**引用 （转）树状数组与逆序数**

[算法世界](http://blog.163.com/justly@yeah/blog/#m=0&t=1&c=fks_087070086084085069087081080095085084080067087081086067084) 2009-09-14 19:36:23 阅读174 评论0   字号：大中小 订阅

[lightyue\_leyuan](http://blog.163.com/lightyue_leyuan/) 的 [（转）树状数组与逆序数](http://blog.163.com/lightyue_leyuan/blog/static/75136077200962903533390)

有关树状数组

一维树状数组基本构造：

（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客一维树状数组  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客#define lowbit(x)  ( (x)&(-(x)) )  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客void  update( int pos, int value )      // 更新 pos 的值  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客{  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    int x= pos;  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    while( x<= n )  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    （转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客{  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        count[x]+= value;  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        x+= lowbit(x);  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    }  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客}  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客int  getsum( int pos )                  //  求 1 到 pos 位置的和  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客{  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    int x= pos, sum= 0;  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    while( x )  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    （转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客{  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        sum+= count[x];  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        x-= lowbit(x);  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    }  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客      
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    return sum;  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客}

相关习题： <http://acm.pku.edu.cn/JudgeOnline/problem?id=2352>

树状数组可以用来求逆序数， 当然一般用归并求。  
如果数据不是很大， 可以一个个插入到树状数组中， 每插入一个数， 统计  
比他小的数的个数，对应的逆序为 i- getsum( data[i] ),其中 i 为当前已  
经插入的数的个数， getsum( data[i] ）为比 data[i] 小的数的个数  
i- sum( data[i] ) 即比 data[i] 大的个数， 即逆序的个数  
但如果数据比较大，就必须采用离散化方法。

一关键字的离散化方法：

所谓离散化也就是建立一个一对一的映射。 因为求逆序时只须要求数据的相应  
大小关系不变。  
如： 10 30 20 40 50  与  1 3 2 4 5 的逆序数是相同的

定义一个结构体  struct Node{ int data;    // 对应数据  
                                                   int pos;     // 数据的输入顺序 };  
先对 data 升序排序， 排序后，pos 值对应于排序前 data 在数组中的位置。  
再定义一个数组 p[N]， 这个数组为原数组的映射。以下语句将按大小关系  
把原数组与 1到 N 建立一一映射。

int id= 1; p[ d[1].pos ]= 1;  
for( int i= 2; i<= n; ++i )  
if( d[i].data== d[i-1].data ) p[ d[i].pos ]= id;  
else                                    p[ d[i].pos ]= ++id;

相关习题：<http://acm.pku.edu.cn/JudgeOnline/problem?id=2299>

（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客poj 2298  
#include <stdio.h>  
#include <stdlib.h>  
#include <string.h>  
  
#define N 500010   
#define lowbit(x)  ( (x)&(-(x)) )  
  
struct Node{ int data, pos; };  
  
int n;  
int count[N], p[N];  
Node d[N];  
  
int cmp( const void\* a, const void\* b )  
{  
    Node\* ta= (Node\*)a;  
    Node\* tb= (Node\*)b;  
      
    return ta->data- tb->data;  
}  
  
void  update( int pos, int value )  
{  
    int x= pos;  
    while( x<= n )  
    {  
        count[x]+= value;  
        x+= lowbit(x);  
    }  
}  
  
int  getsum( int pos )  
{  
    int x= pos, sum= 0;  
    while( x )  
    {  
        sum+= count[x];  
        x-= lowbit(x);  
    }  
      
    return sum;  
}  
  
int main()  
{  
    while( scanf("%d",&n)!= EOF )  
    {  
        if( n== 0 ) break;  
          
        for( int i= 1; i<= n; ++i )  
        {  
            scanf("%d", &d[i].data );  
            d[i].pos= i;   
        }  
              
        qsort( d+ 1, n, sizeof(d[0]), cmp );  
          
        int id= 1; p[ d[1].pos ]= 1;  
  
        for( int i= 2; i<= n; ++i )  
        if( d[i].data== d[i-1].data ) p[ d[i].pos ]= id;  
        else                          p[ d[i].pos ]= ++id;  
          
        memset( count, 0, sizeof(count) );  
          
        \_\_int64 ans= 0;  
        for( int i= 1; i<= n; ++i )  
        {      
            update( p[i], 1 );  
            ans+= (\_\_int64)( i- getsum( p[i] ) );              
        }  
          
        printf("%I64d\n", ans );  
    }  
      
    return 0;  
}

二关健字的离散方法：

先对第一个关键字进行离散化，然后对第二关键字排序。

相关习题：  
[http://acm.hnu.cn:8080/online/?action=problem&type=show&id=10069&courseid=0](http://acm.hnu.cn:8080/on%3Cwbr%3Eline/?act%3cwbr%3eion=problem&type=show&id=10069&courseid=0)

（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客#include <stdio.h>  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客#include <stdlib.h>  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客#include <string.h>  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客#define MAX  100001  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客#define lowbit(x)  ( (x)&( (x)^( (x)- 1 ) ) )  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客struct Point  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客{  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    int x, y, id, xid;  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客};  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客int    n;  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客Point  pos[MAX];  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客int    count[MAX];  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客int    result[MAX];  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客int cmpa( const void\* a, const void\* b )  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客{  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    Point\*  ta= (Point\*)a;  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    Point\*  tb= (Point\*)b;  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客      
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    if( ta->x== tb->x ) return tb->y- ta->x;  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客      
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    return ta->x- tb->x;  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客}  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客int cmpb( const void\* a, const void\* b )  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客{  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    Point\* ta= (Point\*)a;  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    Point\* tb= (Point\*)b;  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客      
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    if( ta->y== tb->y ) return ta->x- tb->x;  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客      
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    return tb->y- ta->y;  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客}  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客void add( int i )  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客{  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    while( i<= MAX )  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    （转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客{  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        count[i]++;  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        i+= lowbit(i);  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    }  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客}  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客int sum( int i )  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客{  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    int total= 0;  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    while( i>= 1 )  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    （转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客{  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        total+= count[i];  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        i-= lowbit(i);  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    }  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    return total;  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客}  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客int main()  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客{  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    while( scanf("%d", &n)!= EOF )  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    （转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客{  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        memset( count, 0, sizeof(count) );  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        memset( result, 0, sizeof(result) );  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客          
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        for( int i= 0; i< n; ++i )  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        （转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客{  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客            scanf("%d%d", &pos[i].x, &pos[i].y );  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客            pos[i].id= i;  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        }  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客          
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        qsort( pos, n, sizeof(pos[0]), cmpa );  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客          
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        int tx= pos[0].x,t= 1;  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        for( int i= 0; i< n; ++i )  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        （转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客{  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客            if( pos[i].x== tx  )  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客                pos[i].xid= t;  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客            else  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客            （转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客{  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客                pos[i].xid= ++t;  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客                  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客                tx= pos[i].x;  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客            }  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        }  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        qsort( pos, n, sizeof(pos[0]), cmpb );  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客          
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        for( int i= 0; i< n; ++i )  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客         （转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客{  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客            result[ pos[i].id ]= sum( pos[i].xid );      
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客              
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客            if( i> 0 && pos[i].x== pos[i-1].x && pos[i].y== pos[i-1].y )   
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客                     result[ pos[i].id ]= result[ pos[i-1].id ];  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客                       
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客            add( pos[i].xid );  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        }  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客              
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        for( int i= 0; i< n; ++i )  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        （转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客{  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客            if( i== 0 )  printf("%d", result[i] );  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客            else         printf(" %d", result[i] );  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        }  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客        printf("\n");  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    }  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客      
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客    return 0;  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客}  
（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客

二维树状数组

基本构造：

（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客（转）树状数组与逆序数 - lightyue_leyuan - lightyue_leyuan的博客  
#define MAXN 1025  
  
int count[MAXN][MAXN];  
int n;  
  
int lowbit( int x ){ return x&(-x); }  
  
void update( int x, int y, int value ){  
    int i= x, j= y;  
    while( i<= n ){  
        j= y;  
        while( j<= n ){  
            count[i][j]+= value;  
            j+= lowbit(j);  
        }      
        i+= lowbit(i);  
    }  
}  
  
// 计算 ( 1, 1 )到 ( x, y )区域的和   
int getsum( int x, int y )   
{  
    int i= x, j= y, sum= 0;  
      
    while(i){  
        j= y;  
        while(j){          
            sum+= count[i][j];  
            j-= lowbit(j);  
        }  
        i-= lowbit(i);  
    }  
    return sum;  
}  
  
// x2> x1 && y2> y1 (x1,y1)到(x2,y2)和   
int total( int x1, int y1, int x2, int y2 )   
{  
    return getsum( x2, y2 )- getsum( x2, y1- 1 )  
        - getsum( x1- 1, y2 )+ getsum( x1- 1, y1- 1 );   
}

相关习题： <http://acm.pku.edu.cn/JudgeOnline/problem?id=3321>