

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-tinted sky and distant hills. The overall scene is a typical Mars surface environment.

信奥
算法

课件下载地址:

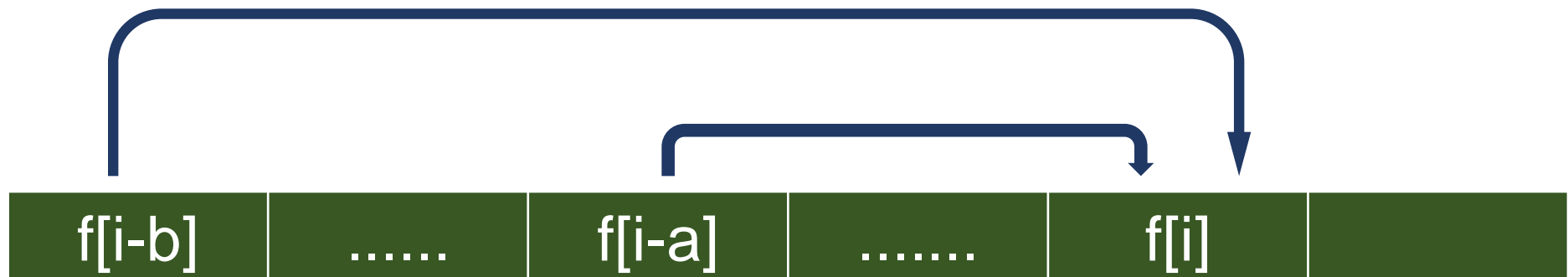
<http://pan.baidu.com/s/1o885tz0>

作业网站:

<http://120.132.18.213:8080/thrall-web/main#home>

作业1：魔鬼的步伐

$f[i]$ 代表走到第*i*级共有多少种方法



作业1：魔鬼的步伐

$f[i]$ 代表走到第*i*级共有多少种方法

初始
条件

当*i*为0时

$$f[0] = 1$$

递推
方程

当*i*大于0时

$$f[i] = f[i - a] | i \geq a \\ + f[i - b] | i \geq b$$

若 $i \geq a$ ，可走a级来

若 $i \geq b$ ，可走b级来

作业1：魔鬼的步伐

```
1 #include<iostream>
2 using namespace std;
3 long long i,n,a,b,f[51];
4 int main(){
5     cin>>n>>a>>b;
6     f[0]=1;
7     for(i=1;i<=n;i++){
8         f[i]=0;
9         if(i>=a) f[i]+=f[i-a];
10        if(i>=b) f[i]+=f[i-b];
11    }
12    cout<<f[n]<<endl;
13    return 0;
14 }
```

作业2：盗墓 解法1

从第1件，第2件，一件一件拿

$f[i]$ 代表拿走第*i*件时，前*i*件最多拿走多少钱

初始
条件

$i \leq 3$ 时

$$f[i] = x[i]$$

递推
方程

$i > 3$ 时

$$f[i] = \max(f[i-1], x[i] + \max_{j=1,2,\dots,i-3}\{f[j]\})$$

```
1  #include<iostream>
2  using namespace std;
3  int i,j,n,ans,f[101],x[101];
4  int main(){
5      cin>>n;
6      for(i=1;i<=n;i++) cin>>x[i];
7      f[0]=0; f[1]=x[1]; f[2]=x[2]; f[3]=x[3];
8      for(i=4;i<=n;i++){
9          int big=f[0];
10         for(j=1;j<=i-3;j++)
11             big=max(big,f[j]);
12         f[i]=max(f[i-1],big+x[i]);
13     }
14     ans=f[0];
15     for(i=1;i<=n;i++)
16         ans=max(ans,f[i]);
17     cout<<ans<<endl;
18     return 0;
19 }
```

作业2： 盗墓 解法2

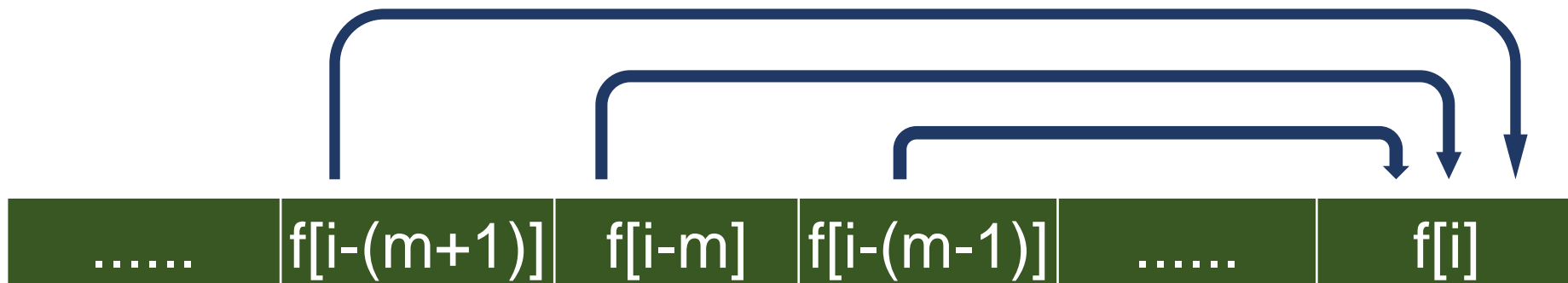
$g[i]$ 代表 前*i*件最多拿走多少钱

初始 条件	i=0时	$g[0] = 0$
	i=1时	$g[1] = x[1]$
	i=2时	$g[2] = \max(x[1], x[2])$
递推 方程	i>3时	$g[i] = \max(g[i - 1], x[i] + g[i - 3])$


```
1 #include<iostream>
2 using namespace std;
3 long long i,n,g[101],x[101];
4 int main(){
5     cin>>n;
6     for(i=1;i<=n;i++) cin>>x[i];
7     g[0]=0;
8     g[1]=x[1];
9     g[2]=max(x[1],x[2]);
10    g[3]=max(x[1],x[2],x[3]);
11    for(i=4;i<=n;i++)
12        g[i]=max(g[i-1],g[i-3]+x[i]);
13    cout<<g[n]<<endl;
14    return 0;
15 }
```

作业3：捡铜币

$f[i]$ 代表 走到第 i 格时最多拿走多少铜币



作业3：捡铜币

$f[i]$ 代表 走到第*i*格时最多拿走多少铜币

初始
条件

$i=1$ 时

$$f[1] = x[1]$$

递推
方程

$i>1$ 时

$$\begin{aligned} f[i] = & x[i] \\ & + \max(f[i - (m - 1)] | i > m - 1, \\ & \quad f[i - m] | i > m, \\ & \quad f[i - (m + 1)] | i > m + 1) \end{aligned}$$

作业3：捡铜币

$f[i]$ 代表 走到第*i*格时最多拿走多少铜币

初始
条件

$i=1$ 时

$$f[1] = x[1]$$

递推
方程

$i>1$ 时

$$f[i] = x[i] + \max(f[i - (m - 1)] | i > m - 1, f[i - m] | i > m, f[i - (m + 1)] | i > m + 1)$$

最优
答案

$$ans = \max\{f[1], f[2], f[3], \dots, f[n - 1], f[n]\}$$

```
1 #include<iostream>
2 using namespace std;
3 int i,n,m,ans,f[1001],x[1001];
4 int main(){
5     cin>>n>>m;
6     for(i=1;i<=n;i++) cin>>x[i];
7     f[1]=x[1];
8     for(i=2;i<=n;i++){
9         int big=0;
10        if(i>m-1) big=max(big,f[i-(m-1)]);
11        if(i>m) big=max(big,f[i-m]);
12        if(i>m+1) big=max(big,f[i-(m+1)]);
13        f[i]=big+x[i];
14    }
15    ans=f[1];
16    for(i=2;i<=n;i++)
17        ans=max(ans,f[i]);
18    cout<<ans<<endl;
19    return 0;
20 }
```