

算法实现题 5-33 0-1 背包问题（习题 5-11）

★问题描述：

试设计一个用回溯法搜索子集空间树的函数。该函数的参数包括结点可行性判定函数和上界函数等必要的函数，并将此函数用于解 0-1 背包问题。

0-1 背包问题描述如下：给定 n 种物品和一个背包。物品 i 的重量是 w_i ，其价值为 v_i ，背包的容量为 C 。应如何选择装入背包的物品，使得装入背包中物品的总价值最大？

在选择装入背包的物品时，对每种物品 i 只有 2 种选择，即装入背包或不装入背包。不能将物品 i 装入背包多次，也不能只装入部分的物品 i 。

0-1 背包问题形式化描述：给定 $C > 0$ ， $w_i > 0$ ， $v_i > 0$ ， $1 \leq i \leq n$ ，要求 n 元 0-1 向量 (x_1, x_2, \dots, x_n) ， $x_i \in \{0, 1\}$ ， $1 \leq i \leq n$ ，使得 $\sum_{i=1}^n w_i x_i \leq C$ ，而且 $\sum_{i=1}^n v_i x_i$ 达到最大。

★数据输入：

由文件 input.txt 给出输入数据。第一行有 2 个正整数 n 和 c 。 n 是物品数， c 是背包的容量。接下来的 1 行中有 n 个正整数，表示物品的价值。第 3 行中有 n 个正整数，表示物品的重量。

★结果输出：

将计算出的装入背包物品的最大价值和最优装入方案输出到文件 output.txt。

输入文件示例

```
input.txt
5 10
6 3 5 4 6
2 2 6 5 4
```

输出文件示例

```
output.txt
15
1 1 0 0 1
```