Contra 解题报告

杭州学军中学 李超

解题思路：

显然，随着 p 的增大，获得的总分的期望也会增大。我们可以二分答案 p，计算答案为 p 的时候的期望得分。首先我们容易想到一个搜索算法，即按数学期望的定义，暴力模拟 chnlich 玩游戏的每一关的胜负，每种情况得到的分数计入总答案，计算数学期望并输出，设二分次数为 T，则时间复杂度为 O(T*2^N)，期望得分 20 分。

容易发现，在第 i 关后的情况只和当前的 u 值和命数有关，令 f[i][j][k]表示在第 i 关，u 值为 j，还剩 k 条生命的得分，g[i][j][k]表示在第 i 关，u 值为 j，还剩 k 条生命的概率，我们可以容易地列出递推式，这里略去，最后的数学期望为 $\sum_{i=1}^{N} f[i][0][0]*g[i][0][0]+\sum_{j=0}^{R}\sum_{k=1}^{Q} f[N][j][k]*g[N][j][k]$。同样设二分次数为 T，则时间复杂度为 O(T*NRQ)，期望得分 50 分。这个算法已经很难继续优化，即使使用矩阵乘法进行优化，由于要记录 f 和 g 两个数组，矩阵的边长达到了 2*R*Q，在极限情况有 200 左右，较难通过极限数据。

为了进一步优化这个转移方程，我们可以从所求的答案入手。我们要知道的并不是方程中每一项的值，而是最后的数学期望值，而<u>每通一关所获得的分数，在最后的时候统计与在动态规划的过程中统计，它的本质是一样的</u>。在之前的动态规划中，我们把每一关所获得的分数拆分到答案的项中，然后再加起来，实际上的值就是该分数乘到达该状态的概率。因此，我们可以只记录到达一个点的概率，而不考虑到达这个点时的数学期望。同样令 g[i][j][k]表示在第 i 关，u 值为 j，还剩 k 条生命的概率，则递推方程如下：

g[i+1][min(j+1,R)][min(k+1),Q]+=g[i][j][k]*p

g[i+1][0][k-1]+=g[i][j][k]*(1-p)

Answer+=g[i][j][k]*p*(j+1)

最后的答案即为 Answer。设二分次数为 T，时间复杂度仍为 O(T*NRQ)，期望得分仍为 50 分。然而，通过这种方式，我们将一个需要两个数组进行计算的方程优化成了只需一个数组即可计算的方程。对于这个方程，由于对于所有 i，从 i 到 i+1 的转移矩阵都是相同的，因此可以快速幂矩阵乘法。由于 u>=Q-1 时，生命一定是 Q，因此有用的状态在极限情况下不超过 30，即矩阵边长只有 30 左右。具体实现时可以先找出所有有用的状态。仍然设二分次数为 T，则总时间复杂度为 O(T*R^3*Q^3*logN)，期望得分 100 分。