



信奥算法

今朝有酒
今朝醉

没有远见
活在当下

贪心法

greedy algorithm

得过且过

鼠目寸光

做一天和尚
撞一天钟

提款取钱

你是一个银行小职员，你的日常工作就是根据客户要求，从银行里为他们提取特定数额的纸币。已知客户需要提取 n 元，你需要用最少的纸币数量凑出 n 元，请问最少要几张？ $n \leq 10000$

国家只有以下面值的纸币：1元，2元，5元，10元，20元，50元，100元

提款取钱

贪心
算法

每次选

的纸币，尽量多取

把纸币按照面额从大到小排序

讨论题：为什么
该贪心法正确？

```
1  #include<bits/stdc++.h>
2  using namespace std;
3  const int N=7;
4  int money[N]={100,50,20,10,5,2,1};
5  int main(){
6      int n,ans=0;
7      cin>>n;
8      for(int i=0;i<N;i++){
9          ans+=
10         n%=money[i];
11     }
12     cout<<ans<<endl;
13     return 0;
14 }
```

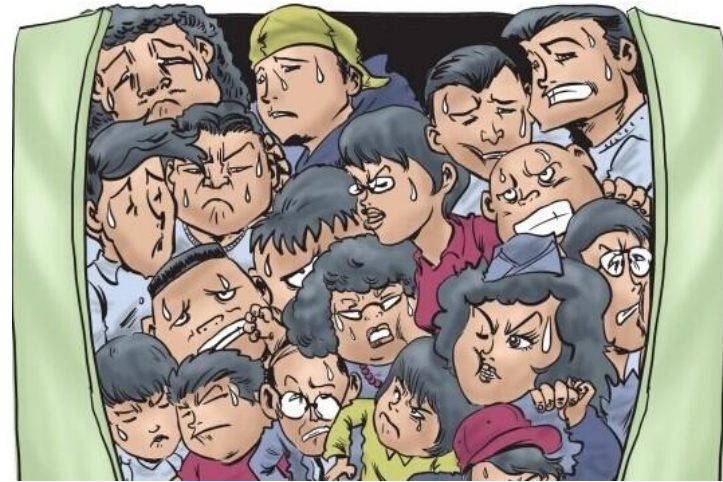
解释每个变量的含义

补全
程序

超重

共 n 个人排队依次进电梯，第 i 个人体重为 w_i ，只可以安排相邻若干个人进入。每台电梯限重 a ，保证每台不超重，求最少要分几次？如无法完成输出-1。

第一行输入 n, a 。第二行包含 n 个数字代表每人
体重。 $n \leq 1000, w_i \leq 100, a \leq 1000$



输入样例：

5 60

10 10 50 20 50

输出样例：

4

10

10

50

20

50

输入样例：

6 110

10 10 50 60 50 60

输出样例：

3

10

10

50

60

50

60

超重

贪心
算法

依次进电梯，尽量多安排人

如果超重了，就安排下一次

无解
情况

如果某人的单人体重就超过限重，那么无解

讨论题：为什么
该贪心法正确？

解释每个变量的含义

```
6  cin>>n>>a;
7  for(int i=0;i<n;i++)cin>>w[i];
8  for(int i=0;i<n;i++)
9      if( ) {
10         cout<<-1<<endl;
11         return 0;
12     }
13  int ans=1,sum=0;
14  for(int i=0;i<n;i++){
15     sum+=w[i];
16     if( ) {
17         ans++;
18         sum= ;
19     }
20 }
21 cout<< <<endl;
```

为什么
是1

补全
程序

快快编程
kkcoding.net

大胃王

大胃王的胃容量是100升，眼前有 n 款饮料，第 i 种饮料：总量共 x_i 升，单价 p_i 。他希望填满最多肚子的前提下，花费最少是多少。

输入第一行为 n ， $n \leq 1000$ ，之后每一行为一种饮料的总量和单价。
输出最少的花费。

输入样例：

3
50 3
100 2
50 1

输出样例：

150

共3种饮料

第1种：有50升,每升3元

第2种：有100升,每升2元

第3种：有50升,每升1元

输入样例：

2
80 1
80 5

输出样例：

180



大胃王

贪心
算法

每次选

的饮料，尽量多喝

把饮料按照单价从小到大排序

需要用struct自定义"饮料"类型drink

需要用cmp函数自定义比较规则

讨论题：为什么
该贪心法正确？

```
5 struct drink{
6     int q,p;
7 };
8 drink d[N];
9 bool cmp(const drink&a,const drink&b){
10     return a.p<b.p;
11 }
```

q代表饮料量，p代表单价

解释每个变量的含义

```
17 sort(d,d+n,cmp);
18 int ans=0,c=100;
19 for(int i=0;i<n&& c;i++){
20     int x=min(c,d[i].q);
21     ans+=;
22     c-=x;
23 }
24 cout<<ans<<endl;
```

补全
程序

监狱长

作为监狱长，你要管理罪犯的牢房位置。监狱的牢房，一字排开，编号依次为0,1,2,...,1000。这1001个牢房里只有k间是有空位的，第i个空牢房的编号是d[i]。为防止高智商罪犯联合越狱，你需要将他们安排在互相间隔较远的牢房，不允许安排两人在同一牢房。如果任意两个高智商罪犯的牢房编号差距不能小于m，请问最多能安排几个高智商罪犯？

第一行输入k,m。第二行包含k个数字代表牢房编号，从小到大。
 $k, m \leq 1000, 0 \leq d[i] \leq 1000$

输入样例：

7 3
1 2 3 4 5 6 7

输出样例：

3

3个牢房编号：1,4,7

输入样例：

5 10
0 7 8 9 20

输出样例：

2

2个牢房编号：0,20

输入样例：

5 9
0 7 8 9 20

输出样例：

3

3个牢房编号：0,9,20

监狱长

贪心
算法

编号从小到大依次考虑每个牢房
能安排就尽量安排

能安排
的条件

牢房编号差距不能小于 m

讨论题：为什么
该贪心法正确？

```
1  #include<bits/stdc++.h>
2  using namespace std;
3  const int N=1009;
4  int k,m,d[N];
5  int main(){
6      cin>>k>>m;
7      for(int i=0;i<k;i++)cin>>d[i];
8      int p=0,cnt=1;
9      for(int i=1;i<k;i++)
10         if(d[i]-d[p]>= ) {
11             cnt++;
12             p= ;
13         }
14     cout<< <<endl;
15     return 0;
16 }
```

猜出每个变量的含义

为什么
是1

补全
程序

防空导弹

敌人有 n 架轰炸机依次向你飞来，第 i 架飞机高度为 $h[i]$ 。你需要用防御塔发出防空导弹炸毁敌机。

你的防御塔有一个优点：第一枚导弹可以打中任意高度。

但是，防御塔也有缺陷：之后每次导弹高度不能比上一枚导弹高。为了击中所有敌机，请问最少要安排几个防御塔？

输入第一行为正整数 n ， $n \leq 1000$

第二行为 n 个正整数，代表飞机高度，均不超过10000

输入样例：

5

1 2 3 4 3

输出样例：

4

输入样例：

4

1 2 1 2

输出样例：

2



防空导弹

错误
贪心
算法

每次用一个防御塔
安排它摧毁最多架飞机

不断在剩余飞机高度里
挑最长的不上升序列

防空导弹

正确 贪心 算法

依次安排每架飞机
安排现有防御塔里可达高度最低的去打击
若没有防御塔可以打到，就新增一个

当前每个防御塔的信息
最重要的就是上一次打击高度是多少

$d[i]$ 记录 i 号防御塔最后一次打击高度

接龙
过程

$d[i]$ 记录 i 号防御塔最后一次打击高度
也就是该条不升子序列的最后一个数字

i	$h[i]$
0	2
1	1

$d[0]$	$d[1]$	$d[2]$	$d[3]$	$d[4]$	$d[5]$
∞	∞	∞	∞	∞	∞
2	∞	∞	∞	∞	∞
1	∞	∞	∞	∞	∞

1条不升子序列

1条不升子序列

接龙
过程

$d[i]$ 记录 i 号防御塔最后一次打击高度
也就是该条不升子序列的最后一个数字

i	$h[i]$
0	2
1	1
2	2

$d[0]$	$d[1]$	$d[2]$	$d[3]$	$d[4]$	$d[5]$
∞	∞	∞	∞	∞	∞
2	∞	∞	∞	∞	∞
1	∞	∞	∞	∞	∞
1	2	∞	∞	∞	∞

- 1条不升子序列
- 1条不升子序列
- 2条不升子序列

接龙
过程

$d[i]$ 记录 i 号防御塔最后一次打击高度

也就是该条不升子序列的最后一个数字

i	h[i]
0	2
1	1
2	2
3	7

$d[0]$	$d[1]$	$d[2]$	$d[3]$	$d[4]$	$d[5]$
∞	∞	∞	∞	∞	∞
2	∞	∞	∞	∞	∞
1	∞	∞	∞	∞	∞
1	2	∞	∞	∞	∞
1	2	7	∞	∞	∞

1条不升子序列

1条不升子序列

2条不升子序列

3条不升子序列

接龙
过程

$d[i]$ 记录 i 号防御塔最后一次打击高度
也就是该条不升子序列的最后一个数字

i	h[i]
0	2
1	1
2	2
3	7
4	2

$d[0]$	$d[1]$	$d[2]$	$d[3]$	$d[4]$	$d[5]$
∞	∞	∞	∞	∞	∞
2	∞	∞	∞	∞	∞
1	∞	∞	∞	∞	∞
1	2	∞	∞	∞	∞
1	2	7	∞	∞	∞
1	2	7	∞	∞	∞

- 1条不升子序列
- 1条不升子序列
- 2条不升子序列
- 3条不升子序列
- 3条不升子序列

接龙
过程

$d[i]$ 记录 i 号防御塔最后一次打击高度

也就是该条不升子序列的最后一个数字

i	h[i]
0	2
1	1
2	2
3	7
4	2
5	3

$d[0]$	$d[1]$	$d[2]$	$d[3]$	$d[4]$	$d[5]$
∞	∞	∞	∞	∞	∞
2	∞	∞	∞	∞	∞
1	∞	∞	∞	∞	∞
1	2	∞	∞	∞	∞
1	2	7	∞	∞	∞
1	2	7	∞	∞	∞
1	2	3	∞	∞	∞

1条不升子序列

1条不升子序列

2条不升子序列

3条不升子序列

3条不升子序列

3条不升子序列

防空导弹

```
8  int cnt=0;
9  for(int i=0,j;i<n;i++){
10      for(j=0;j<cnt;j++)
11          if(d[j]>=h[i])break;
12      d[j]=h[i];
13      if(j==cnt)cnt++;
14  }
15  cout<<cnt<<endl;
```

快快编程作业

1049

436

1050

拓展题

694,413