

# CS101

A Mars rover, likely a Curiosity rover, is shown on a rocky, reddish-brown landscape. The rover is white with various instruments and cameras. It has six large, treaded wheels. The background shows a hazy, orange sky and distant hills. The overall scene is a typical Mars surface environment.

信奥  
算法

---

课件下载地址:

<http://pan.baidu.com/s/1nu6kYkL>

# 作业01：级数求和

已知：  $S_n = 1 + 1/2 + 1/3 + \dots + 1/n$ 。显然对于任意一个整数  $K$ ，当  $n$  足够大的时候，  $S_n$  大于  $K$ 。

现给出一个整数  $K$ （ $1 \leq k \leq 15$ ），要求计算出一个最小的  $n$ ；使得  $S_n > K$ 。

# 作业01

```
1  #include<iostream>
2  using namespace std;
3  double tot;
4  int i,k;
5  int main(){
6      cin>>k;
7      tot=0;
8      for(i=1;tot<=k;i++)
9          tot+=1.0/i;
10     cout<<i-1<<endl;
11     return 0;
12 }
```

# 作业02： 收入支出查询

小明他以每天为单位把每一天的总收入或支出都记录下来，共 $n$ 天，正数代表收入，负数代表支出。有一天他老婆突然对他进行查账，提出共 $m$ 个查询问题，每个问题都是关心某两天之间的存款变了多少。

输入第一行为 $n$ 和 $m$ （ $1 \leq n, m \leq 10000$ ），第二行为 $n$ 个整数代表当天的收入或支出，第三行为 $2*m$ 个正整数，每两个数代表一次查询的起始日期和结束日期。输出一行共 $m$ 个数，代表每一个查询结果。数字由空格隔开。

# 作业02

```
1  #include<iostream>
2  #define N 10000
3  using namespace std;
4  int base,n,m,x,y,i,d[N],s[N];
5  int main(){
6      cin>>n>>m;
7      for(i=0;i<n;i++) cin>>d[i];
8      s[0]=d[0];
9      for(i=1;i<n;i++) s[i]=s[i-1]+d[i];
10     for(i=0;i<m;i++){
11         cin>>x>>y;
12         if(x==1) base=0;
13         else base=s[x-2];
14         cout<<s[y-1]-base<<" ";
15     }
16     return 0;
17 }
```

# 作业03：最大连续子序列和

输入 $n(1 \leq n \leq 100000)$ ，再依次输入 $n$ 个整数组成的数组(每个数的绝对值在100以内)，求数组中最大连续子序列和（至少包含一个数字）

# 作业03

```
1  #include<iostream>
2  #define N 100005
3  using namespace std;
4  int tot,ans,n,i,x[N];
5  int main(){
6      cin>>n;
7      for(i=0;i<n;i++) cin>>x[i];
8      tot=ans=x[0];
9      for(i=1;i<n;i++){
10         tot=max(tot+x[i],x[i]);
11         ans=max(tot,ans);
12     }
13     cout<<ans<<endl;
14     return 0;
15 }
```



# 贪心算法

Greedy Algorithm

# 纸币凑和

有1元，5元，10元，20元，50元，100元纸币的数量分别为 $x_1, x_5, x_{10}, x_{20}, x_{50}, x_{100}$ 张。现需要支付 $n$ 元，求至少要几张纸币？（本题确保能够支付）

输入样例

66

2 2 2 2 2 2

输入样例

88

3 3 3 3 3 3

输出样例

4

输出样例

7

# 纸币凑和

正确的贪心算法：

不断循环重复：

尽量用大面值的纸币

```
1  #include<iostream>
2  using namespace std;
3  int ans,i,n,c[6],v[6]={1,5,10,20,50,100};
4  int main(){
5      cin>>n;
6      for(i=0;i<6;i++) cin>>c[i];
7      for(i=5;i>=0;i--) {
8          ans+=min(n/v[i],c[i]);
9          n%=v[i];
10     }
11     cout<<ans<<endl;
12     return 0;
13 }
```

# 大胃王

大胃王的胃容量是100升，眼前有3款饮料，每种都有对应的总量和单价：

饮料1：共 $x_1$ 升，单价 $p_1$

饮料2：共 $x_2$ 升，单价 $p_2$

饮料3：共 $x_3$ 升，单价 $p_3$

大胃王希望填满最多的肚子的前提下，花费最少是多少

输入样例

50 3

100 2

50 1

输入样例

50 1

20 2

20 3

输出样例

150

输出样例

150

# 大胃王

正确的贪心算法：

不断循环重复：

尽量喝便宜的饮料

可以先对饮料按照价格排序

# 活动选择： 不相交区间

电影节上有 $n$ 项电影放映活动可以观看，第 $i$ 部电影分别从时刻 $s_i$ 开始，时刻 $t_i$ 结束。最多可以完整观看几部电影？（不允许结束时刻和开始时刻重叠， $n \leq 100000$ ）

输入样例

2  
1 0  
4 1

输出样例

1

输入样例

5  
1 2 4 6 8  
3 5 7 9 10

输出样例

3

# 活动选择： 不相交区间

错误的贪心算法：

不断循环重复选择活动：

每次在可选活动中，选择**用时最少**的活动

能否举出反例？



# 活动选择： 不相交区间

正确的贪心算法：

不断循环重复选择活动：

每次在可选活动中，选择**结束最早**的活动

可以先对活动按照结束时间排序

# 哨兵站岗：点覆盖区间

直线上有 $n$ 个点，点 $i$ 的位置是 $x_i$ 。从中选出若干个点标记为哨兵位置，如果要求每个点的距离 $r$ 范围内至少有一个哨兵，请问最少要安排多少个哨兵？

$x_i, r, n \leq 100000$

输入样例

6 10

20 30 50 1 7 15

输入样例

3 5

0 10 4 6

输出样例

3

输出样例

2

# 哨兵站岗：点覆盖区间

正确的贪心算法：

1. 将 $n$ 个点从左到右排序
2. 从左到右依次覆盖每个点，安排哨兵尽量靠右

# 删数问题

输入一个正整数 $n(n \leq 10^9)$ ，去掉其中任意 $x$ 个数字后，使得剩下的数字最小为多少？

输入样例

178543

4

输入样例

213

1

输出样例

13

输出样例

13

# 删数问题

**错误的贪心算法：**

为了使剩下的数字最小，在n中去掉其中最大的x个数字

输入样例

213

1

输出样例

13

# 删数问题

正确的贪心算法：

为了使剩下的数字最小，循环x次：  
每次找到第一个出现的下降起点

178543

213

17543

1543

143

# 讨论题：数字合并

有 $n$ 个正整数，现在进行若干次操作：每次删去2个数 $a$ 和 $b$ ，然后加入1个数 $a*b+1$ 。反复操作直到只有一个数，求最小剩下几？（ $n \leq 1000$ ）

输入样例

3

1 2 3

输入样例

6

8 6 5 9 7 1

输出样例

8

输出样例

15367

思考题：此问题使用  
“贪心算法”是啥步骤？

# 讨论题：木条切割

有一根长木条能恰好被切割成 $n$ 块。这 $n$ 块的长度是 $L_1, L_2, L_3, \dots, L_n$ 。开始时，木条总长为 $(L_1 + L_2 + L_3 + \dots + L_n)$ 。每次切断一段木条时，费用为这段木条的长度。如果按照要求切割完成，最少需要多少费用？

输入样例

3

8 8 5

输入样例

5

1 2 3 4 5

输出样例

34

思考题：此问题使用  
“贪心算法”是啥步骤？



# 作业

---

作业网站:

<http://120.132.20.20/thrall-web/main#home>