牛客竞赛
AC.NOWCODER.COM

NIO

# 2022 牛客
# 暑期多校训练营

单击此处替换副标题

# 通过人数

- C 1262/6144
- A 867/2407
- J 464/3889
- H 213/1946
- F 101/1013
- D 27/66
- B 18/417
- G 8/492
- I 4/102
- E 3/18

- 我们的期望：

  没有队伍AK ✕ 两个队AK了，tql

  每道题都过队伍过 √

  队伍数随题数指数下降 √ 算是完成了一半，以意想不到的方式

  没有很长的代码 √

  所以出了很多要想比较久的题

# 反思和谢罪

- 出题人英语不好，A题面有错误导致大家看不懂
- F题数据超过范围了，导致了大概7个队伍段错误/答案错误，耽误了半个小时，对不起
- C题看上去在卡常数，但其实不是

- 对不起！！

- 想找点借口：
- 验题的时候确实都看懂了，没预料到
- C题本来本地跑的时候数据出的比较大，在oj上爆内存了改小了一点，但是本地三倍以上差距，到oj上只有两倍了

- 题意：
  - 给定n个字符串，求一个将他们拼接起来的方案，使得结果的字典序最小。

- 一个简单的做法是，对n个字符串排序，a在b前面的条件是ab<ba。使用stable_sort可以保证复杂度为O(|S|log n)。该做法的正确性请各位自行证明。

- 考虑如何优化上述排序过程。将n个字符串建一个trie。对于两个字符串a, b，不妨设|a| < |b|。如果a不是b的前缀，则可以直接比较二者在dfs序的大小；否则，可以用Z-algorithm(也称exkmp)来判断ab和ba的大小关系。

- 因此我们可以维护一个有序的链表，按dfs序依次将字符串插入链表中。对于一个字符串s，找到他的所有出现过的前缀，可以在O(|s|)的时间复杂度内求出每个前缀在自己前面还是后面（如果相等，可以任意规定是前面还是后面）。假如在自己前面的前缀集合为A，在自己后面的集合为B。

- 则B中所有元素在链表里必然是相邻的，且一定位于链表末尾。证明：如果不满足这个条件，说明存在一个其他元素x，和B中的元素B1，满足B1<x。如果x是s的前缀，则x∈A，因此x<s；否则，由于x在dfs序中位于s前面，所以x<s。总之，x<s，而B1>s，和前面的B1<x矛盾。

# C A J H F D B G I E

- 题意：
  - 给定n个字符串，求一个将他们拼接起来的方案，使得结果的字典序最小。

- 一个简单的做法是，对n个字符串排序，a在b前面的条件是ab<ba。使用stable_sort可以保证复杂度为O(|S|log n)。该做法的正确性请各位自行证明。

- 因此我们可以维护一个有序的链表，按dfs序依次将字符串插入链表中。对于一个字符串s，找到他的所有出现过的前缀，可以在O(|s|)的时间复杂度内求出每个前缀在自己前面还是后面（如果相等，可以任意规定是前面还是后面）。假如在自己前面的前缀集合为A，在自己后面的集合为B。
- 因此可以找出B中在链表里最靠前的元素，将s插进链表中。总复杂度是线性的。

- 由于judge不太稳，担心卡常数影响队伍体验，时限开得松。最后结果是暴力排序也让过了，作为一个简单题。

# C A J H F D B G I E

- 题意：
  - 给出两棵编号1-n的树A B，A B树上每个节点均有一个权值，给出k个关键点的编号$x_1 \ldots x_n$，问有多少种方案使得去掉恰好一个关键点使得剩余关键点在树A上LCA的权值大于树B上LCA的权值。

- 预处理出关键点序列的在树A B上的前缀LCA和后缀LCA，枚举去掉的关键节点并使用前后缀LCA算出剩余节点的LCA比较权值即可。

C A J H F D B G I E

- 题意：
  - 给定一个城市有若干十字路口，右转需要等红灯，直行、左转和掉头都需要，求起点到终点最少等几次红灯

把每条路看做点，十字路口处连边，形成一个边权为0/1的有向图。
最简单的做法是dijkstra求最短路。
也可以用BFS解决，注意用一个deque维护队列。
没有刻意卡spfa。

- 题意：
  给出长度为n的小写字符串A和k个长度为m的小写字符串$B_1 \ldots B_k$，$B$的每个位置拥有统一的权值$v_1 \ldots v_m$，对于每个$B_i$求最大和区间满足该区间构成的字符串是A的子串（空区间合法）。

- 题解：我们可以将问题进行转化，相当于对$B_i$的每个位置求出它作为结束位置在A中的最长子串长度，然后在该区间求最大子段和，所有位置的最大值即为答案。对于每个位置的最长子串，可以对A建后缀自动机，然后$B_i$从左往右在A的后缀自动机上转移，如果当前节点无法转移跳至父亲节点，最后无法转移则长度为0，转移成功则为转移前节点的最大长度+1。

# J H F D B G I E

- 题意：给定一个无向图，每次询问两点x, y，求是否存在一个n的排列，使得第一个元素为x，最后一个元素为y，且排列的任意一个前缀、任意一个后缀都连通。

- 该题意等价于询问是否存在一个以x、y为极点的双极定向。双极定向存在，当且仅当添加一条边(x, y)以后图是点双联通的。关于图的双极定向问题，可以自行查找资料了解。

J H F **D** B G I E

- 题意：给定一棵树和一个起点，1号节点为终点，随机选其中K条边变成指向终点的单向边，在树上随机游走，求到达终点的期望步数

- 我们知道，从任意一点出发，随机选择一条边移动，在树上移动到其父亲的期望步数为2 * 子树大小-1，移动到根，就是每一条边的期望步数之和。

- 之后考虑单向边的影响，单向边对于其下方的子树没有任何影响，但是会使上方的节点的子树大小减少（因为是单向边，不能走到这里，相当于没有这个子树）。

- 那么我们考虑每个单向边的贡献，如果一条单向边与上方的边之间的没有其它单向边，全都是双向的，那么那么这条单向边对于期望的贡献就是 -2*子树大小。那么我们也要计算产生这种情况的概率作为系数，发现这个概率只与两条边之间的距离有关，所以我们只需某一距离有多少对边即可。发现一条边可以产生的贡献的距离是连续的，所以直接用区间加法统计即可。

D B G I E

- 题意：给两个凸包，各有速度，求相撞时间。
- 去掉一些特殊情况以后，相当于有一个静止凸包，另一个凸包向一个方向移动，求是否会相撞以及相撞时间。
- 有多种方法解决这个问题。首先可以明确的是，相撞时一定是某个凸包的顶点与另一个凸包相撞。
- 根据速度的法向量将凸包分成上下凸壳，枚举另一个凸包顶点，直接lower_bound或者扫描线判断其与凸壳的交点。
- 也可以二分移动距离，将一个凸包拉长，然后求两凸包的闵可夫斯基和判断是否相交。
- 二分加闵可夫斯基和的做法常数非常大，且有很多精度问题。标程使用set耗时132ms，使用扫描线耗时52ms。如果出现了卡精度和卡时间的问题，可以尝试优化一些细节来解决。

# H F D B G I E

- 题意：需要把N个人派遣到K个城市，每个城市需要的人数是固定的。把不同的人派遣到不同城市，代价都是不同的，求最小代价。
- 一个比较朴素的想法就是最小费用最大流，但是复杂度明显是不对的，点数1e5，边数高达1e6。

- 但是我们注意到，增广时，每次的流量只有1，因为我们把N个人都看成是不同的。同时这个图只有两层，将一个已经分配好的人换到另一个城市就意味着走一条反向边，再走一条正向边，我们发现这个过程就是复杂度的瓶颈。所以我们考虑把这个繁琐的过程简化，用一条边直接代表正向边+反向边。

- 这样我们的图上一共建立K个点代表K个城市，我们直接把N个人先分配到第1个城市，这样其它城市再从第1个城市抢人。对于每个人，我可以从当前所在城市变换到其它任意城市，代价就是两者之差，所以一共要建K条边。

- 这样我们就发现，图上有很多代价不同的重边。用K^2个堆维护两点之间的重边，跑SPFA（最小费用），每转移一个人重新维护一次边表。
- 复杂度 O(N * SPFA(K^2) + N K log(N))

D B G I E

- 有 $n$ 杯饮料，每杯饮料有对应的加料，现在把加料随机放入不同的杯子
- 设 $x$ 表示加了正确的加料的杯数，求 $x^k$ 的期望，对 862118861 取模

- 实际上答案必定是整数，并且答案就等于
- $\sum_{i=0}^{n} {k \brace i}$
- （后面再证）

- 如果 $k \le n$，答案就是贝尔数的第 $k$ 项

# D B G I E

- 注意到模数等于 $857 \times 997 \times 1009$，因此我们可以借助 Touchard 同余计算贝尔数
- Touchard 同余：
- $B_{n+p} \equiv B_n + B_{n+1} \pmod p$，$p$ 是质数
- 用 $O(k^2 \log n)$ 的科技加速一下，最后中国剩余定理

- 如果 $k > n$，可以先算出贝尔数，然后把多的斯特林数减掉
- 注意到 $k - n$ 并不大，所以可以套用第二类欧拉数的等式：
- $\left\{ {x \atop x-n} \right\} = \sum_{k=0}^{n-1} \left\langle\!\!\left\langle {n \atop k} \right\rangle\!\!\right\rangle \binom{x+n-k-1}{2n}$
- $\left\langle\!\!\left\langle {n \atop k} \right\rangle\!\!\right\rangle$ 是第二类欧拉数，稍后再说
- 预处理一下阶乘之后用 Lucas 定理算组合数

## D B G I E

- 第二类欧拉数直接用递推公式计算：
- $\left\langle\!\!\left\langle {n \atop k} \right\rangle\!\!\right\rangle = (2n-k-1)\left\langle\!\!\left\langle {n-1 \atop k-1} \right\rangle\!\!\right\rangle + (k+1)\left\langle\!\!\left\langle {n-1 \atop k} \right\rangle\!\!\right\rangle$

- 设 $q_i$ 是模数的任意一个质因子，复杂度为
- $O(\sum q_i^2 \log n + (k-n)^2)$

- 附一个简略证明:

$$
\begin{aligned}
n!E\left(x^k\right) &= \sum_{i=0}^{n} \binom{n}{i} i^k d_{n-i} \\
&= \sum_{i=0}^{n} \binom{n}{i} d_{n-i} \sum_{j=0}^{k} \binom{i}{j} \left\{ {k \atop j} \right\} j! \\
&= \sum_{j=0}^{k} \left\{ {k \atop j} \right\} j! \sum_{i=0}^{n} \binom{n}{i} \binom{i}{j} d_{n-i} \\
&= \sum_{j=0}^{k} \left\{ {k \atop j} \right\} j! \sum_{i=j}^{n} \binom{n}{j} \binom{n-j}{i-j} d_{n-i} \\
&= \sum_{j=0}^{k} \left\{ {k \atop j} \right\} \binom{n}{j} j! \sum_{i=0}^{n-j} \binom{n-j}{i} d_{n-j-i} \\
&= \sum_{j=0}^{k} \left\{ {k \atop j} \right\} \binom{n}{j} j!(n-j)! \\
&= \sum_{j=0}^{\min\{n,k\}} \left\{ {k \atop j} \right\} n!
\end{aligned}
$$

$j \le k$ 是显然的（否则斯特林数为 0），因此答案就是

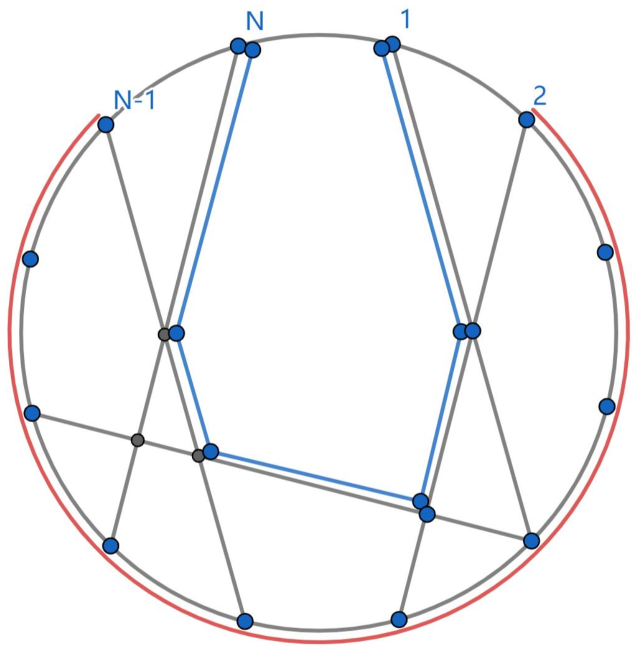
$$E\left(x^k\right) = \sum_{i=0}^{n} \left\{ {k \atop i} \right\}$$

# BGIE



- 题意：给一个圆环，上面有一些等长的弦，要在环和弦上放两根不相交的电线，分别1~N,2~N-1。电线必须沿着从小到大的顺序，可以在弦交点处转弯。

- 如果我们求出所有交点，并以交点和圆环上的点为节点建图，可以发现这是一个DAG。
- 假设两根电线在某一位置相交，我们在它们最早相交的位置，交换它们的后半部分，可以发现这样两根线就分别从1~N-1,2~N了。所以任何相交的方案与直接错位连接的方案是一一对应的。问题就变成了容斥。
- 记u走到v的方案数为f(u,v)，则答案就是f(1,N)*f(2,N-1)-f(1,N-1)*f(2,N)

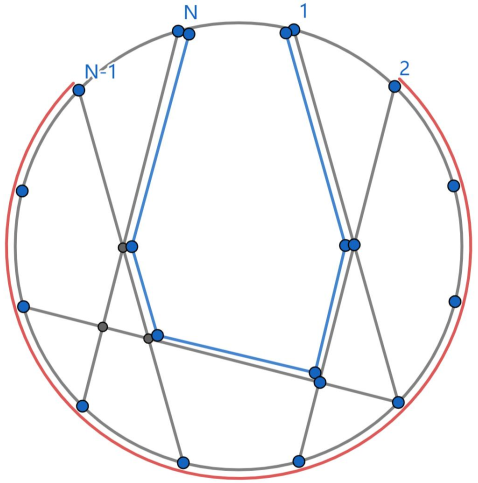


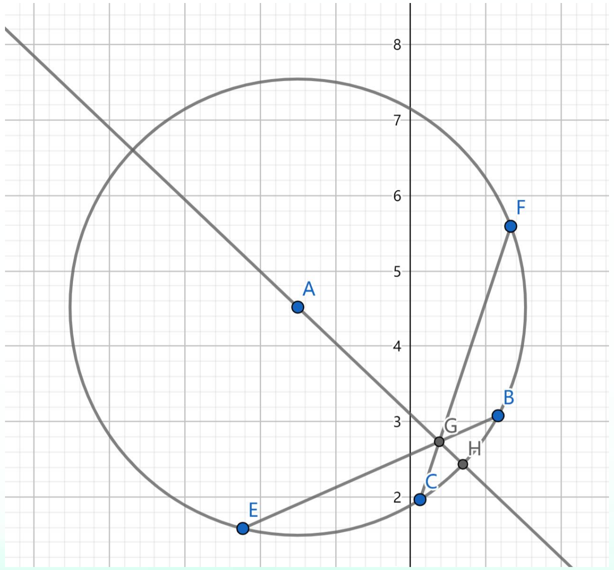
# B G I E

- 不难发现这个圆环的弦有一些性质
- 两弦相交的顺序与弦的起始位置正相关。
- 因为弦长相等，我们可以画出它们的对称线，刚好穿过它们的交点和圆心，所以随着EB上下滑动，H也是BC的中心，G就是AH和CF的交点。
- 所以判断两弦是否相交，以及相交在什么位置（我们不关心具体位置，只关心顺序）就很简单了。

- 之后我们假设初始时只有一个圆环，按顺序逐步加入弦，另f[i]表示当前第i个弦的最后一个交点的方案数，dp即可。

- 复杂度 O(N + M ^2)



牛客竞赛
AC.NOWCODER.COM

NIO

# THANKS！

AC.NOWCODER.COM