

太戈编程
etiger.vip

信奥算法

序列变种RSQ

公式推导

$s[i]$ 表示 $h[1]+h[2]+\dots+h[i]$

$g[i]$ 表示 $h[1]*1+h[2]*2+\dots+h[i]*i$

$$\begin{aligned} & h[1]*1 + h[1+1]*2 + \dots + h[r]*(r-1+1) \\ = & h[1]*1 + h[1+1]*(1+1) + \dots + h[r]*r \\ - & (h[1] + h[1+1] + \dots + h[r]) * (1-1) \\ = & g[r]-g[1-1] - (s[r]-s[1-1])*(1-1) \end{aligned}$$

```
31 void solve(){
32     for(ll i=1;i<=n;++i){
33         s[i]=s[i-1]+h[i];
34         g[i]=g[i-1]+h[i]*i;
35     }
36     for(ll i=1;i<=m;++i){
37         ll l,r;
38         cin>>l>>r;
39         ll ans=
40         cout<<ans<<" ";
41     }
42 }
```

快快编程2665

序列切割/分段

动态规划

```
1 /*
2  $f[i]$  表示前 $i$ 座楼房分组后不关闭的最小高度总和
3  $n=6, L=3$ 
4      $i=$     1,    2,    3,    4,    5,    6
5  $h[i]=$     2     2     5     4     5     1
6  $f[i]=$ 




```

序列切割的典型决策


最后一段有多长

前一段结尾在哪里

识别此题特性	选择分段长度为1或L即可	典型最优解 简化形态
	使用其他长度并不会更优	

```
f[i] = min{
    f[i-1]+h[i],
    f[i-L]+sum(i-L+1,i)-rmq(i-L+1,i)
}
```



```
27 cin>>n>>L;
28 for(ll i=1;i<=n;++i)cin>>h[i];
29 for(ll i=1;i<=n;++i)s[i]=s[i-1]+h[i];
30 RMQwL();
31 for(ll i=1;i<=n;++i){
32     f[i]=f[i-1]+h[i];
33     
34     f[i]=min(f[i], f[i-L]+s[i]-s[i-L]-bst[i]);
35 }
36 cout<<f[n]<<endl;
```

讨论：哪几行
可以改变顺序

```
13 11 bst[N];
14 11 q[N];
15 void RMQwL(){
16     11 l=0,r=0;
17     for(11 i=1;i<=n;++i){
18         
19         
20         
21         
22     }
23 }
```

快快编程2122

序列切割/分段

动态规划

$f[i][x]$ 表示第 i 楼安排 x 且第 $i+1$ 楼不安排 x 时 前 i 楼最小费用	$x=0,1$
---	---------

$i=1,2,\dots,n-1$

答案如何表示?

```

1  /*
2  f[i][x]表示第i楼安排x且第i+1楼不安排x时，前i楼最小费用。
3  答案统计：枚举最后一段起始位置和安排
4  n=4
5      i=  1,   2,   3,   4
6  0: t[i]=  4   0   1   2
7  1: p[i]=  5   2   3   1
8  f[i][0]= 



 -
9  f[i][1]= 



 -
10 ans=7

```

决策
状态转移

$f[i][x]$ 表示第 <i>i</i> 楼安排 <i>x</i> 且第 <i>i+1</i> 楼不安排 <i>x</i> 时 前 <i>i</i> 楼最小费用	$x=0,1$
--	---------

序列切割的典型决策

最后一段有多长

前一段结尾在哪里

识别此题特性	最左段：只能往右走
	中间段：分半就近走
	最右段：只能往左走

请写出几个函数的含义

```
62 for(ll i=1;i<=n-1;++i)
63     for(ll x=0;x<=1;++x){
64         f[i][x]=INF;
65         if(i<n)
66             upd(f[i][x],moveRight(1,i,x^1));
67         for(ll j=1;j<=i-1;++j)
68             upd(f[i][x],f[j][x^1]+cost(j+1,i,x^1));
69     }
70 ll ans=INF;
71 for(ll i=1;i<=n-1;++i)
72     for(ll x=0;x<=1;++x)
73         upd(ans,f[i][x]+moveLeft(i+1,n,x));
74 cout<<ans<<endl;
```

两部分循环能否合二为一?

快快编程2529

集合



排序

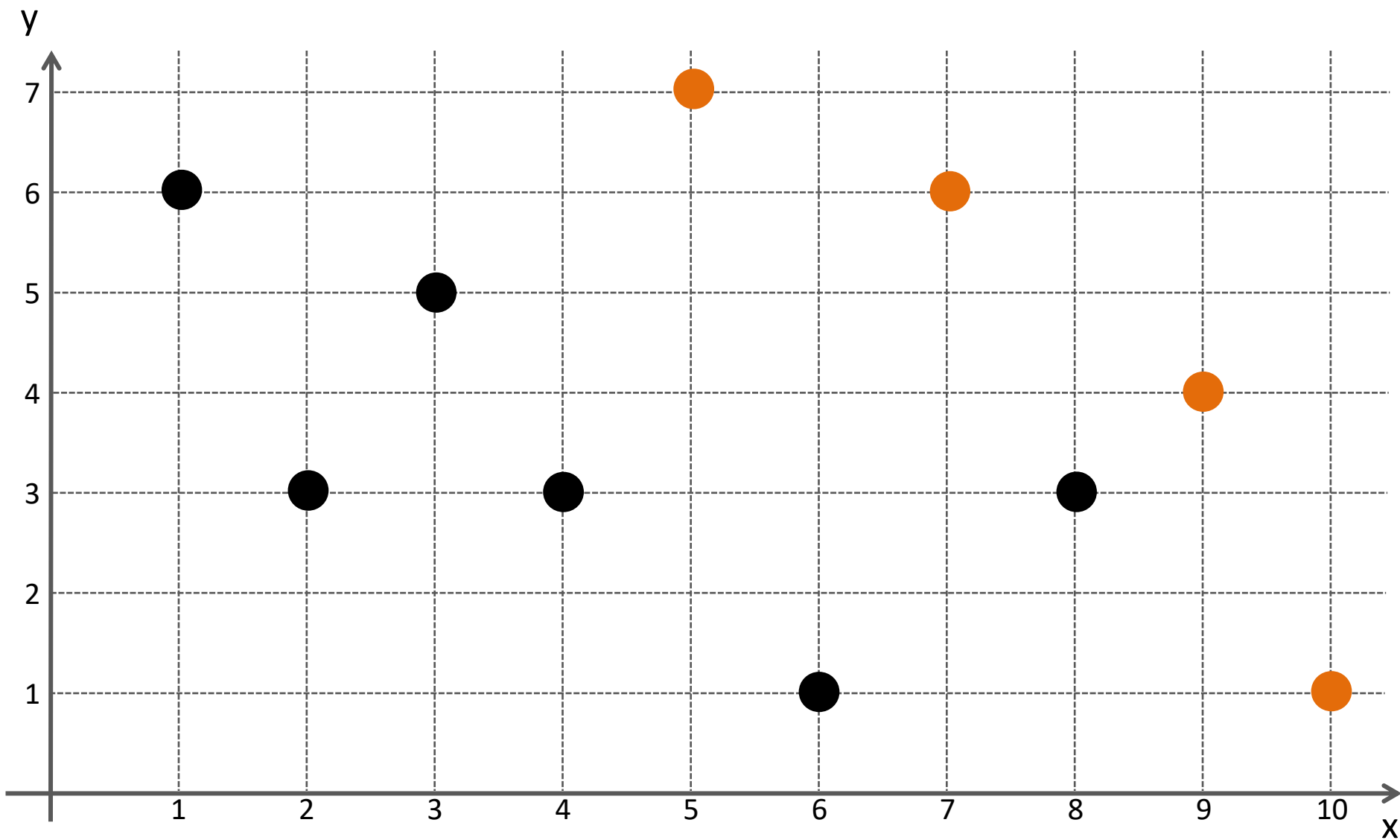
序列

剔除
无效
元素

集合
选分组

序列
选分段

有效点形成
右上方轮廓



去除无效书籍

排序：x从小到大，若x相同y从小到大

排序后依次查看：
若当前书y不小于前一本就覆盖前一本

```
25 for(ll i=1;i<=n;++i)
26     cin>>bk[i].h>>bk[i].w;
27 sort(bk+1,bk+1+n,cmp);
28 ll m=0;
29 for(ll i=1;i<=n;++i){
30     while
31         --m;
32     bk[++m]=bk[i];
33 }
```

```

1  /*
2  f[i]表示前i本书最小的占据面积
3
4  1 100
5  15 15
6  20 5
7  100 1
8      i= 1, 2, 3, 4
9  f[i]=
10
11 f[i]=min{f[j]+w[j+1]*h[i]}
12 j号直线:  $y=f[j]+w[j+1]*x$ 
13 截距b=
14 斜率k=
15 计算f[i]时取点横坐标h[i], 递增
16 */

```

```
25 ld X(ll u,ll v){
26     return -(ld)(lines[u].b-lines[v].b)
27         /(lines[u].k-lines[v].k);
28 }
```

```
44 lines[0]=(Line){0,bk[1].w};
45 q[1]=0;
46 ll l=1,r=2;
47 for(ll i=1;i<=m;++i){
48     while( )++l;
49     ll j=q[1];
50     f[i]= ;
51     lines[i]= ;
52     while( )--r;
53     q[r++]=i;
54 }
```



太戈编程

2665, 2122, 2629