

算法实现题 3-0 独立任务最优调度问题（习题 3-3）

★问题描述：

用 2 台处理机 A 和 B 处理 n 个作业。设第 i 个作业交给机器 A 处理时需要时间 a_i ，若由机器 B 来处理，则需要时间 b_i 。由于各作业的特点和机器的性能关系，很可能对于某些 i ，有 $a_i \geq b_i$ ，而对于某些 $j, j \neq i$ ，有 $a_j < b_j$ 。既不能将一个作业分开由 2 台机器处理，也没有一台机器能同时处理 2 个作业。设计一个动态规划算法，使得这 2 台机器处理完这 n 个作业的时间最短（从任何一台机器开工到最后一台机器停工的总时间）。研究一个实例：
 $(a_1, a_2, a_3, a_4, a_5, a_6) = (2, 5, 7, 10, 5, 2)$ ； $(b_1, b_2, b_3, b_4, b_5, b_6) = (3, 8, 4, 11, 3, 4)$ 。

★编程任务：

对于给定的 2 台处理机 A 和 B 处理 n 个作业，找出一个最优调度方案，使 2 台机器处理完这 n 个作业的时间最短。

★数据输入：

由文件 input.txt 提供输入数据。文件的第 1 行是 1 个正整数 n ，表示要处理 n 个作业。接下来的 2 行中，每行有 n 个正整数，分别表示处理机 A 和 B 处理第 i 个作业需要的处理时间。

★结果输出：

程序运行结束时，将计算出的最短处理时间输出到文件 output.txt 中。

输入文件示例

```
input.txt
6
2 5 7 10 5 2
3 8 4 11 3 4
```

输出文件示例

```
output.txt
15
```