

#### 算法实现题 4-15 套汇问题（习题 4-32）

##### ★问题描述：

套汇是指利用货币汇兑率的差异将一个单位的某种货币转换为大于一个单位的同种货币。例如，假定 1 美元可以买 0.7 英镑，1 英镑可以买 9.5 法郎，且 1 法郎可以买到 0.16 美元。通过货币兑换，一个商人可以从 1 美元开始买入，得到  $0.7 \times 9.5 \times 0.16 = 1.064$  美元，从而获得 6.4% 的利润。

##### ★编程任务：

给定  $n$  种货币  $c_1, c_2, \dots, c_n$  的有关兑换率，试设计一个有效算法，用以确定是否存在套汇的可能性。

##### ★数据输入：

由文件 input.txt 提供输入数据。文件含多个测试数据项。每个测试数据项的第一行中只有 1 个整数  $n$  ( $1 \leq n \leq 30$ )，表示货币总数。其后  $n$  行给出  $n$  种货币的名称。接下来的一行中有 1 个整数  $m$ ，表示有  $m$  种不同的货币兑换率。其后  $m$  行给出  $m$  种不同的货币兑换率，每行有 3 个数据项  $c_i$ ， $r_{ij}$  和  $c_j$ ，表示货币  $c_i$  和  $c_j$  的兑换率为  $r_{ij}$ 。文件最后以数字 0 结束。

##### ★结果输出：

程序运行结束时，对每个测试数据项  $j$ ，如果存在套汇的可能性则输出“Case  $j$  Yes”，否则输出“Case  $j$  No”。所有结果输出到文件 output.txt 中。

##### 输入文件示例

```
input.txt
3
USDollar
BritishPound
FrenchFranc
3
USDollar 0.5 BritishPound
BritishPound 10.0
FrenchFranc
FrenchFranc 0.21 USDollar
0
```

##### 输出文件示例

```
output.txt
Case 1 Yes
Case 2 No
```