

太戈编程
etiger.vip

信奥算法

太戈编程2687

暴力模拟：数组平移

```
9   for(int i=1;i<=n;i++)f[i]=i;
10  int y;
11  for(int i=1;i<=m;i++){
12      scanf("%d",&y);
13      printf("%d ",f[y]);
14      
15      for(int j=y;j<=n;j++)
16          f[j]=f[j+1];
17  }
```

暴力模拟为什么慢?

整体平移浪费时间

思考不平移的解法

存在标签**01**序列
+
额外身份列表

位置编号
tag[i]=
身份编号

i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8
0	0	1	0	1	0	1	0
-	-	3	-	1	-	2	-

输入

3 5

2 1 2 3 2

存在标签01序列

身份列表

```
3 const int N=300009;  
4 bool tag[N];  
5 vector<int> v;
```

注意三个不同序列概念
位置编号序列
身份编号序列
存在标签01序列


```
14 for(int iQ=1;iQ<=m;iQ++){
15     scanf("%d",&y);
16     int cnt=0;
17     int id=0;
18     while(cnt<y)
19         cnt+=
20     tag[id]=0;
21
22     if(id<=n) ans=id;
23     else
24
25     printf("%d ",ans);
26 }
```

存在标签**01**序列
+
额外身份列表

BIT求**01**序列前缀和
二分定位

请设计**bit[i]**含义

`bit[i]`表示在位置编号区间
`[i-LSB(i)+1,i]`内存在几个人

33

```
mx=n+m;
```

34

```
for(int i=1;i<=n;i++)add(i,1);
```

能否改成
`bit[i]=LSB(i)`

```
35 for(int iQ=1;iQ<=m;iQ++){
36     scanf("%d",&y);
37     int cnt=0;
38     int id=search(y);
39     add(id,-1);
40     add(n+iQ,1);
41     if(id<=n) ans=id;
42     else ans=v[id-n-1];
43     v.push_back(ans);
44     printf("%d ",ans);
45 }
```

请用最优化问题形式
描述定位问题

```
16 int search(int y){  
17     int l=1;  
18     int r=mx;  
19     int res=mx;  
20     while(l<=r){  
21         int mid=l+(r-l)/2;  
22         if   
23             res=mid,r=mid-1;  
24         else  
25             l=mid+1;  
26     }  
27     return res;  
28 }
```

太戈编程989

模拟问题

核心操作：发牌，销牌

类似约瑟夫问题

扑克牌排一圈

销牌 r 张

牌不动，游标动 r 步

剩余牌数在减少

请设计 `bit[i]` 含义

```
23 int p=1;
24 for(int i=0;i<n;i++){
25     int m=n-i;
26     p+=r[i]; p%=m;
27     
28     int id=
29     add(id,1);
30     printf("%d\n",id);
31 }
```


太戈编程472

请识别难点

一维升二维：
空间要共享,行列互动

```

3 typedef long long ll;
4 const ll N=300009;
5 ll n,m,nQ,mx;
6 struct Query{ll x,y;} q[N];
7 vector<ll> v[N];
8 vector<ll> xq[N];
9 ll bit[N*2];
10 ll clmn[N];

```

询问数组

额外身份列表

询问行号分类列表

存在标签01序列BIT

clmn[iQ]表示iQ号询问的人对应的原始列号

```
36 for(11 i=1;i<=nQ;++i){
37     scanf("%11d %11d",&q[i].x,&q[i].y);
38     if(q[i].y==m)continue;
39     
40 }
41 mx=max(n,m)+nQ;
42 for(11 i=1;i<=mx;++i) 
```

```
43 for(ll x=1;x<=n;++x){
44     for(ll k=0;k<xq[x].size();++k){
45         ll iQ=xq[x][k];
46         clmn[iQ]=search(q[iQ].y);
47         add(clmn[iQ],-1);
48     }
49
50
51
52
53 }
```

```
55 for(ll iQ=1;iQ<=nQ;++iQ){
56     ll row=search(q[iQ].x);
57     add(row,-1);
58     if(row<=n) ans=row*m;
59     else ans=v[0][row-n-1];
60     if(q[iQ].y!=m){
61         v[q[iQ].x].push_back(ans);
62         if(clmn[iQ]<m)
63             ans=(q[iQ].x-1)*m+clmn[iQ];
64         else
65             ans=v[q[iQ].x][clmn[iQ]-m];
66     }
67     v[0].push_back(ans);
68     printf("%lld\n",ans);
69 }
```



太戈编程

2687,989

拓展题472