[**完全背包问题**](http://www.cnblogs.com/gzydn/archive/2009/10/19/1586204.html)

Posted on 2009-10-19 21:56 [YDN](http://www.cnblogs.com/gzydn/) 阅读(374) [评论(1)](http://www.cnblogs.com/gzydn/archive/2009/10/19/1586204.html#commentform) [编辑](http://www.cnblogs.com/gzydn/archive/2009/10/19/1586204.html) [收藏](http://www.cnblogs.com/gzydn/archive/2009/10/19/1586204.html) 所属分类: [递归与回溯](http://www.cnblogs.com/gzydn/category/191172.html), [动态规划](http://www.cnblogs.com/gzydn/category/191175.html) http://www.cnblogs.com/gzydn/aggbug/1586204.html?type=1&webview=1

完全背包问题：

一个旅行者有一个最多能用m公斤的背包，现在有n种物品，每件的重量分别是W1，W2，...,Wn,

每件的价值分别为C1,C2,...,Cn.若的每种物品的件数足够多.

求旅行者能获得的最大总价值。

本问题的数学模型如下：

设 f(x)表示重量不超过x公斤的最大价值，

则 f(x）=max{f(x-i)+c[i]} 当x>=w[i] 1<=i<=n

可使用递归法解决问题程序如下:

http://www.cnblogs.com/Images/OutliningIndicators/ContractedBlock.gifhttp://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifCode  
 1http://www.cnblogs.com/Images/OutliningIndicators/None.gifprogram knapsack04;  
 2http://www.cnblogs.com/Images/OutliningIndicators/None.gifconst maxm=200;maxn=30;  
 3http://www.cnblogs.com/Images/OutliningIndicators/None.giftype ar=array[0..maxn] of integer;  
 4http://www.cnblogs.com/Images/OutliningIndicators/None.gifvar m,n,j,i,t:integer;  
 5http://www.cnblogs.com/Images/OutliningIndicators/None.gifc,w:ar;  
 6http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
 7http://www.cnblogs.com/Images/OutliningIndicators/None.giffunction f(x:integer):integer;  
 8http://www.cnblogs.com/Images/OutliningIndicators/None.gifvar i,t,m:integer;  
 9http://www.cnblogs.com/Images/OutliningIndicators/None.gifbegin  
10http://www.cnblogs.com/Images/OutliningIndicators/None.gifif x=0 then f:=0 else  
11http://www.cnblogs.com/Images/OutliningIndicators/None.gifbegin  
12http://www.cnblogs.com/Images/OutliningIndicators/None.gif   t:=-1;  
13http://www.cnblogs.com/Images/OutliningIndicators/None.gif   for i:=1 to n do  
14http://www.cnblogs.com/Images/OutliningIndicators/None.gif    begin  
15http://www.cnblogs.com/Images/OutliningIndicators/None.gif     if x>=w[i] then m:=f(x-i)+c[i];  
16http://www.cnblogs.com/Images/OutliningIndicators/None.gif     if m>t then t:=m;  
17http://www.cnblogs.com/Images/OutliningIndicators/None.gif    end;  
18http://www.cnblogs.com/Images/OutliningIndicators/None.gif   f:=t;  
19http://www.cnblogs.com/Images/OutliningIndicators/None.gif   end;  
20http://www.cnblogs.com/Images/OutliningIndicators/None.gifend;  
21http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
22http://www.cnblogs.com/Images/OutliningIndicators/None.gifbegin  
23http://www.cnblogs.com/Images/OutliningIndicators/None.gifreadln(m,n);  
24http://www.cnblogs.com/Images/OutliningIndicators/None.giffor i:= 1 to n do  
25http://www.cnblogs.com/Images/OutliningIndicators/None.gif   readln(w[i],c[i]);  
26http://www.cnblogs.com/Images/OutliningIndicators/None.gifwriteln(f(m));  
27http://www.cnblogs.com/Images/OutliningIndicators/None.gifend.

当m不大时,可以通过,但当m较大时,容易超时，

**改进的递归法**

改进的的递归法的思想还是以空间换时间,这只要将递归函数计算过程中的各个子函数的值保存起来,开辟一个一维数组即可

其实这种以空间换时间的存储式搜索就是动态规划的思想

http://www.cnblogs.com/Images/OutliningIndicators/ContractedBlock.gifhttp://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifCode  
 1http://www.cnblogs.com/Images/OutliningIndicators/None.gifprogram knapsack04;  
 2http://www.cnblogs.com/Images/OutliningIndicators/None.gifconst maxm=2000;maxn=30;  
 3http://www.cnblogs.com/Images/OutliningIndicators/None.giftype ar=array[0..maxn] of integer;  
 4http://www.cnblogs.com/Images/OutliningIndicators/None.gifvar m,n,j,i,t:integer;  
 5http://www.cnblogs.com/Images/OutliningIndicators/None.gif    c,w:ar;  
 6http://www.cnblogs.com/Images/OutliningIndicators/None.gif    p:array[0..maxm] of integer;  
 7http://www.cnblogs.com/Images/OutliningIndicators/None.giffunction f(x:integer):integer;  
 8http://www.cnblogs.com/Images/OutliningIndicators/None.gifvar i,t,m:integer;  
 9http://www.cnblogs.com/Images/OutliningIndicators/None.gifbegin  
10http://www.cnblogs.com/Images/OutliningIndicators/None.gif   if p[x]<>-1 then   f:=p[x]  //标记是否计算过避免重复计算（很重要）  
11http://www.cnblogs.com/Images/OutliningIndicators/None.gif   else   
12http://www.cnblogs.com/Images/OutliningIndicators/None.gif   begin  
13http://www.cnblogs.com/Images/OutliningIndicators/None.gif     if x=0 then p[x]:=0 else  
14http://www.cnblogs.com/Images/OutliningIndicators/None.gif      begin  
15http://www.cnblogs.com/Images/OutliningIndicators/None.gif        t:=-1;  
16http://www.cnblogs.com/Images/OutliningIndicators/None.gif        for i:=1 to n do  
17http://www.cnblogs.com/Images/OutliningIndicators/None.gif         begin  
18http://www.cnblogs.com/Images/OutliningIndicators/None.gif           if x>=w[i] then m:=f(i-w[i])+c[i];  
19http://www.cnblogs.com/Images/OutliningIndicators/None.gif           if m>t then t:=m;  
20http://www.cnblogs.com/Images/OutliningIndicators/None.gif         end;  
21http://www.cnblogs.com/Images/OutliningIndicators/None.gif        p[x]:=t;  
22http://www.cnblogs.com/Images/OutliningIndicators/None.gif      end;  
23http://www.cnblogs.com/Images/OutliningIndicators/None.gif     f:=p[x];  
24http://www.cnblogs.com/Images/OutliningIndicators/None.gif   end;  
25http://www.cnblogs.com/Images/OutliningIndicators/None.gifend;  
26http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
27http://www.cnblogs.com/Images/OutliningIndicators/None.gifbegin  
28http://www.cnblogs.com/Images/OutliningIndicators/None.gif  readln(m,n);  
29http://www.cnblogs.com/Images/OutliningIndicators/None.gif  for i:= 1 to n do  
30http://www.cnblogs.com/Images/OutliningIndicators/None.gif    readln(w[i],c[i]);  
31http://www.cnblogs.com/Images/OutliningIndicators/None.gif  fillchar(p,sizeof(p),-1);   //搜索中用于存储各阶段的最优质  
32http://www.cnblogs.com/Images/OutliningIndicators/None.gif  writeln(f(m));  
33http://www.cnblogs.com/Images/OutliningIndicators/None.gifend.