**算法设计**

**1、问题描述**

某商店中每种商品都有一个价格。例如，一朵花的价格是2元，一个花瓶的价格是5元，为了吸引顾客，商店提供了一组优惠商品价，优惠商品是把一种或多种商品分成一组，并降价销售。例如，三朵花的价格不是6元，而是5元，两个花瓶加一朵花的优惠价格是10元，试设计一个算法计算出某以顾客所购商品应付的最少费用。在给出预购商品的价格和数量，以及优惠商品价的情况下，编程计算所购商品应付的最少费用。

**输入数据：**用两个文件表示输入数据。第一个文件INPUT．TXT描述顾客所购物品（放在购物筐中）;第二个文件描述商店提供的优惠商品及价格（文件名为OFFER．TXT）;两个文件中都只用整数。第一个文件INPUT．TXT的格式为:第一行是一个数字B（0≤B≤5），表示所购商品种类数。下面共B行，每行中含3个数C，K，P。C 代表商品的编码（每种商品有一个唯一的编码），1≤C≤999。K代表该种商品购买总数，1≤K≤5。P是该种商品的正常单价（每件商品的价格），1≤P≤999。请注意，购物筐中最多可放5\*5＝25件商品。  
第二个文件OFFER．TXT的格式为:第一行是一个数字S（0≤S≤99），表示共有S种优惠。下面共S行，每一行描述一种优惠商品的组合中商品的种类。下面接着是几个数字对（C，K），其中C代表商品编码，1≤C≤9 99。K代表该种商品在此组合中的数量，1≤K≤5。本行最后一个数字P（1≤P≤9999）代表此商品组合的优惠价。当然，优惠价要低于该组合中商品正常价之总和。  
   
**输出数据：**在输出文件OUTPUT．TXT中写 一个数字（占一行）， 该数字表示顾客所购商品（输入文件指明所购商品）应付的最低货款。  
**2、问题分析**  
该问题与背包问题类似，由于动态规划要满足无后效性和最优化原理，所以我们来分析此题是否满足以上两点。先来状态表示的方法，商品不超过5种，而每种采购的数量又不超过5，那么用一个5元组来表示第I种商品买AI的最小费用：

**F（A1，A2，A3，A4，A5）**

1. 考虑这个状态的由来，当然，我们不用优惠商品也可以买，显然这样不是最优。于是如果我们能够使用第I条商品组合的话，状态就变为了：  
   **F（A1-SI1，A2-SI2，A3-SI3，A4-SI4，A5-SI5）**

（2）这样的话，状态1的费用为状态2的费用加上SI的费用，而状态2的费用必须最低（用反证法），同时，我们也不管状态2是如何来的（因为每一个优惠商品组合的使用是没有限制的），所以本题既满足无后效性，又符合最优化原理，同时还有重叠子问题产生，可用动态规划解决。  
通过对问题的分析，我们知道了状态的表示和转移的基本方法，我们得到一个状态转移方程：  
**F [a, b, c, d, e] = Min {F [a-S1, b-S2, c-S3, d-S4, e-S5] + SaleCost [S]}**初始条件为：  
**F [a, b, c, d, e] = Cost [1]\*a+Cost [2]\*b+Cost [3]\*c+Cost [4]\*d+Cost [5]\*e**即不用优惠的购买费用