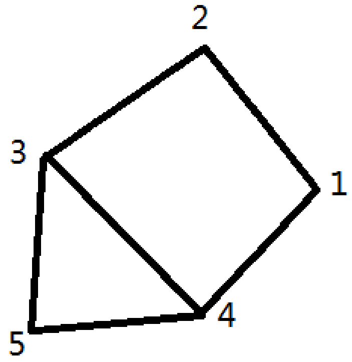
黑魔法师之门

题目来源：XLk 提供



大致思路：
实际上每次操作后的答案就是 2^(图中”元”环的个数) -1。
元环的意思如右图所示，(1-2-3-4-1)和(3-4-5-3)是元环，1-2-3-5-4-1 不是，因为它可以看做由上述的两个环合成。
因为一个环里每个点的度数都是大于零的偶数，我们可以这样来构造答案：每个环有选和不选两种选择，如果选择了该环，那么环上所有边的“选择次数”+1。最后取所有“选择次数”为奇数的边构成一个边集，就是一个答案。可以证明这样构造出来的解不重复且涵盖了所有情况。因此答案就是 2^(图中”元”环的个数)。实现方法非常简单，只需要一个并查集即可。

具体实现方法：
并查集维护连通性，初始化 ans=1。
加入一条边(x,y)时，如果 x 和 y 在同一集合内，ans*=2。
每次询问输出 ans-1。
时间复杂度 O(Mα(N))， α(N)代表并查集的复杂度。

守卫者的挑战

题目来源：Poet_shy 摘录自 CodeForces
比较明显的动态规划。F[i][j][k]表示经过前 i 项挑战，目前背包容量为 j，有 k 项挑战获得了胜利的概率。j>0 代表背包有 j 的剩余空间，j<0 代表目前有-j 的地图残片还未装入。j 的取值范围是-200~200，超出范围没有意义，直接与±200 取 Min/Max 即可。
转移方式只有两种：第 i 次挑战成功/失败.
时间复杂度 O(400*n^2)。

终极武器

题目来源：lydrainbowcat 改编自 NEERC2009(POJ3872)
这道题是一道防 AK 题目= = 想得到满分是很困难的。不过仔细观察数据范围可以发现，这道题拿到 80 分左右还是比较可行的。

方法一：
随机数据很容易造成每个数字独立一个等价类。因此直接输出 1~9，每行一个数字。
25%的随机数据是为这种算法准备的。

方法二：
暴力枚举每个波段内的数，枚举每一位上的数字修改成什么，判断修改后的数是否在波段内。
前 25%的数据是为这种算法准备的。

到此可以发现暴力加上分析随机数据性质就能拿到大约 50 分，数据还是非常仁慈的，包括

接下来 N=1，K=1 的数据，充分给予了选手其他算法的施展空间。

方法三：
N=1 的数据用简单的数位统计就可以处理，乱搞也能得到不少分数。
有 25%的数据是为这种算法准备的。

方法四：
K=1 的数据只能修改个位。不难想到，如果限定只有个位可以换成其他的数，那么这个题是不是就变成水题了呢？
首先建立一个有 9 个点的完全图，代表 1~9 这九个数字，然后扫描所有的[10T + 0, 10T + 10)（T∈N）这样的区域，由于我们只考虑个位，并且不考虑 0，那么就相当于 1~9 这九个数字被分成了两组，两组之间的数不能互相交换，于是我们在原图中把两组数之间的所有边都断掉。扫描完所有的区间之后，原图应该还剩下若干个团，这样的话每个团中的数就是 K-等价的。考虑到数据范围能达到 10^18，扫描的时候只需要扫描包含区间的端点的区域，这样我们就得到了一个 O(N)的算法。
这种算法是引导选手走向满分解法的路径，因此为这种算法准备了 30%的数据。

方法五：
读入所有区间之后，转化为前闭后开形式即 [L,R)，存入数组……然后我们拿出其中的一个区间来考虑。。。
单独对于这个区间，我们考虑第 k 位上的数字变化。如果我们知道比第 k 位低的位上都是都是什么数字的话，显然可以很容易地判断出这一位上的数字 1~9 可以分成哪两组等价类……
比如区间[2012,6278)，考虑十位。如果规定各位数字是 0，那么[2020,6280)中的数在区间内，剩下的在区间外，即十位上[2,8)是一个等价类，其余数字是另一个。如果规定个位数字是 2，那么十位上是[1,8)，如果个位数字是 8，那么十位上是[1,7)。
可以发现最初第 k 位上的数字的其中一组是[L/10^k %10 +1, R/10^k %10 +1)，当后面的位经过 L%10^k 这个数的时候区间左端减一，经过 R%10^k 这个数的时候区间右端减一。
因此我们可以依次处理每一位，对于每一位上，把读入的 n 个区间中，所有这样引起该位数字区间变化的数值记录下来，排序离散化之后再依次处理。可以发现引起变化的位置只可能是这 n 个区间的左右端点，并且变化规律就是上面一段所说的。
大概梳理一下整个算法过程：
初始化：建一个 9*9 的完全图的邻接矩阵。
特殊处理个位情况。
主要环节：1.枚举每一位，设当前枚举到第 k 位。
2.按照上面所述的方法计算引起变化的位置，排序，离散化，离散后的每个位置开一个 1*9 的数组记录哪些数字在数字区间内，哪些在外面。
3.首先令后 k-1 位都是 0，计算各个数字区间（1*9 的数组）的初始值。
4.依次循环每个变化位置，对应的 1*9 的数组里进行加减。
5.每次变化后把对应的 1*9 数组的信息反映到 9*9 的矩阵里，不同集合内的边断掉。
输出答案。

该算法综合运用到了邻接矩阵、排序离散化、Hash 等多项 NOIP 中进行数据统计处理的知识，可以处理 100%的数据，得到满分。