

CS102

A Mars rover, likely a Curiosity rover, is shown on a rocky, orange-hued landscape. The rover is positioned in the center-left of the frame, facing right. It has six large, treaded wheels and a complex body with various instruments and cameras. The background shows a hazy, orange sky and distant, low hills. The overall scene is a typical Mars surface environment.

C++
算法

蠕动区间

枚举法求最优连续段

尺取法

双游标

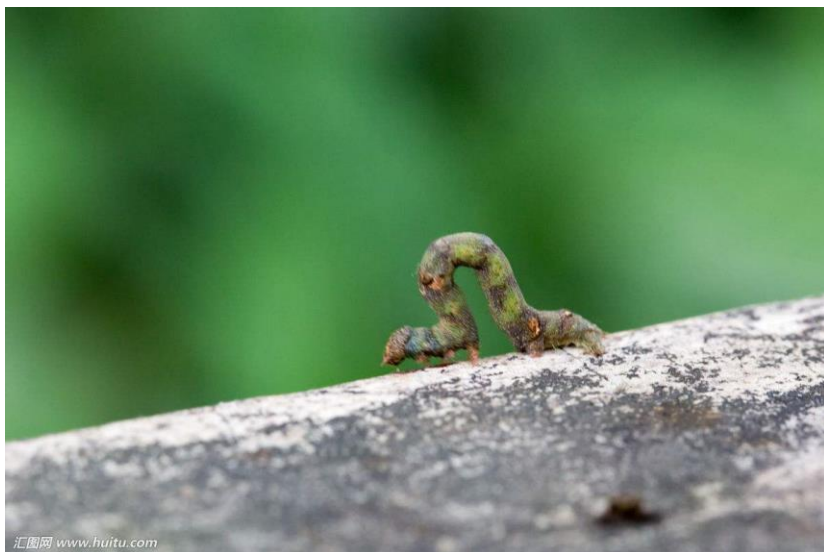
蠕动区间



蠕动区间



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

输入样例

7 15

5 1 3 5 10 7 4

输出样例

3

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

输入样例

7 15

5 1 3 5 10 7 4

输出样例

3

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

```
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
 $O(n^2)$

暴力枚举演示：至少15

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

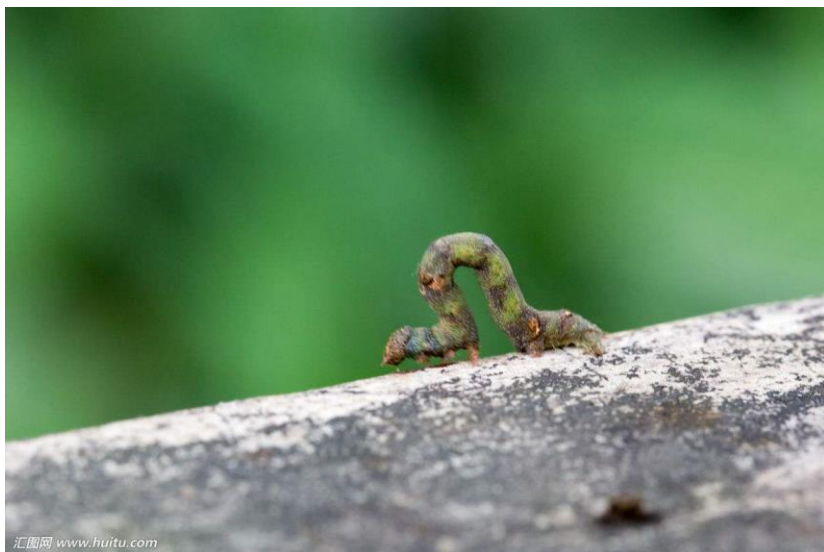
改进枚举过程

sum = 5	5	1	3	5	10	7	4	右端将前进
sum = 6	5	1	3	5	10	7	4	右端将前进
sum = 9	5	1	3	5	10	7	4	右端将前进
sum = 14	5	1	3	5	10	7	4	右端将前进
sum = 24	5	1	3	5	10	7	4	左端将前进
sum = 19	5	1	3	5	10	7	4	左端将前进
sum = 18	5	1	3	5	10	7	4	左端将前进
sum = 15	5	1	3	5	10	7	4	左端将前进
sum = 10	5	1	3	5	10	7	4	右端将前进
sum = 17	5	1	3	5	10	7	4	左端将前进
sum = 17	5	1	3	5	10	7	4	右端将前进
sum = 11	5	1	3	5	10	7	4	

蠕动区间



区间右端
先向右前进



区间左端
再向右前进

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;
```

当head没越界且总和不够

累加连续和 **head前进**

若总和不够就结束枚举

更新最优长度

扣除连续和 **tail前进**

时间复杂度
 $O(n)$

双游标位置



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？

如何控制蠕动区间？

收集三原色

```
14  rgb.insert("red");
15  rgb.insert("green");
16  rgb.insert("blue");
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;
```

当head没越界且宝石不够

head前进

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

扣除三原色
宝石数量