

原创

2017 ACM-ICPC 亚洲区（南宁赛区）网络赛 F. Overlapping Rectangles (扫描线)

2019-08-23 20:26:21 _Y-_-Y- 阅读数 32 更多

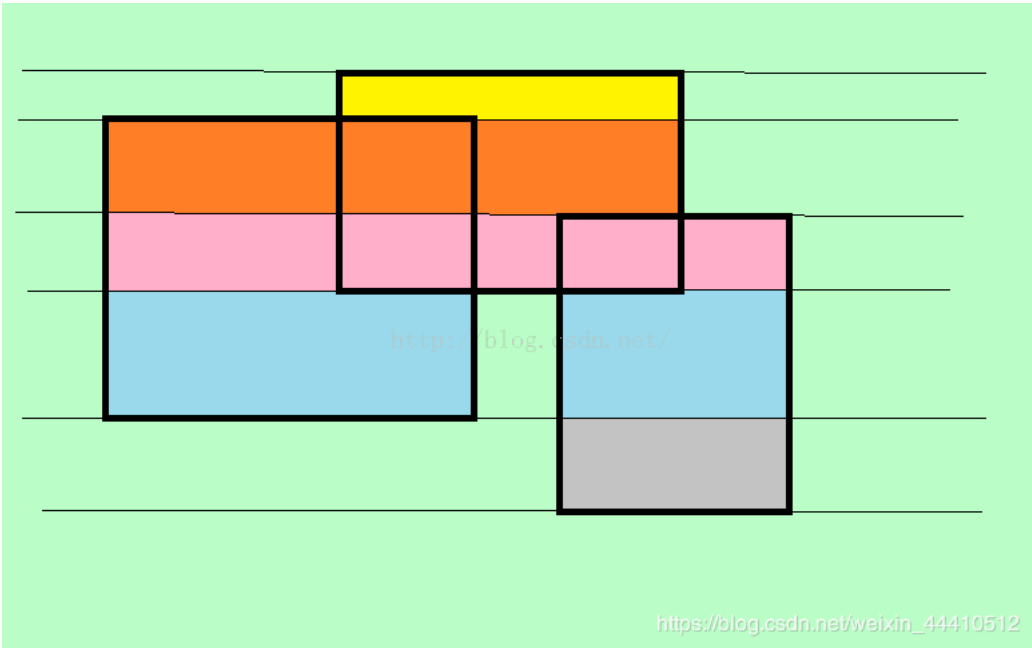
编辑

版权声明：本文为博主原创文章，遵循 CC 4.0 BY-SA 版权协议，转载请附上原文出处链接和本声明。
本文链接：https://blog.csdn.net/weixin_44410512/article/details/100044727

参考：<https://blog.csdn.net/tomorrowtodie/article/details/52048323>

Overlapping Rectangles

矩形面积并 经典的扫描线问题
扫描线可以想象成一根假想的线，
将图从左往右或从右往左或自下而上或自上而下“扫描”一遍



可以看到 6 根扫描线将图像分成了 5 个部分
各个颜色就是有效面积，

假设从下往上看，首先，看到了最下面灰色矩形的下边，
用这个下边的长度（宽）乘以这条边和上一条边的高度差（高）即得到灰色矩形面积，
继续看到蓝色的矩形的下边，虽然蓝色矩形有两个，
但我们计算时自然会用结合律将两个矩形的下边加起来再去乘以同样的高，
然后重复这样的操作，我们最终可以求得整个图形的面积。

机器是这样实现的

存图，只需要保存这张图里面的所有水平的边
边的属性：边的左右端点（横坐标），边的高度（纵坐标），边属于矩形的上边还是下边

排序，因为要从下往上依次扫描
离散，因为题目中给的数据往往是浮点数或者是给定区间范围很大

计算有效面积的‘宽’ 用到了线段树
怎么去维护这个区间的有效长度（即被覆盖的长度）
因为是从上而下的扫这个图，
在同一个矩阵中，扫到下边相当于刚刚进入这个矩形，扫到上边则是要离开这个矩形
那么如果有扫描线在这个矩阵中间，那么中间的扫描线肯定会有这个矩阵的有效长度，
所以线段树会随着扫描线的移动而记录有效长度
如果扫到的这条边是某矩形的下边，则往区间插入这条线段
表示在没到这个矩阵上边的时候，这个区间是有效长度

如果扫到的这条边是某矩形的上边，则往区间删除这条线段

表示这个区间不是有效长度

计算有效面积的 ‘高’ 两个扫描线之间的高

AC code

```

1  #include <cstdio>
2  #include <cstring>
3  #include <algorithm>
4  #include <iostream>
5  using namespace std;
6  #define maxn 5010
7  struct node{
8      int ss;
9      double l,r,h;
10     node(double l1=0,double r1=0,double h1=0,int ss1=0){
11         l=l1;
12         r=r1;
13         h=h1;
14         ss=ss1;
15     }
16 }edge[maxn<<2];
17 int add[maxn];
18 double x[maxn<<2],sum[maxn<<2];
19 bool cmp(node a,node b){
20     return a.h<b.h;
21 }
22 void pushup(int now,int l,int r){
23     if(add[now]) sum[now]=x[r+1]-x[l];
24     else if(l==r) sum[now]=0;
25     else sum[now]=sum[now<<1]+sum[now<<1|1];
26 }
27 void update(int now,int L,int R,int val,int l,int r){
28     if(L<=l&&r<=R){
29         add[now]+=val;
30         pushup(now,l,r);
31         return ;
32     }
33     int mid=(l+r)>>1;
34     if(L<=mid) update(now<<1,L,R,val,l,mid);
35     if(R>mid) update(now<<1|1,L,R,val,mid+1,r);
36     pushup(now,l,r);
37 }
38 int main(){
39     int n;
40     double x1,x2,y1,y2,ans;
41     while(cin>>n && n){
42         ans=0;
43         int top=0,l,r;
44         for(int i=0;i<n;i++){
45             cin>>x1>>y1>>x2>>y2;
46             x[top]=x1;
47             edge[top++]=node(x1,x2,y1,1);
48             x[top]=x2;
49             edge[top++]=node(x1,x2,y2,-1);
50         }
51         sort(x,x+top);
52         sort(edge,edge+top,cmp);
53         int k = 1;
54         for(int i=1; i<top; i++){
55             {
56                 if(x[i] != x[i-1])
57                     x[k++] = x[i];
58             }
59             memset(add,0,sizeof(add));
60             memset(sum,0,sizeof(sum));
61             for(int i=0; i<top-1; i++){
62                 l=lower_bound(x,x+k,edge[i].l)-x;
63                 r=lower_bound(x,x+k,edge[i].r)-x-1;
64                 update(1,l,r,edge[i].ss,0,k-1);

```

```
65         ans+=(sum[1]*(edge[i+1].h-edge[i].h));
66     }
67     cout<<(int)ans<<endl;
68 }
69 cout<<"*<<endl;
70 return 0;
71 }
72
```

有 0 个人打赏

文章最后发布于: 2019-08-23 20:26:21

©2019 CSDN 皮肤主题: 大白 设计师: CSDN官方博客

编程语言大PK，你选谁？

关闭