

昵称: [proverbs](#)

园龄: 2年7个月

粉丝: 21

关注: 26

< 2013年2月 >						
日	一	二	三	四	五	六
27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	1	2
3	4	5	6	7	8	9

搜索

常用链接

[我的随笔](#)[我的评论](#)[我的参与](#)[最新评论](#)[我的标签](#)[更多链接](#)

我的标签

[POJ\(101\)](#)[计算几何\(61\)](#)[BZOJ\(54\)](#)[网络流\(37\)](#)[DP\(33\)](#)[字符串\(13\)](#)[费用流\(12\)](#)[HDU\(12\)](#)[SPOJ\(11\)](#)[TYVJ\(9\)](#)[更多](#)

随笔档案(241)

[2013年8月 \(1\)](#)[2013年4月 \(2\)](#)[2013年3月 \(32\)](#)[2013年2月 \(70\)](#)[2013年1月 \(74\)](#)[2012年11月 \(1\)](#)[2012年10月 \(37\)](#)[2012年9月 \(2\)](#)[2012年8月 \(22\)](#)

最新评论

[1. Re:OI教会我的](#)

看到你的博客更新停留在了2013年的8月，感觉非常伤感。常常看你写的题解。

-JoeFan

[2. Re:POJ 2175 费用流（消圈）](#)

@lwkds 退流用的反向边边权就是负的...

-Lytning

[3. Re:POJ 2175 费用流（消圈）](#)

边的权值都是正数，为什么会有负圈，求解？

-lwkds

阅读排行榜

[1. SPOJ 1825 Free tour II 树分治\(2475\)](#)[2. SBT 专辑\(2440\)](#)[3. Splay 专辑\(2396\)](#)[4. BZOJ 2038 小Z的袜子 莫队算法\(777\)](#)[5. BZOJ 1049 \[HAOI2006\]数字序列 DP\(694\)](#)

评论排行榜

BZOJ 1901 树状数组+函数式线段树

题解:

带点更新的区间第k大~

函数式线段树秒杀了~

与不带修改的函数式线段树唯一不同的在于线段树中的每个结点维护的都是这个位置的结点的树状数组值~

然后自己随便怎么胡搞啊神马的就 $n\log^2n$ 了~

```
1 #include <iostream>
2 #include <cstdlib>
3 #include <cstdio>
4 #include <cstring>
5 #include <algorithm>
6
7 #define N 110040
8 #define lowbit(x) x&-x
9
10 using namespace std;
11
12 int h[N],ls[N<<5],rs[N<<5],sum[N<<5];
13 int cnt,n,m,gs,num;
14 int a[N],bh[N];
15 int dat[N][4],ln[N],rn[N],lc,rc;
16
17 inline int newnode(int s,int l,int r)
18 {
19     ++cnt; sum[cnt]=s; ls[cnt]=l; rs[cnt]=r;
20     return cnt;
21 }
22
23 inline void build(int l,int r,int &rt)
24 {
25     rt=newnode(0,0,0);
26     if(l==r) return;
27     int mid=(l+r)>>1;
28     build(l,mid,ls[rt]);
29     build(mid+1,r,rs[rt]);
30 }
31
32 inline void updata(int last,int pos,int val,int l,int r,int &rt)
33 {
34     rt=newnode(sum[last]+val,ls[last],rs[last]);
35     if(l==r) return;
36     int mid=(l+r)>>1;
37     if(pos<=mid) updata(ls[last],pos,val,l,mid,ls[rt]);
38     else updata(rs[last],pos,val,mid+1,r,rs[rt]);
39 }
40
41 inline void modify(int x,int pos,int val)
42 {
43     int tmp;
44     while(x<=n)
45     {
46         updata(h[x],pos,val,1,num,tmp);
47         h[x]=tmp;
48         x+=lowbit(x);
49     }
50 }
51
52 inline int query(int l,int r,int k)
53 {
54     if(l==r) return l;
55     int tr=0,tl=0;
56     for(int i=l;i<=rc;i++) tr+=sum[ls[rn[i]]];
57     for(int i=l;i<=lc;i++) tl+=sum[ls[ln[i]]];
58     tr-=tl;
59     int mid=(l+r)>>1;
60     if(k<=tr)
```

1. BZOJ [BeiJing2006]狼抓兔子 平面图最大流(8)
2. POJ 2104/HDU 2665 区间k大值 函数式线段树(5)
3. POJ 3608 旋转卡壳(5)
4. 我的 2013(4)
5. BZOJ 1069 [SCOI2007]最大土地面积 旋转卡壳(3)

推荐排行榜

1. OI教会我的(3)
2. 为自己而奋斗(3)
3. BZOJ 1088 [SCOI2005]扫雷Mine(1)

```

61  {
62      for(int i=1;i<=rc;i++) rn[i]=ls[rn[i]];
63      for(int i=1;i<=lc;i++) ln[i]=ls[ln[i]];
64      return query(l,mid,k);
65  }
66  else
67  {
68      for(int i=1;i<=rc;i++) rn[i]=rs[rn[i]];
69      for(int i=1;i<=lc;i++) ln[i]=rs[ln[i]];
70      return query(mid+1,r,k-tr);
71  }
72 }
73
74 inline int getans(int l,int r,int k)
75 {
76     rc=lc=0;
77     while(r)
78     {
79         rn[++rc]=h[r];
80         r-=lowbit(r);
81     }
82     while(l)
83     {
84         ln[++lc]=h[l];
85         l-=lowbit(l);
86     }
87     return query(1,num,k);
88 }
89
90 inline void read()
91 {
92     scanf("%d",&n,&m);
93     for(int i=1;i<=n;i++)
94     {
95         scanf("%d",&a[i]);
96         bh[++gs]=a[i];
97     }
98     char str[4];
99     for(int i=1;i<=m;i++)
100     {
101         scanf("%s",str);
102         if(str[0]=='Q')
103         {
104             dat[i][0]=0;
105             for(int j=1;j<=3;j++) scanf("%d",&dat[i][j]);
106         }
107         else
108         {
109             dat[i][0]=1;
110             scanf("%d%d",&dat[i][1],&dat[i][2]);
111             bh[++gs]=dat[i][2];
112         }
113     }
114 }
115
116 inline void go()
117 {
118     sort(bh+1,bh+1+gs);
119     num=unique(bh+1,bh+1+gs)-bh-1;
120     for(int i=1;i<=n;i++) a[i]=lower_bound(bh+1,bh+1+num,a[i])-bh;
121     build(1,num,h[0]);
122     for(int i=1;i<=n;i++) modify(i,a[i],1);
123     for(int i=1;i<=m;i++)
124     {
125         if(dat[i][0]==0) printf("%d\n",bh[getans(dat[i][1]-1,dat[i][2],dat[i][3])]);
126         else
127         {
128             int pos=lower_bound(bh+1,bh+1+num,dat[i][2])-bh;
129             modify(dat[i][1],a[dat[i][1]],-1);
130             a[dat[i][1]]=pos;
131             modify(dat[i][1],a[dat[i][1]],1);
132         }
133     }
134 }
135
136 int main()
137 {
138     read(),go();
139     return 0;

```



没有人能阻止我前进的步伐，除了我自己！

标签: [树状数组](#), [BZOJ](#), [可持久化](#), [BZOJ 1901](#)

绿色通道:

[好文要顶](#)

[关注我](#)

[收藏该文](#)

[与我联系](#)



[proverbs](#)

[关注 - 26](#)

[粉丝 - 21](#)

[+加关注](#)

0

0

(请您对文章做出评价)

« 上一篇: [POJ 2104/HDU 2665 区间k大值 函数式线段树](#)

» 下一篇: [SPOJ 1811 LCS 后缀自动机](#)

posted @ 2013-02-19 19:09 proverbs 阅读(438) 评论(0) 编辑 收藏

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问](#) 网站首页。

【推荐】50万行VC++源码: 大型组态工控、电力仿真CAD与GIS源码库

【活动】ACE一键建站服务免费公测中

融云，免费为你的App加入IM功能——让你的App“聊”起来！！

最新IT新闻:

- [吃掉每一盘在你眼前的东西](#)
 - [AnyPresence](#)为移动与物联网开发者发布了一个新的API平台
 - [为什么说招到合适的人比融到钱更加重要?](#)
 - [在线教育先驱Lynda.com创始人: 二十年磨一剑](#)
 - [过来人告诉你众筹产品后期融资最佳时机](#)
- » [更多新闻...](#)

最新知识库文章:

- [驱动方法不能改变任何事情](#)
 - [推行TDD的思考](#)
 - [首席工程师揭秘: LinkedIn大数据后台是如何运作的](#)
 - [码农的思维训练: 超越专家](#)
 - [如何摆脱令人头疼的架构依赖?](#)
- » [更多知识库文章...](#)