

信奥算法

蠕动区间

尺取法

双游标

区间=连续段



区间右端
先向右前进



区间左端
再向右前进

练功速成

要练成一门武林神功需要连续几天闭关累积 m 小时的刻苦修炼，对于学业繁忙的同学们只有最近 n 天可以考虑，第 i 天可以修炼 $x[i]$ 小时。请问最少需要安排几天闭关修炼？
输入 n, m 和各个 $x[i]$ ，数据都小于100。
输出天数。保证有办法能练成

输入样例

8 10

4 1 1 4 4 4 1 4

输出样例

3

输入样例

7 15

5 1 3 5 10 7 4

输出样例

2

区间在哪里？

暴力枚举会这么做？

可以怎样优化？

总和达标，最小区间

有 n 个正整数排成一排，需要取出连续若干个求和，总和不小于 m 。请问最少要取几个数？

输入 n, m 和各个 $x[i]$ ，数据都小于100。

输出个数。保证有解

输入样例

8 10

4 1 1 4 4 4 1 4

输出样例

3

输入样例

7 15

5 1 3 5 10 7 4

输出样例

2

区间在哪里？

暴力枚举会这么做？

可以怎样优化？

暴力枚举

sum代表x数组编号为区间[tail,head]内的数值总和

```
8  int ans=n+1;
9  for(int tail=0;tail<n;tail++){
10     int sum=0;
11     for(int head=tail;head<n;head++){
12         sum+=x[head];
13         if(sum>=m) {
14             ans=min(ans,head+1-tail);
15             break;
16         }
17     }
18 }
19 cout<<ans<<endl;
```

枚举区间尾部位置

枚举区间头部位置

累加连续和

若连续和满足条件

更新最优长度

头部不再前进

时间复杂度 $O(n^2)$

总和至少15

枚举法演示

sum = 24

5	1	3	5	10	7	4
---	---	---	---	----	---	---

sum = 19

5	1	3	5	10	7	4
---	---	---	---	----	---	---

sum = 18

5	1	3	5	10	7	4
---	---	---	---	----	---	---

sum = 15

5	1	3	5	10	7	4
---	---	---	---	----	---	---

sum = 17

5	1	3	5	10	7	4
---	---	---	---	----	---	---

sum = 11

5	1	3	5	10	7	4
---	---	---	---	----	---	---

sum = 4

5	1	3	5	10	7	4
---	---	---	---	----	---	---

总和至少15

枚举法演示

sum = 5
sum = 6
sum = 9
sum = 14
sum = 24
sum = 19
sum = 18
sum = 15
sum = 10
sum = 17
sum = 7
sum = 11

5	1	3	5	10	7	4
5	1	3	5	10	7	4
5	1	3	5	10	7	4
5	1	3	5	10	7	4
5	1	3	5	10	7	4
5	1	3	5	10	7	4
5	1	3	5	10	7	4
5	1	3	5	10	7	4
5	1	3	5	10	7	4
5	1	3	5	10	7	4
5	1	3	5	10	7	4
5	1	3	5	10	7	4
5	1	3	5	10	7	4
5	1	3	5	10	7	4
5	1	3	5	10	7	4

右端将前进

右端将前进

右端将前进

右端将前进

左端将前进

左端将前进

左端将前进

左端将前进

右端将前进

左端将前进

右端将前进

观察区间的
变化过程

请同学总结
蠕动规律

5	1	3	5	10	7	4
5	1	3	5	10	7	4
5	1	3	5	10	7	4
5	1	3	5	10	7	4
5	1	3	5	10	7	4
5	1	3	5	10	7	4
5	1	3	5	10	7	4
5	1	3	5	10	7	4
5	1	3	5	10	7	4
5	1	3	5	10	7	4
5	1	3	5	10	7	4
5	1	3	5	10	7	4



区间右端
先向右前进

右端右移
直到区间总和达标



区间左端
再向右前进

左端右移
直到区间总和不达标

蠕动区间

sum代表x数组编号为区间[tail, head-1]内的数值总和

```
8   int sum=0, ans=n+1;  
9   int tail=0, head=0;
```

tail==head时，对应区间为空，不包含任何元素

由于本题是最小化问题，初始化答案尽量大ans=n+1
之后会逐渐减小

蠕动区间

sum代表x数组编号为区间[tail, head-1]内的数值总和

```
8      int sum=0, ans=n+1;
9      int tail=0, head=0;
10     while(1){
11         while(head<n&&sum<m)
12             sum+=x[head++];
13         if(sum<m) break;
14         ans=min(ans, head-tail);
15         sum-=x[tail++];
16     }
17     cout<<ans<<endl;
```

当head没越界且总和不够

累加连续和 **head前进**

若head到顶后总和还不够就结束

更新最优长度

扣除连续和 **tail前进**

快快编程
kkcoding.net

滑动区间

```
8   int sum=0,ans=n+1;
9   int tail=0,head=0;
10  while(1){
11      while(head<n&&sum<m)
12          sum+=x[head++];
13      if(sum<m)break;
14      ans=min(ans,head-tail);
15      sum-=x[tail++];
16  }
17  cout<<ans<<endl;
```

时间复杂度

$O(n)$

因为每个元素
最多进入区间1次
最多离开区间1次

滑动区间-易错点

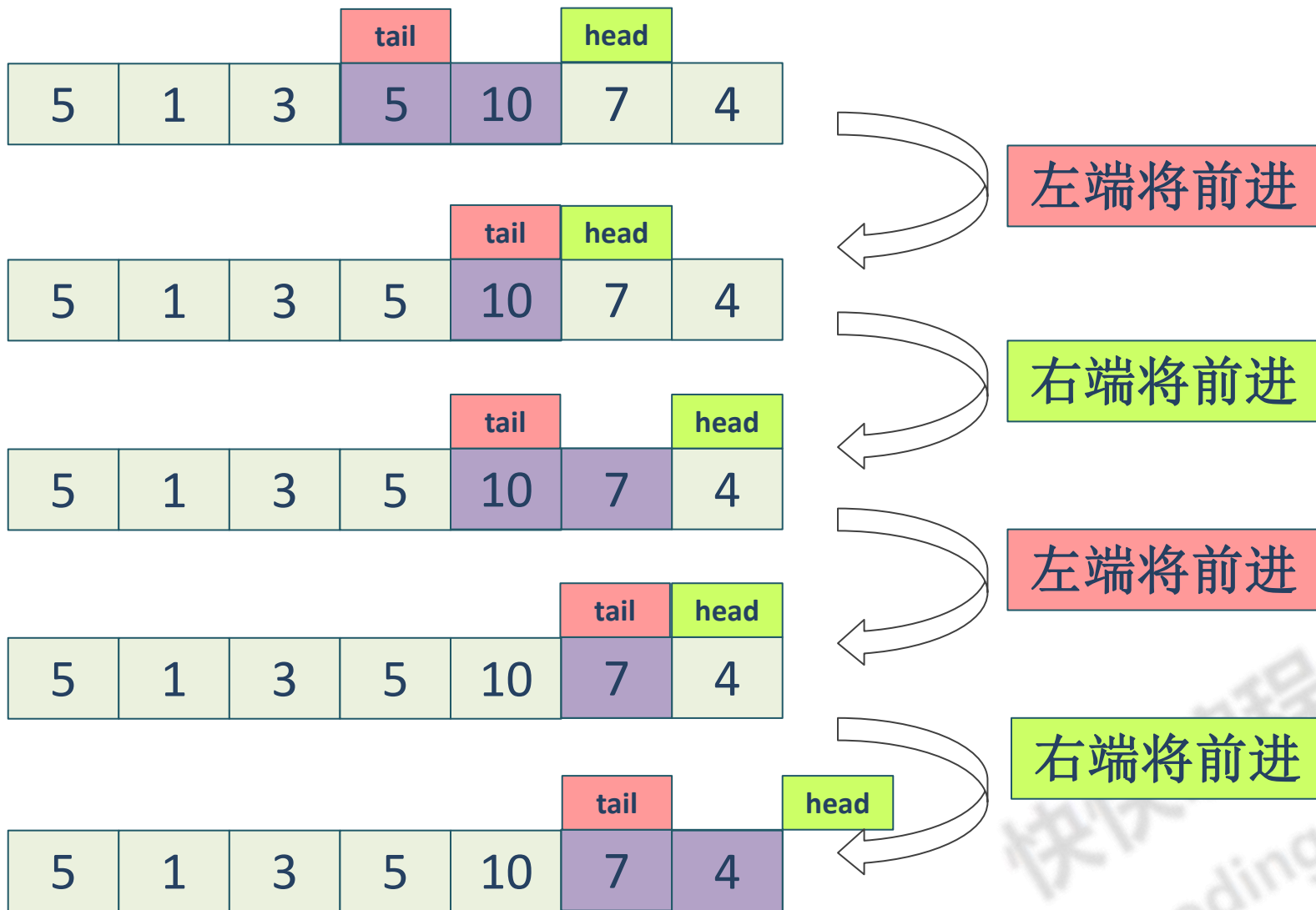
```
8   int sum=0,ans=n+1; ←
9   int tail=0,head=0; ←
10  while(1){
11      while(head<n&&sum<m)
12          sum+=x[head++];
13      if(sum<m)break; ←
14      ans=min(ans,head-tail);
15      sum-=x[tail++]; ←
16  }
17  cout<<ans<<endl;
```

时间复杂度
 $O(n)$

总和至少15

双游标位置

右端点head
始终在区间外面



错误代码1

```
8  int sum=0,ans=n+1;
9  int tail=0,head=0;
10 while(1){
11     while(sum<m) //头部移动
12         sum+=x[head++];
13     if(sum<m)break; //终止条件
14     ans=min(ans,head-tail); //更新最优解
15     sum-=x[tail++]; //尾部移动
16 }
17 cout<<ans<<endl;
```

错在哪里？
为什么错？

错误代码2

```
10 while(1){  
11     while(head<n&&sum<m) //头部移动  
12         sum+=x[head++];  
13     if(head>=n)break; //终止条件  
14     ans=min(ans,head-tail); //更新最优解  
15     sum-=x[tail++]; //尾部移动  
16 }
```

错在哪里？
为什么错？

收集三原色

共 n 个宝石排成一排，你可以买到任意连续段的若干宝石。你希望收集到三原色的宝石：red,green,blue。请问最少需要买几个宝石？ $n \leq 100$ ，保证有解

输入样例

6

yellow red blue black green blue

输出样例

4

区间在哪里？

区间=连续段

“连续”是关键词

如何控制蠕动区间？

右端右移
直到收集齐三原色

____端__移
直到_____

0	1	2	3	4	5	6	7	8	9	越界
y	r	g	y	b	b	g	r	y	g	

右端右移
直到收集齐三原色

左端右移
直到没集齐三原色

如何判断是否集齐三原色？

sum记录在当前区间里3原色共有几种

数据结构

```
string x[N];  
map<string,int> cnt;  
  
set<string> rgb;  
rgb.insert("red");  
rgb.insert("green");  
rgb.insert("blue");
```

`x[i]`代表序列中*i*号颜色

`cnt`映射记录每种3原色在当前区间里出现了几次

`rgb`容器里只储存3原色

`sum`记录在当前区间里3原色共有几种

收集三原色

```
14 int sum=0,ans=n+1;
15 int tail=0,head=0;
16 while(1){
17     while(head<n&&sum<3) {
18         string color=x[head++];
19         if(!rgb.count(color))continue;
20         ++cnt[color];
21         if(cnt[color]==1)sum++;
22     }
23     if(sum<3)break;
24     ans=min(ans,head-tail);
25     string color=x[tail++];
26     if(!rgb.count(color))continue;
27     --cnt[color];
28     if(cnt[color]==0)sum--;
29 }
30 cout<<ans<<endl;
```

当head没越界且宝石不够

head前进

非三原色就忽略

累加三原色
宝石数量

若宝石不够就结束枚举

更新最优长度

tail前进

非三原色就忽略

扣除三原色
宝石数量

易错点

```
17 int sum=0,ans=n+1;
18 int tail=0,head=0;
19 while(1){
20     while(head<n&&sum<3) {
21         string color=x[head++];
22         if(!rgb.count(color))continue;
23         if(++cnt[color]==1)sum++;
24     }
25     if(sum<3)break;
26     ans=min(ans,head-tail);
27     string color=x[tail++];
28     if(!rgb.count(color))continue;
29     if(--cnt[color]==0)sum--;
30 }
31 cout<<ans<<endl;
```

现场挑战

快快编程490

快快编程
kkcoding.net

思路分析

思考收集七龙珠问题
和收集三原色问题的关联性

收集七龙珠

在序列中求最短的区间
包含7种目标字符串

收集三原色

在序列中求最短的区间
包含3种目标字符串

现场挑战

快快编程491

快快编程
kkcoding.net

思路分析

思考收集国家问题
和收集三原色问题的关联性

收集国家

在序列中求最短的区间
包含 k 种不同数字

所有数字都是目标数字

收集齐的条件用 k 描述

收集三原色

在序列中求最短的区间
包含3种目标字符串

现场挑战

快快编程492

快快编程
kkcoding.net

思路分析

思考492问题
和491问题的关联性

492

在序列中求最短的区间
包含所有不同数字

用k统计不同数字的总个数

问题转换为491

491

在序列中求最短的区间
包含k种不同数字

快快编程作业

490

491

492

拓展题

170