

信奥算法



现场挑战 快快编程961

快快编程
kkcoding.net

段更新

段查询

整体增加

整段最值查询

Range Maximum Query
简称RMQ

编号i	1	2	3	4	5	6	7	8	9	10
a[i]	3	1	14	1	5	9	12	16	5	3

编号i	1	2	3	4	5	6	7	8	9	10
b[i]	1	1	1	2	2	2	3	3	3	4

第1块	第2块	第3块	第4块
-----	-----	-----	-----

bm[k]代表k号块的块内最大值

编号k	1	2	3	4
bm[k]	14	19	16	3

编号k	1	2	3	4
c[k]	0	10	0	0

c[k]代表k号块内整体被延后的增量	尚未更新
--------------------	------

数组含义

a数组

$a[i]$ 代表原数组*i*号数值

共*n*个

b数组

$b[i]$ 代表原*i*号元素对应块号

共*n*个

c数组

$c[k]$ 代表*k*号块内整体被延后的增量

bm数组

$bm[k]$ 代表*k*号块的块内最大值

```

49  cin>>n>>m;
50  B=sqrt(n)+1;
51  for(ll i=1;i<=n;i++)b[i]=
52  for(ll i=1;i<=n;i++){
53      cin>>a[i];
54      bm[b[i]]=max(a[i],bm[b[i]]);
55  }
56  for(ll i=1;i<=m;i++){
57      ll t,l,r,k;
58      cin>>t>>l>>r>>k;
59      if(t==1) add(l,r,k);
60      else cout<<rmq(l,r)<<endl;
61  }

```

补全
程序

易错点汇总

易错点

头尾端点在同一块内

易错点

元素真实数值尚未更新
 $a[i] + c[b[i]]$ 才是真值

易错点

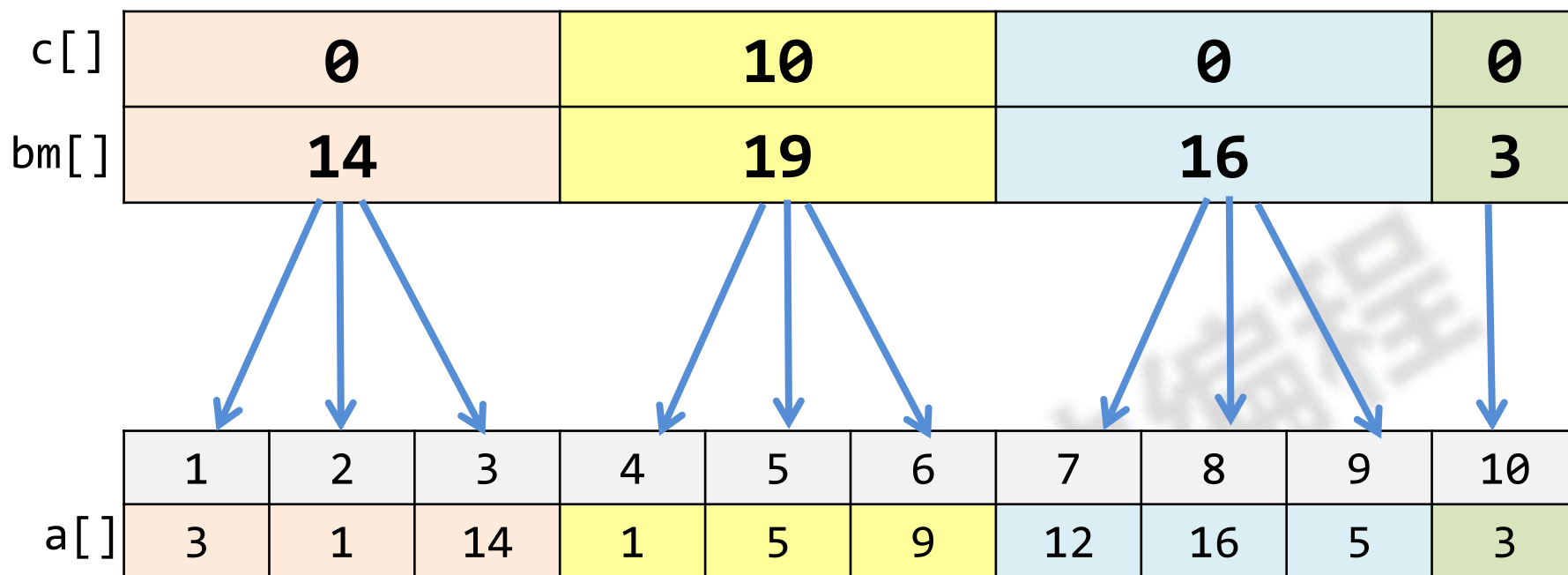
$bm[k]$ 是真最值

易错点

元素编号 i 和块号 k 混淆



分块：二级索引





现场挑战 快快编程1058

快快编程
kkcoding.net

输入	输出几
10 5	输出
4 3 4 1 5 9 4 2 1 8	3 1 10 4 5
1 8 3	
6 6 9	
1 10 10	
3 7 7	
4 9 5	

静态

段查询

编号区间+值域区间
元素计数

尝试简化问题获得灵感

若所有询问都是 $[1, n]$ 编号区间
请同学写出解法?

排序后+二分查找做元素统计

静态

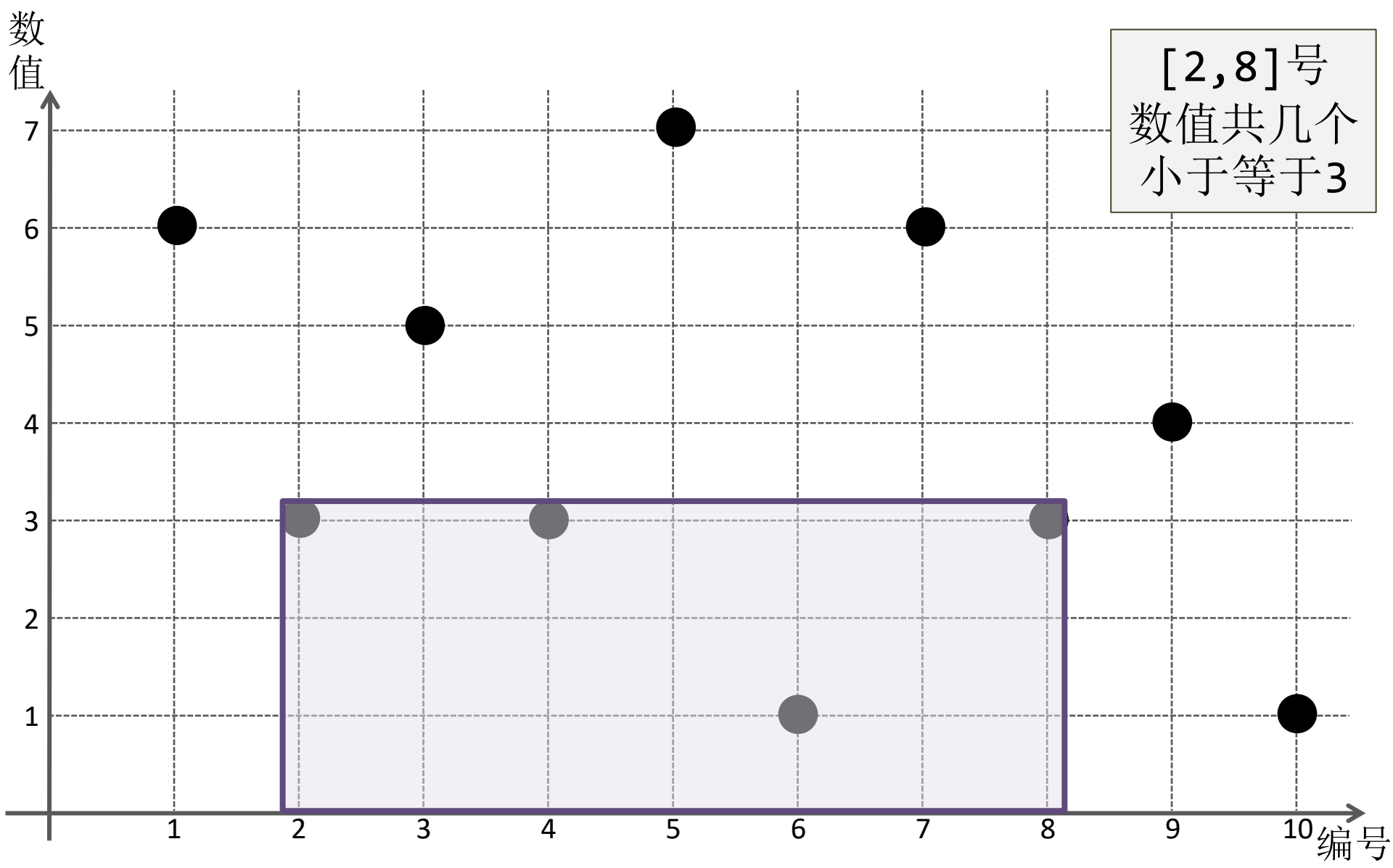
段查询

编号区间+值域区间
元素计数

一维数组用二维平面点可视化

快快编程
kkcoding.net

6	3	5	3	7	1	6	3	4	1
---	---	---	---	---	---	---	---	---	---



静态

段查询

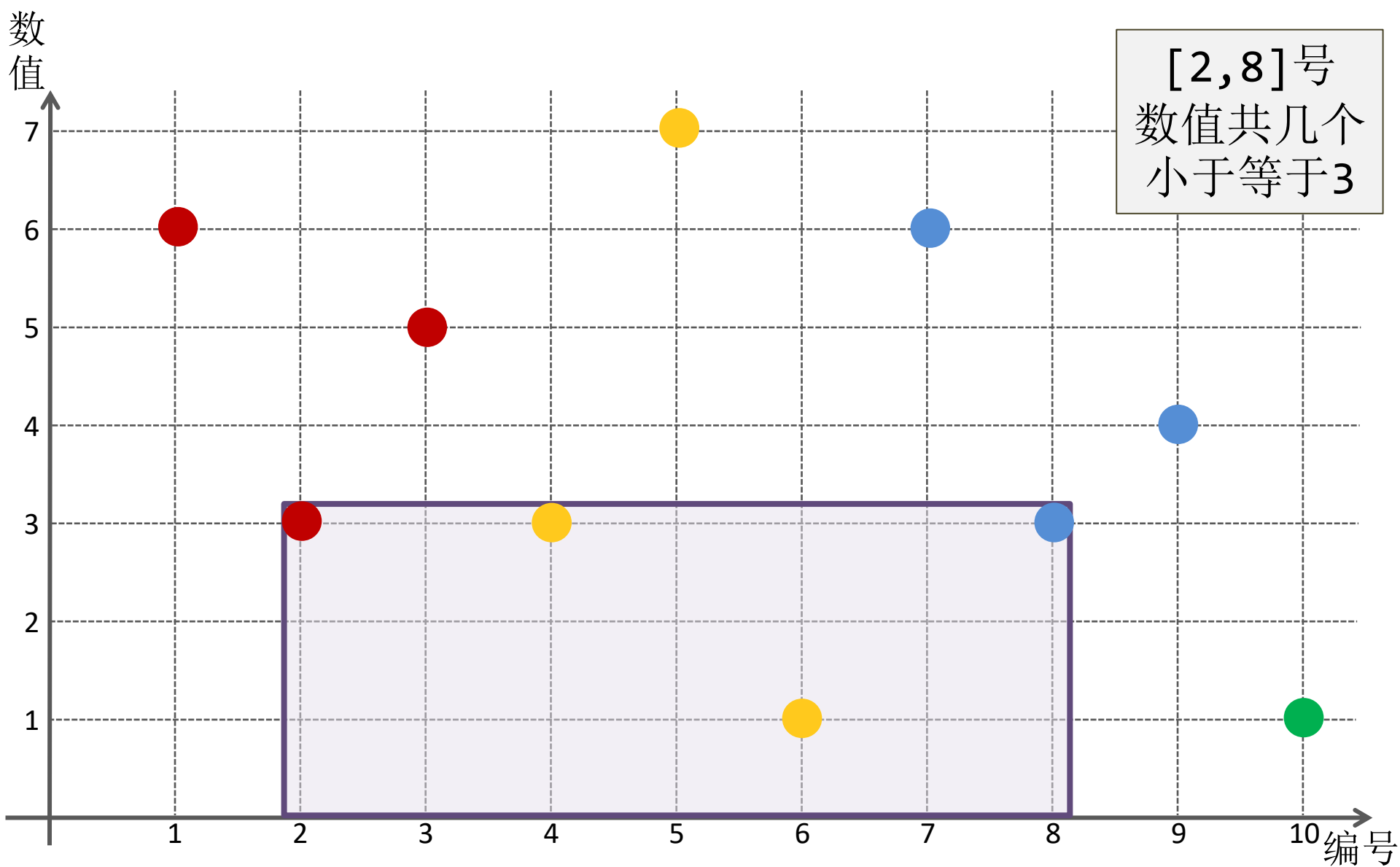
编号区间+值域区间
元素计数

分块算法

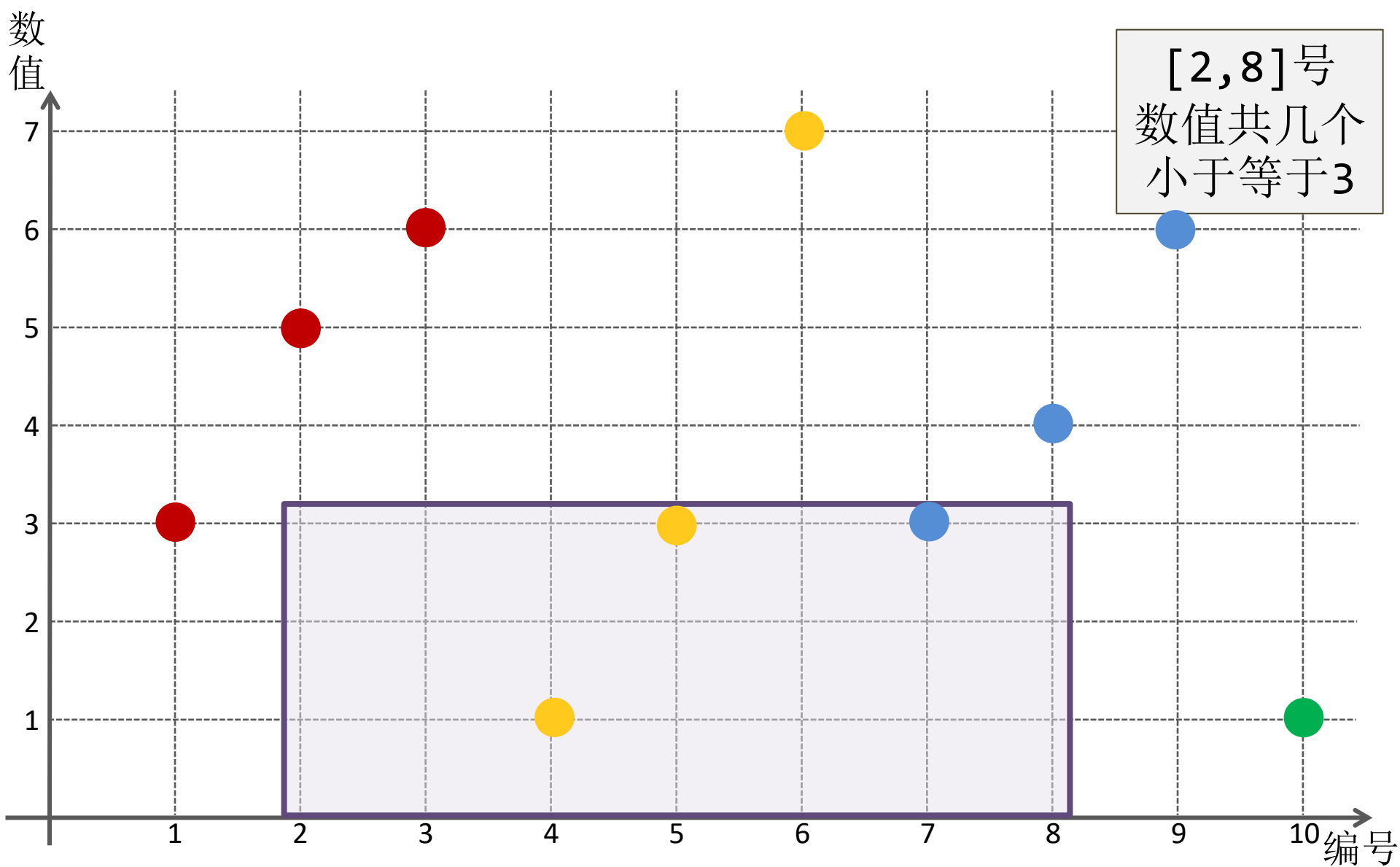
小区间拼凑大区间

观察之后2个询问演示
思考如何使整块内统计较快

6	3	5	3	7	1	6	3	4	1
---	---	---	---	---	---	---	---	---	---



3	5	6	1	3	7	3	4	6	1
---	---	---	---	---	---	---	---	---	---



块内排序

静态

段查询

编号区间+值域区间
元素计数

分块算法

小区间拼凑大区间

块内排序后
可以二分查找
统计特定值域的元素个数

```
20 cin>>n>>m;  
21 for(ll i=1;i<=n;i++)cin>>a[i];  
22 B= 分块大小稍后计算
```

```
23 for(ll i=1;i<=n;i++)b[i]=(i-1)/B+1;  
24 for(ll j=1;j<=b[n]-1;j++) {  
25     x[j]=1+(j-1)*B;  
26     y[j]=j*B;  
27 }
```

请写出x[j],y[j]含义

```
28 x[b[n]]=1+(b[n]-1)*B;  
29 y[b[n]]=n;
```

```
30 for(ll i=1;i<=n;i++) as[i]=a[i];  
31 for(ll j=1;j<=b[n];j++) sort(as+x[j],as+y[j]+1);
```

块内排序

```
32 for(ll i=1;i<=m;i++){  
33     ll l,r,k;  
34     cin>>l>>r>>k;  
35     cout<<query(l,r,k)<<" ";  
36 }
```

请写出as[i]含义

```
6 11 query(11 l,11 r,11 k){
7     11 cnt=0;
8     if(b[l]==b[r]){
9         for(int i=l;i<=r;i++)cnt+=(a[i]<=k);
10        return cnt;
11    }
12    for(int i=l;b[i]==b[l];i++)cnt+=(a[i]<=k);
13    for( )cnt+=(a[i]<=k);
14    for(int j=b[l]+1;j<=b[r]-1;j++)
15        cnt+=
16    return cnt;
17 }
```

最优分块大小

选取分块大小B
使时间复杂度最小化

时间复杂度分2部分
整块计算量+边角计算量

$$\text{单次查询复杂度} = \frac{n}{B} \log B + B \geq 2 \sqrt{\frac{n}{B} \log B \times B} = 2 \sqrt{n \log B}$$

基本
不等式

等号成立条件

$$\frac{n}{B} \log B = B$$

$$B^2 = n \log B$$

$$B = \sqrt{n \log B} \approx \sqrt{n \log n}$$

$$B = \text{sqrt}(n * \log_2(n));$$



现场挑战 快快编程1060

快快编程
kkcoding.net

模拟过程 观察特点	输入
	5
	1 4 2 2 1

{1,2,3,4,5}

{2,3,4,5}

{2,3,4}

{2,4}

{2}

{}

动态

删除第k小

第一步

定位第k小

第二步

移除第k小

请写出算法步骤或者难点

请思考如何设计分块算法

纯暴力步骤+分块框架

暴力纯模拟

```
8      cin>>n;
9      vector<int> v;
10     for(int i=1;i<=n;i++) v.push_back(i);
11     for(int i=1,f;i<=n;i++){
12         cin>>f;
13         ans[i]=
14         v.erase();
15     }
16     cout<<ans[n-1]<<" "<<ans[n]<<endl;
```

请写出时间复杂度
$O(n^2)$

分块算法

```
3  const int N=200009;
4  int n,m,B,a[N],b[N],ans[N];
5  vector<int> v[N];



19  cin>>n;
20  B=sqrt(n);
21  for(int i=1;i<=n;i++)b[i]=(i-1)/B+1;
22  for(int i=1;i<=n;i++)v[b[i]].push_back(i);
23  for(int i=1,f;i<=n;i++){
24      cin>>f;
25      ans[i]=submit(f);
26  }
27  cout<<ans[n-1]<<" "<<ans[n]<<endl;
```

分块算法

```
3  const int N=200009;
4  int n,m,B,a[N],b[N],ans[N];
5  vector<int> v[N];

19  cin>>n;
20  B=sqrt(n);
21  for(int i=1;i<=n;i++)b[i]=(i-1)/B+1;
22  for(int i=1;i<=n;i++)v[b[i]].push_back(i);
23  for(int i=1,f;i<=n;i++){
24      cin>>f;
25      ans[i]=submit(f);
26  }
27  cout<<ans[n-1]<<" "<<ans[n]<<endl;
```

分块算法

```
6 int submit(int f){
7     int cnt=0, j;
8     for(j=1; j<=b[n]; j++){
9         cnt+=v[j].size();
10        if(cnt>=f)
11            
12    }
13    cnt-=v[j].size();
14    int res=v[j][];
15    v[j].erase(v[j].begin()+f-cnt-1);
16    return res;
17 }
```

cnt表示从1号块到
当前块的元素总个数

动态

删除第k小

第一步

定位第k小

第二步

移除第k小

逆向思维/正难则反

利用时间倒流推理出学号信息

逆向过程 观察特点	输入 5 1 4 2 2 1
--------------	----------------------

提交的人名依次称作ABCDE

$x[5]$ 必然是1

E在最后1人里学号排第1小

$x[4]$ 为2
说明D的学号大于E的学号

E在最后2人里学号排第1小

$x[3]$ 为2
说明C的学号大于E的学号

E在最后3人里学号排第1小

$x[2]$ 为4
说明B的学号大于E的学号

E在最后4人里学号排第1小

$x[1]$ 为1
说明A的学号小于E的学号

E在最后5人里学号排第2小

E的学号已经可以推理出来

D的学号如何推理？

逆向过程 观察特点	输入
	5
	1 4 2 2 1

提交的人名依次称作ABCDE

$x[4]$ 为2

说明D的学号大于E的学号

$x[3]$ 为2

说明C的学号小于D的学号

$x[2]$ 为4

说明B的学号大于D的学号

$x[1]$ 为1

说明A的学号小于D的学号

D在最后2人里学号排第2小

D在最后3人里学号排第3小

D在最后4人里学号排第3小

D在最后5人里学号排第4小



快快编程作业

961

1058

1060

拓展题

1059