

出题人: 不朽之夜



# ・ 题目大意:

- 有一个n个点m条边的带权图,你一开始在1号点,你要按顺序完成k个任务,第i个任务是先去a[i]再走到b[i]。当你走到一个点上的时候,你可以在这个点创建一个传送门。当同时存在两个传送门的时候,你可以在传送门之间不耗代价地传送。如果已经存在了两个传送门,你想再创建一个,就必须选择之前的一个传送门关掉(关掉这个操作不耗时间,并且是远程操作,不需要走过去)。问完成所有任务的最短总行走距离。
- 1 <= n, k <= 300
- 1<=m<=40000



- 任务就是要依次经过2k个点
- 考虑最暴力的DP
- 设f[i][u][a][b]表示已经完成了前i个任务, 当前在点u, 两个传送门分别位于a和b的最短距离。
- 转移有三种:
- 1.在u设立传送门
- 2.从u走到一个相邻节点
- 3.如果u等于a或者b,那么可以使用传送门
- 这个DP的转移显然是有环的,因此需要使用Dijkstra算法。



- 仔细观察,发现记录两个传送门的位置是没有用的,只需要记录一个。因为如果我们要使用一个传送门一定是走到那个节点再使用(我们可以随时在当前节点创建传送门)。
- 设f[i][u][p]表示当前已经完成了前i个任务,在节点u,其中一个传送门位于点p。
- 转移有4种:
- 1.从u走到相邻节点
- 2.在u设置传送门,令p=u
- 3.从u传送到p
- 4.在u设置传送门,传送到p,并只保留u的传送门(交换u和p)





- 再仔细观察,发现可以继续精简状态,设c[i]表示第i个任务的节点。
- 设f[i][p]表示当前已经完成了前i个任务, 当前正在c[i]上, 传送门的位置在p。
- 可以证明, 只需要3种转移, 就可以覆盖所有情况:
- 1.直接从c[i]走到c[i+1]
- 2.枚举走到c[i+1]之后,传送门的位置变为了哪个节点,设这个节点是q。第二种转移是从c[i]走到q/在q设置传送门,从q传送到p,再从p走到c[i+1]
- 3.第三种转移是从c[i]传送到p,从p走到q,在q设置传送门,最后从q走到c[i+1]
- 复杂度为O(kn^2)



# B- Graph

- **大致题意**:给一棵树,每条边有边权。可以任意加边和删边,但要满足任何时刻图连通,而且任何一个环的边权异或和为0。求操作后最小权值和。
- 做法: 可以发现任意两个点之间连边的权值都是固定的。由于图始终联通,所以两点间始终存在至少一条路径,如果存在多条,根据环的异或和为0,两点间的路径的异或和应该相等,且始终是固定的。
- 所以可以给每个点一个权值, 那么两点间的连边权值就应该是两端点权的异或。
- 接下来的问题就是异或最小生成树。
- 字典树
- boruvka



# C- Easy

## • 题意:

Mr. W is writing sequences. If he writes two positive integer sequences A and B with length K which satisfy  $\sum_{i=1}^{K} a_i = N$ ,  $\sum_{i=1}^{K} b_i = M$ ,

he will get  $P = \prod_{i=1}^{K} min(a_i, b_i)$  points.

You want to know the sum of total points he can get in all possible sequences he can write.

#### • 做法:

生成函数  $\Sigma$ min(N, M)\*(x^N)\*(y^M) = xy/((1-x)(1-y)(1-xy))

枚举(1/(1-xy))^K的次数

用1/(1-x)^K 和 1/(1-y)^K补到N-K和M-K



# **D- Drop Voicing**

## ・ 题目大意:

- 给定一个1~n的排列,有两种操作:
- 操作1: 可以将倒数第二个数放到开头
- 操作2: 可以将开头的第一个数放到最后
- 连续若干次操作1 (包括1次) 称为一段
- 现在要将排列变成1~n,要使得段数尽可能少,输出这个最小值。
- 2<=n<=500</li>





# **D- Drop Voicing**

- 我们将模型转换成有一个圆盘,圆盘上有n个位置,且有一个指针(初始时指针指向原来的最后一个元素)
- 连续若干次操作1等价为改变指针指向的数所处的位置
- 连续若干次操作2等价为改变指针的位置
- 容易观察到,我们只需要取环上最长的一个上升子序列,然后调整其它数,即可得到答案
- 时间复杂度O(n^3)或O(n^2)





# **E- Bogo Sort**

#### • 题意:

• 给定置换, 求有多少排列可以通过这个置换变成顺序。

#### • 做法:

- 所有 cycle 长度的 LCM。
- · 由于cycle 长度总和是 n,所以一定不会大于 n 位数,不取模即可。
- 最长的答案也就几百位。



# F- DPS

# • 题意:

• 模拟显示游戏中的伤害数据。

# • 做法:

• 模拟即可。注意恰好爆 int。double 可能有精度问题。





# **G- Greetings Souvenir**

#### • 题目大意:

- 给定一棵大小为n的树,且树上的每个节点有一个权值a[i],现在要给每个节点确定一个数值b[i],定义val[i]=b[i]\*(i子树内有多少个点的权值a等于b[i])。要使得所有点val[i]的mex尽可能大,输出这个最大值。
- 1<=n<=20000
- 时间限制8s,空间限制64MB





# **G- Greetings Souvenir**

- 首先进行预处理,我们可以通过DFS以及启发式合并预处理出每个子树中的颜色分布,从而计算得到每个节点u可以得到哪些不同的value[u]
- 我们可以由此得到一个二分图:左侧为n个不同的点,右侧为n个不同的value,如果左侧的节点u可以 得到右侧的value则进行连边。
- 可惜这样边数为O(n^2),而本题内存只有64MB,是不能通过此题的。





# **G- Greetings Souvenir**

- 事实上我们可以把二分的log去掉
- 假设当前答案在[L,R]之间, 我们先判断mid=(L+R)/2是否合法
- 如果mid不合法,我们继续对[L,mid]进行二分
- 如果mid合法,我们继续对[mid,R]进行二分,此时我们只需要在mid状态下的残余网络上继续进行网络流即可。
- 时间复杂度O(n^2logn)
- 这里向本题被卡TLE / MLE的选手表示一下抱歉,可能这个题确实卡的比较紧。



#### H- Interval

Mr. W have a sequence A with length N.

$$F(l,r) = A_l \& A_{l+1} \& ... \& A_r$$

• 题意: Set  $S(l,r) = \{F(a,b) \mid min(l,r) \le a \le b \le max(l,r)\}$ 

Mr.W makes Q queries. For each query he wants to know the size of S(L,R) for given L, R.

L, R will not be given directly. He will give you L' and R'.

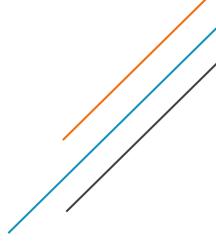
$$L = (L' \oplus lastans)\%N + 1$$

$$R = (R' \oplus lastans)\%N + 1$$

⊕ means XOR.

Lastans donates the answer of last query. It's zero at the beginning.

- 注意L可能大于R (应该都注意到了)
- 做法:
  - 对于固定的左端点或者右端点,不同的值至多只有log个
  - / 先把这些值找出来, 然后去掉具有包含关系的相同的值



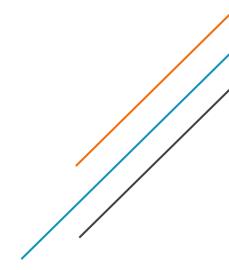




# H- Interval

# • 做法:

- 对于固定的左端点或者右端点,不同的值至多只有log个
- 先把这些值找出来,然后去掉具有包含关系的相同的值
- 单纯的二维偏序求和无法处理相同的值
- 那么把相同的值按顺序放在一起 (l1,r1)(l2,r2)...(lp,rp)
- 那么如果查询包含了(l1,r2)(l2,r3)...(lp-1,rp)这些区间,答案就应该-1
- 所以设置一些权值为-1的区间,再二维偏序即可。







# I- Hard Math Problem

#### • 题意:

· 有一个无穷大的二维网格,每个格子可以是1、2或者3,每个1旁边要有一个2和3,要使机器的占比 最大。

#### • 做法:

- (i+j)%3 =0 交错2和3
- 答案是2/3
- 因为这样每个1旁边恰好有一个2和3,而任意两个2和3不相邻。
- 可以计算出这是上限。



# J- Cone Walker

・ 题意: 无穷大的圆锥表面上两个不同的圆, 求相交的面积

思路: 计算几何, 分类讨论

做法:

1.先把圆锥面按照两个圆心的角度分成四块,分别展开成扇形,转化为扇形和两个圆的交

2.分类讨论,按照两个圆的位置关系(包含和相交),扇形顶点和两个圆的位置关系(在交的内部和外部),以及扇形的两条射线和两圆相交区域的交点情况(相交,相切,不相交)分类讨论。



# K- Git Merge

# • 题意:

• 合并一些代码,使代码长度最短,输出方案。

# ・ 做法:

- 大模拟
- 先预处理出两个文件,然后 dp[i][j][0/1/2] 表示第一个文件第 i 行,第二个文件第 j 行,在公用/ifdef 1/ifdef 2 内,切换有代价 1,dp n m 0 是答案。
  dp 可以存在 short 里。



# Thanks

