例子：假设程序的输入域划分为个不相交的分区，每个分区中有个测试用例。初始化MAPT和RAPT的两个参数,。根据以往的测试经验或者测试历史，不妨设置每个分区的惩罚上限。

**根据MAPT算法步骤，测试程序f的过程如下：**

测试任务开始前，设置初始测试剖面，初始状态转移矩阵如下：

第一次测试时，根据选择分区，不妨假设被选中，然后从中随机选择测试用例并执行。

* 如果t揭示了软件故障，根据公式6和7，调整状态转移矩阵的第一行，结

果如下：

* 如果t没有揭示软件故障，根据公式8和9，调整状态转移矩阵的第一行，

结果如下：

然后根据更新后的转移概率选取下一个测试用例所在的分区，依据测试用例的执行结果更新状态转移矩阵的第行。

**根据RAPT算法步骤，测试程序f的过程如下：**

测试任务开始前，设置初始测试剖面，每一个分区的惩罚因子，奖励因子，其中。

在测试过程中，根据选择分区，不妨假设被选中，然后从中随机选择测试用例并执行。

* 如果t揭示了软件故障，令，。接下来在中

选择的第二个测试用例揭示了故障，则，第3个测试用例没揭示故障，此时,。根据公式10和11调整测试剖面，然后令。

* 如果t没有揭示故障并且，根据公式12和13调整测试剖面

，然后令。

* 如果t没有揭示故障并且，认为的失效率很低，根据公式

12和13调整测试剖面，然后令。（选取概率不会一直是0，原因是下一个选取分区中的测试用例没有揭示故障时，选取概率增加。）

**注意：**在测试过程中，计算机的计算精度导致可能出现分区选取概率的和不等于1或者转移矩阵某一行的状态转移概率的和不等于1的情况。