1000 pt.

这道题目是比较麻烦的DP状态压缩题。

首先容易想到的是dp[p][x]：[p+1,N]的数字已经确定了，它们需要满足当前元素开始小于等于x，[0,p]共有多少种可能情况，则最终的解为dp[N][无穷]。

但是x很大，需要将x转换成别的什么小一点的状态空间。

需要注意到，每一个数字所能变成的最后的数字的个数是有限的，由于是16位的原数字，则最多2^16-1个可能的数字，那么可以将这些所有可能的数字排序，而后以下标索引即可。

所以将状态空间简化为dp[p][k]：他们需要满足当前元素开始小于等于x[p][k]，这个条件下的所有可能情况数目。则最终的解为dp[N][x[N][M-1]]。

则当前选择有两种，一种是选择了x[p][k]这个数字，这种状态下的子问题为dp[p-1][j]，j:在p-1的所有可能数字中满足x[p-1][j]<=x[p][k]]。

另一种是选择了小于x[p][k]的其他数字，这种状态等价于dp[p][k-1]的定义。除了k=0，这种情况下没有其他选择。

Dp[p][k]=dp[p-1][j]+dp[p][k-1]。

DP 从小到大，逐步计算每个子问题，直到其扩大到原问题。