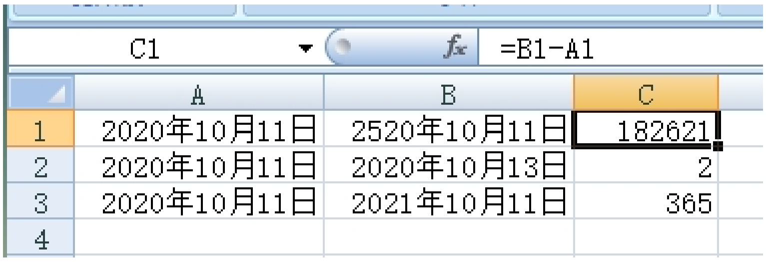
01.五百年

蓝桥杯 office 考核赛？



C1 =B1-A1

	A	B	C
1	2020年10月11日	2520年10月11日	182621
2	2020年10月11日	2020年10月13日	2
3	2020年10月11日	2021年10月11日	365
4			

02.进制转换

进制转换一般都会先将某进制转化为 10 进制，再由 10 进制转化为想要的进制。这里题目给的是 36 进制的“ILOVENANLI”数字，从左至右让当前数位乘以 36 后加入下一数位，直到最后一位。转化为 3 进制就一直除 3 存余数，然后倒着输出余数即可。

需要写一个程序来实现，当然也可手推。

（不了解进制转换的同学可得加把劲好好学了）

03.福报数

题目需要求数字中存在连续的“996”，直接 for 循环求即可。

循环遍历 1~n，然后将当前数字不断除以 10 缩小，并用模 1000 来检验数字中是否含有“996”这个数字。

即便范围很大暴力杯也可挂机求解，不用担心程序耗时。但若题目数据超过 10^9 时，注意储存类型需要用 long long int 或 unsigned long long int 等类型的使用。

04.等差数列

题目需要求十个 2020 以内的数构成的等差数列，公差大于 0。

同上题一样需要 for 循环遍历求解，遍历数组模拟等差数列的首数字和公差，然后验证末数字即可。

05.单词方阵

题目要求填入四个单词，深搜填入每个单词即可。

注意单词填入有四个方向，且需要注意超界问题。同时需要注意第一个单词“nan”和第三个单词“new”可以存在公共部分“n”，也可以不存在，需要分开讨论。最后放“b”的时候就统计空余的方格数就是需要加的方案数。

06.排灯

贪心思路，可以想到最左边亮灯的灯必须由其下开关所控制，所以最少要将此开关拨动。所以贪心思路为从左至右模拟开关拨动的状态。

代码只需从左至右遇见亮灯的便拨开关，然后模拟之后两个灯的亮灭即可。

07.重建道路

虽然题目看起来花里胡哨，其实还是最裸的最小生成树。

题目保证所有点联通，将“完好的道路”直接用并查集连接，完好道路的费用在其中没什么作用，无需考虑直接丢弃。

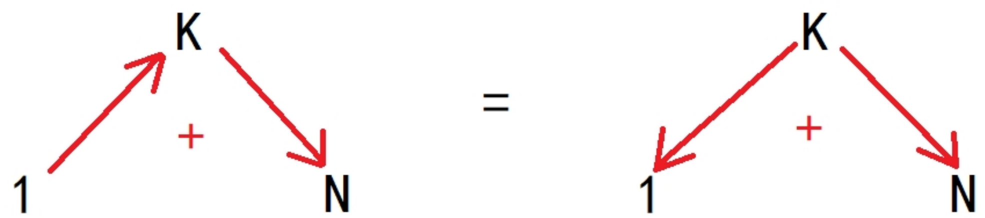
然后便是最小生成树的写法了，将剩余地震震坏的道路以价值排序，然后用是否连接两个不同集合的城市为判断一句，是否要加入到维修费用里。

题目需要注意的地方是在大数据范围时维修费用可能会超出 int 范围，需要用 long long int 储存。

算是一个最小生成树的模板题了。

08.金融危机

最短路中迪杰斯特拉的模板题，题目中所给道路有重边，可以用邻接矩阵读取并去重。题目所求为从 1——K——N 的最短路，其实只用从 K 点求一遍最短路，然后用 K——1 加上 K——N 的最短路即可。



数据比较水，两次迪杰斯特拉跑最短路也可在时限内运行出答案。

09.AC 钥匙

题目仅需要找到钥匙 A 和钥匙 C，钥匙 B 在读图时就可变为通路。在拿取钥匙时，钥匙 A 和钥匙 C 不分先后顺序，所以只需要经过一个 A 经过一个 C 即可。

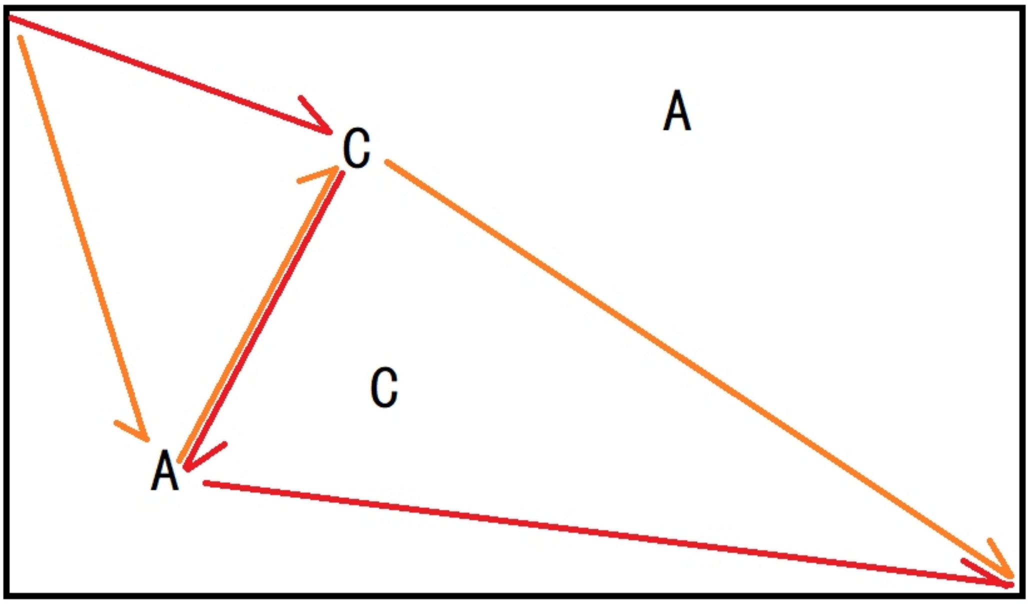
在各有一把钥匙时，可以看出路线为 (1,1)——A——C——(N,M) 或者 (1,1)——C——A——(N,M)，进行三次广度优先搜索即可。

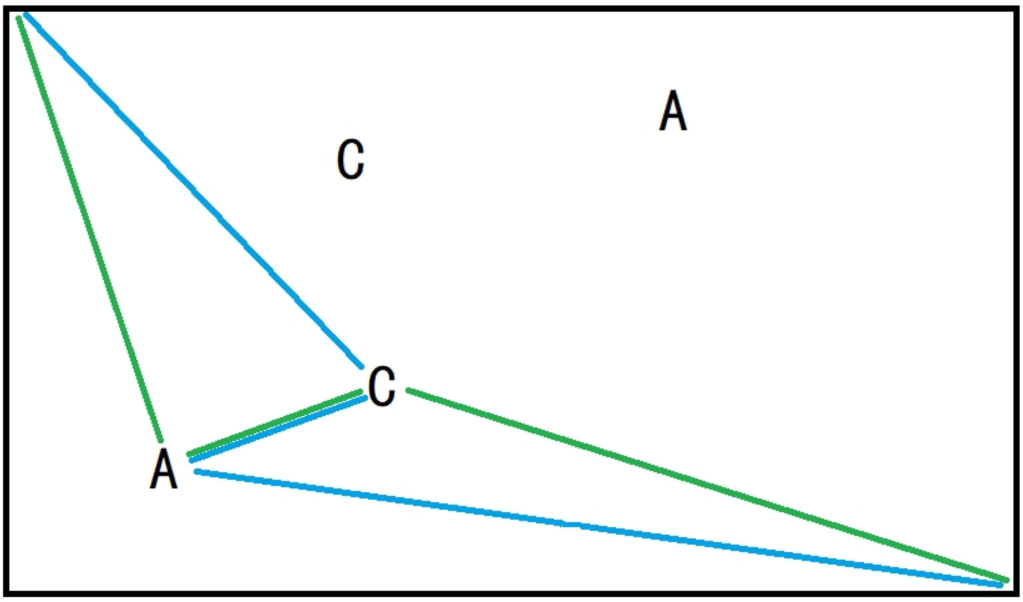
当钥匙数量为多把时，首先两次广搜预处理起点(1,1)和终点(N,M)到所有点位置。

考虑从(1,1)位置出发，先到某一钥匙 A（C），再到某一钥匙 C（A），最后走向终点(N,M)。这里可以看出必须经过一条 AC 之间的道路，因 A 钥匙数量要少于 C 钥匙，对钥匙 Ai 所在位置进行广搜，每对一个位置进行广搜便可得钥匙 Ai 到所有钥匙 C 的最短路，然后再加上起点和终点到两个钥匙的耗时。最后比较一个最小耗时即可。

比较公式：

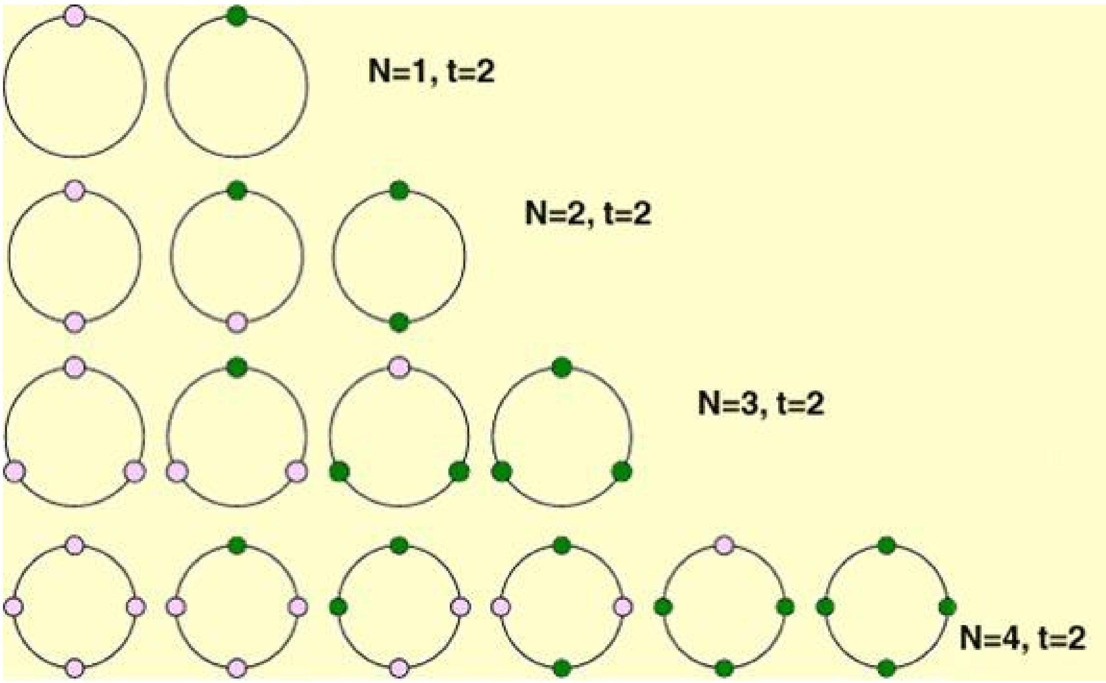
```
s = A[k].Begin + Step[C[i].x][C[i].y] + C[i].End;
s = A[k].End + Step[C[i].x][C[i].y] + C[i].Begin;
```





如图所示，通过一次对 Ai 的广搜可搜出所有 Ai——C 的道路，加上预处理起点终点到所有点值即可求出解。

10.奇特的手链



题意：

手链由若干珠子穿成的环形首饰，且可以进行翻转和旋转操作。输入整数 N 和 t，分别输出用 t 种颜色的 N 颗珠子（每种颜色的珠子颗数不加以限制）能制作手链的方案数。

分析：

等价类计数问题。一共有两种置换，选择以及翻转。手链有两种置换。设所有珠子按逆时针编号 0~n-1。

旋转置换：如果逆时针旋转 i 颗珠子的间距，则珠子 0、i、2i、…构成一个循环。这个循环有 n/gcd(i,n)个元素。根据对称性，所有循环的长度相同，因此一共有 n/(n/gcd(i,n)) = gcd(i,n)个循环。该置换的不动点数为 t^(gcd(i,n))。所有置换的不动点总数为 a = sum{t^gcd(i,n) | i = 0,1,…,n - 1}。

翻转置换：分情况讨论。当 n 是奇数时，对称轴有 n 条，每条对称轴形成(n-1)/2 个长为 2 的循环以及 1 个长为 1 的循环，即(n+1)/2 个循环。这些置换的不动点总数是 b=nt^((n+1)/2)。当 n 是偶数时，有两种对称轴。穿过珠子的对称轴有 n/2 条，各形成 n/2-1 个长为 2 的循环，还形成两个长为 1 的循环；不穿过珠子的对称轴有 n/2 条，各形成 n/2 个长为 2 的循环。这些置换的不动点总数是 b = n / 2 * (t ^ (n / 2 + 1) + t ^ (n / 2))。

根据 Polya 定理，手链总数是(a + b) / (2n)。

另可参考 UVa10294