

## Evil Book 解题报告

绍兴市第一中学 王文涛

### 试题来源

CodeChef MARCH12 EVILBOOK

### 试题大意

有 $N$ 个厨师，对于第 $i$ 个厨师可以付出 $C_i$ 点努力打败他并得到 $M_i$ 点魔力，同时 $M_i$ 变为0。在拥有魔力不少于 $X$ 时可以使用帮助，即通过消耗 $X$ 点魔力使某个厨师的 $C_i$ 和 $M_i$ 都乘以 $\frac{1}{3}$ 。初始时拥有魔力为0。求收集至少666点魔力所需最少的总努力。如果不可能，输出“impossible”。

数据范围：

$$1 \leq T \leq 5$$

$$1 \leq N \leq 10$$

$$10 \leq X < 666$$

$$0 \leq C_i \leq 10^9$$

$$0 \leq M_i \leq 10^9$$

时间限制：CodeChef上的3s

### 算法介绍

首先应该观察到一些基本的结论。

#### 结论1

打败某个厨师获得的魔力必然大于对他使用帮助所消耗的魔力。

如果打败某个厨师后魔力反而减少了，那还不如不打。因此打败一个厨师后魔力必然增加。设对第 $i$ 位厨师共使用了 $k$ 次帮助。 $k$ 应满足 $\frac{M_i}{3^k} > kX$ 。

## 结论2

我们可以只在打败某个厨师之前的时刻对他使用帮助。

由于使用帮助需要消耗魔力，因此在更早的时候对他使用帮助只会使得更长时间范围内的魔力值降低，而这显然是不优的。

## 结论3

对于第 $i$ 位厨师，令 $K$ 为最小的非负整数使得 $\frac{M_i}{3^K} < 666$ 。设当前在魔力的限制下最多能用 $can$ 次帮助。则对该位厨师使用的帮助次数 $k$ 满足 $k \geq \min(can, K-1)$ 。

因为如果 $k \leq K-1$ ，在打败这个厨师之后必然能完成任务。在一定打败这个厨师的情况下，为了使得付出的努力 $C_i$ 尽可能小， $k$ 应该尽可能大。

## 最终算法

我们可以使用搜索。每次选择一个未被挑战的厨师，枚举其需要使用的帮助次数 $k$ ，进入下一层搜索。可以记录每个厨师使用的帮助次数的状态，如果之前已经搜索过这种状态了那么就不再搜索（因为只要确定了每个厨师使用的帮助次数，对之后的影响是一样的，继续搜索得到的答案是相同的）。我们还可以加上一些常用的剪枝：加上剩下的所有魔力都不能达到666点（可行性剪枝）或者按最优的投入产出比值计算都不比当前答案优（最优性剪枝）就退出。

设 $r$ 为第 $i$ 位厨师的 $k$ 最多可能的取值种数。经计算可知 $r \leq 4$ 。这样状态数就是 $r^N = 1048576$ 。但实际上的状态数可以更少。比如，当 $M_i \leq \frac{666}{3}$ 时最多只有3种状态，因为最多只能使用2次帮助；而当 $M_i > \frac{666}{3}$ 时，最多只会有2个恰好使用了 $K$ 次帮助的厨师，因为如果有3个的话就一定能够达到666点魔力了。因此状态数不会多于 $3^N + N \times 3^{N-1} + \frac{N(N-1)}{2} \times 3^{N-2} = 551124$ 。但实际状态数远远小于这个值。

时间复杂度： $O(T \times r \times N \times \text{状态数})$ 。