**回溯法解决多维背包问题（MKP问题）设计思路分析**

测试数据分析：

1、理解测试数据所对应的的关系和结构

2、测试数据，举个最简单的测试数据，其他数据也类似······

6 10 3800 //6表示物品个数，10表示约束条件个数，3800是最优方案的价值

100 600 1200 2400 500 2000 //价值

8 12 13 64 22 41 //维度，6行10列的维度数组

8 12 13 75 22 41

3 6 4 18 6 4

5 10 8 32 6 12

5 13 8 42 6 20

5 13 8 48 6 20

0 0 0 0 8 0

3 0 4 0 8 0

3 2 4 0 8 4

3 2 4 8 8 4

80 96 20 36 44 48 10 18 22 24 //背包上限，80对应第一行维度，以此类推，也就是约束条件

3、思路分析：

①在保证不超过80的背包上限下选取每一维度的重量（容量等）所得的价值最多且保证不超过3800，则可选取最优方案。

②我先解决其中一个维度，也就是一个多维背包问题，我把这个测试数据分成多个多维背包问题来解决，当满足每一维度的约束条件就得到一个维度的01背包问题，利用回溯算法搜索最优方案，做10次循环，且满足约束条件，则可以得到路径和价值，也就是最优方案和最优价值。

③从键盘输入测试数据存入对应数组，初始化背包的当前的参数，定义构造方法访问对象的属性，执行回溯算法，从根节点开始回溯，然后先把所有的路径遍历出来，0表示不选取物品，1表示选取物品，当物品数量为6时，000000表示都不选取，选取则要满足约束条件。然后把一次次的排列结果存入数组中，先解决一个一维背包问题，判断物品选取方案是否满足约束条件，重量、价值，是否超出背包上限，则该物品选取方案的当前价值的总和与最大值比较，记录最大值的下标，输出最优路径。

④如果超出，减去上一个重量和价值，判断下一个选取方案，以此类推，执行10次，直到搜索完所有数据，得到十个一维背包问题的最优方案，存入数组，遍历数组搜索出公共最优方案，输出最优解决办法的下标和价值。

⑤得到该背包的最大价值以及所对应的最优解决方案。