;i<n;i++) rankk[i]=rankk1[i];
62.         }
63.     }
64.
65. int main(){
66.     int n,i,j,ans,start,len,mmax,mmin,lab1,lab2;
```

```
67. // while(!=EOF){
68.     scanf("%d",&n);
69.     mmax=-inf;
70.     for(i=n-1;i>=0;i--){
71.         scanf("%d",&num[i]);
72.         num[i]++;
73.         mmax=max(mmax,num[i]);
74.     }
75.     num[n]=0;
76.     da(num,n+1);
77. // calHeight(num,n);
78.     mmin=inf;
79.     for(i=2;i<n;i++){
80.         if(rankk[i]<mmin){
81.             mmin=rankk[i];
82.             lab1=i;
83.         }
84.     }
85.     for(i=lab1;i<n;i++){
86.         printf("%d\n",num[i]-1);
87.     }
88.
89.     int k=lab1;
90.     for(i=0;i<lab1;i++){
91.         num[k++]=num[i];
92.     }
93.     mmin=inf;
94.     num[k]=0;
95.     da(num,k+1);
96. // calHeight(num,k);
97.     for(i=1;i<lab1;i++){
98.         if(rankk[i]<mmin){
99.             mmin=rankk[i];
100.            lab2=i;
101.        }
102.    }
103.    for(i=lab2;i<lab1;i++){
104.        printf("%d\n",num[i]-1);
105.    }
106.    for(i=0;i<lab2;i++){
107.        printf("%d\n",num[i]-1);
108.    }
109. // }
110. return 0;
111. }
```



分享到:  

◀ [\[字符串hash\]poj 1200: Crazy Search](#) | [\[贪心\]poj 3623: Best Cow Line. Gold](#) ▶

2012-02-23 21:26:27 | 浏览 134 | [评论\(2\)](#) | 分类: [编程语言](#) | [相关推荐](#) [MORE](#)

评论

2 楼 [暴风雪](#) 2012-02-24

luozhong915127 写道

输入是5 10 1 2 3 4。我们从后向前读入就变为5 4 3 2 1 10会是这样的结果吗, 🤔

你如果看题目的话就会知道，第一个数字(这里的5)表示的是这个数列中数字的个数，自然不用反过来~~囧

1 楼 [luozhong915127](#) 2012-02-24

输入是5 10 1 2 3 4。我们从后向前读入就变为5 4 3 2 1 10会是这样的结果吗，🤔

发表评论



[您还没有登录,请您登录后再发表评论](#)