

## 个人资料



haha593572013

访问: 291128次

积分: 9194

等级: BLOG &gt; 5

排名: 第708名

原创: 632篇 转载: 6篇

译文: 0篇 评论: 128条

## 文章搜索

## 文章分类

Dynamic Programming (76)

Contests (72)

Data Structures (54)

Codeforces (83)

插头DP (0)

Graphs (29)

Geometry (9)

Strings (10)

math (4)

SRM (23)

Good Problems (26)

算法总结 (8)

SGU (1)

Regional (5)

集训队 (7)

心路历程 (6)

Codechef (1)

linux (1)

## 文章存档

2015年02月 (2)

2015年01月 (1)

2014年06月 (1)

[微信公众平台开发入门](#)
[Markdown编辑器轻松写博文](#)
[软考, 你过了吗?](#)
[天天爱答题 一大波C币向你袭来](#)
[读文章说感想送好书](#)

## poj 2104 hdu 2665 区间第k大 可持久化线段树

分类: Data Structures

2012-09-30 01:12

3513人阅读

评论(2)

收藏

举报

query

build

好像又可以叫函数式线段树, 反正主要思想就是充分利用历史信息, 共用空间, 具体的看论文吧

这里讲的比较详细

<http://hi.baidu.com/wyl8899/item/e00796a9cb2df73d020a4d68>

这题A的太爽了\*\_\*

看CLJ标称的时候用的是动态申请内存的线段树, 非常不习惯--

昨天听说可以静态实现, 而且很简单, 于是重新想了下实现的思路, 果断开敲, 敲之前把各种细节, 包括如何调试都想的比较清楚, 然后实现起来真的是有如行云流水, 呵呵, 有一种新的想法, 然后通过不断的实践去实现它, 这种感觉真好啊

好了, 就献上我的代码吧, 我写线段树就是懒得开结构体。。。

注: 原先写的代码有个bug, poj数据太水, 被我过了, 在hdu wa了, 现在已经修正

献上能在hdu AC的代码

```
#include<cstdio>
#include<cstring>
#include<algorithm>
using namespace std;
const int maxn = 100010;
int ls[maxn*20], rs[maxn*20], sum[maxn*20];
int T[maxn], tot, rt;
void build(int l, int r, int &rt)
{
    rt=++tot;
    sum[rt]=0;
    if(l==r) return ;
    int m=l+r>>1;
    build(l, m, ls[rt]);
    build(m+1, r, rs[rt]);
}
void update(int last, int p, int l, int r, int &rt)
{
    rt=++tot;
    ls[rt] = ls[last];    rs[rt] = rs[last];    sum[rt] = sum[last] + 1;
    if(l==r) return ;
    int m=l+r>>1;
    if(p<=m) update(ls[last], p, l, m, ls[rt]);
    else update(rs[last], p, m+1, r, rs[rt]);
}
int query(int ss, int tt, int l, int r, int k)
{
    if(l==r) return l;
    int m=l+r>>1;
    int cnt=sum[ls[tt]] - sum[ls[ss]];
    if(k <= cnt)
        return query(ls[ss], ls[tt], l, m, k);
    else
        return query(rs[ss], rs[tt], m+1, r, k-cnt);
}
int num[maxn], san[maxn], n, m;
void gogogo()
{
    int l, r, k;
    for(int i=1; i<=n; i++)
    {
        scanf("%d", &num[i]);
        san[i]=num[i];
    }
    tot=0;
    sort(san+1, san+n+1);
    int cnt=unique(san+1, san+n+1)-san-1;
```

2014年05月 (2)

2014年04月 (1)

展开

阅读排行

2012 成都网络赛小记

(3739)

2012 天津网络赛小记

(3626)

poj 2104 hdu 2665 区间

(3512)

2013 ACMICPC Asia Re

(2844)

bzoj 1901 有更新区间第k

(2590)

hdu 4441 splaytree 201

(2415)

codeforces 208 E 时间戳

(2270)

长春网络赛小记 hdu 427

(2180)

zoi 3408 GAO 最短路 DF

(2022)

codeforces 117C 拓扑排

(1804)

最新评论

嗯，我是一个偏执狂。  
NiceSpace: 3xian大神orz

hdu 4419  
hq572241670: lz 你那个build里面，n-2是什么意思？为什么减2？

poj 2002 哈希函数的设计  
XIANGYUAN2014: 正在学习这个东西，初学者，谢谢分享，如果能够配上详细的解释就好了~

AC 两年 + 大学生活小记  
请叫我acm渣渣: 呵呵 歧哥加油

14年广东省赛，北师邀请赛  
陵子胥: J题怎么做啊

第十一届浙江省大学生程序设计j  
crazy\_石头: 噢....铭爷去WC被杰哥和歧哥都提到了....OTL

recently  
Take\_me\_fly: 加油楼主，再搞一年不？

第十一届浙江省大学生程序设计j  
Nero\_\_\_: 为什么你们都抓住了窝去WC这个梗--b

Codeforces Round #236 (Div. 1  
sdad120: 问下叉姐的博客是..?

hdu 2126 DP  
癡子也疯狂TT: 教教我吧，大神。qq: 2380349552加我。

```
build(1,cnt,T[0]);
for(int i=1;i<=n;i++) num[i]=lower_bound(san+1,san+cnt+1,num[i])-san;
for(int i=1;i<=n;i++) update(T[i-1],num[i],1,cnt,T[i]);
while(m--){
    scanf("%d%d%d",&l,&r,&k);
    int id=query(T[l-1],T[r],1,cnt,k);
    printf("%d\n",san[id]);
}
}
int main()
{
    int t;
    scanf("%d",&t);
    while(t--){
        scanf("%d%d",&n,&m);
        gogogo();
    }
    return 0;
}
```

上一篇

zoi 3645

下一篇

splay的替罪羊数据结构

顶

1

踩

0

主题推荐

poj

algorithm

namespace

函数式

cstring

猜你在找

ACM练级日志可持久化线段树初级-POJ 2104	HDU2665函数式线段树-区间第K大
hdu 2665 Kth number 可持久化线段树	HDU2665 Kth number 可持久化线段树
划分树学习小记 Poj 2104+Poj 2761+Hdu 2665 区间	POJ-2104-K-th Number函数式线段树
poj 2104 K-th Number划分树或者函数式线段树	poj 2104 K-th Number函数式线段树
poj2104 K-th Number 函数式线段树	poj 2104 K-th Number 函数式线段树

准备好了么？跳吧！

更多职位尽在 CSDN JOB

PHP程序员 （薪资10k+）	我要跳槽	【20K-35K】2D游戏Cocos2D客户端开发	我要跳槽
上海润宏信息技术有限公司	9-15K/月	北京七六网络技术有限公司	20-35K/月
虚拟化研发工程师	我要跳槽	虚拟化产品产品研发工程师	我要跳槽
上海优刻得信息技术有限公司	10-20K/月	奇虎360	8-25K/月

查看评论

2楼 ACMaxx2014-01-13 14:43发表

C

不用了，终于调过了。%>\_<%

1楼 ACMaxx2014-01-13 13:41发表

C

大牛，能告诉我，你发现的bug是什么吗？  
我在POJ上过了，在HDU上一直WA，看了好几个小时了，还是没看出来哪里有问题。

您还没有登录,请[\[登录\]](#)或[\[注册\]](#)

\* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

核心技术类目

全部主题   Hadoop   AWS   移动游戏   Java   Android   iOS   Swift   智能硬件   Docker   OpenStack  
VPN   Spark   ERP   IE10   Eclipse   CRM   JavaScript   数据库   Ubuntu   NFC   WAP   jQuery  
BI   HTML5   Spring   Apache   .NET   API   HTML   SDK   IIS   Fedora   XML   LBS   Unity  
Splashtop   UML   components   Windows Mobile   Rails   QEMU   KDE   Cassandra   CloudStack  
FTC   coremail   OPhone   CouchBase   云计算   iOS6   Rackspace   Web App   SpringSide  
Maemo   Compuware   大数据   aptech   Perl   Tornado   Ruby   Hibernate   ThinkPHP   HBase  
Pure   Solr   Angular   Cloud Foundry   Redis   Scala   Django   Bootstrap

[公司简介](#) | [招贤纳士](#) | [广告服务](#) | [银行汇款帐号](#) | [联系方式](#) | [版权声明](#) | [法律顾问](#) | [问题报告](#) | [合作伙伴](#) | [论坛反馈](#)

[网站客服](#)   [杂志客服](#)   [微博客服](#)   [webmaster@csdn.net](#)   400-600-2320 | 北京创新乐知信息技术有限公司 版权所有 | 江苏乐知网络技术有限公司 提供商务支持  
京 ICP 证 070598 号 | Copyright © 1999-2014, CSDN.NET, All Rights Reserved 