gift

subtask2

对于一个确定的排列 , 最优的交换策略一定是把相关的元素用一个媒介来进行交换

也就是说把 a_i,b_i 连边之后 , 形成的环内进行交换

所以最小交换次数就是 n 减去环的个数

subtask1

$O(n!)$ 枚举之后 , 用 $subtask1$ 的方法统计一下答案就好了

subtask3

全是 0 的数据 , 我们可以任意地构造环

用 n 个点构成 m 个环的方案数显然是 $s(n,m)$, s 是第一类斯特林数

由于每一个点的取值有 n 种 , 所以答案就是 $s(n,m)*n!$

subtask4,5

对于有已知数字的情况 .

这些点要么构成环 , 要么构成链 , 要么就是 $0-0$ 的自环 , 已知的环的情况可以不用考虑了 , 只会在统计答案时用到.

对于链我们只需要考虑数量即可

但是在 $x->0$ 和 $0->x$ 合并时 , 需要借助 $0-0$ 边 , 放在一起考虑不方便.

所以我们把先把链分为 $x->0$ 和 $0->x$ 两种分开来考虑 , 合并成若干个环 .

我们可以先分别把同种链内部合并 , 构成若干个环 , 构成恰好 i 个环的方案不太好算

考虑容斥 , 设 $f[i]$ 表示至少有 i 个环的方案 .

枚举用 j 条链来合并 , 设 m 为 $0-0$ 这样的自环的个数 , n 表示这一类边的数量.

$f[i]=\sum_{j=i}^{n}C[n][j]*S[j][i]*A[m+n-j][n-j]$

大意是从 n 条链中选出 j 条 , 合并成 i 个环 , 并且剩下的 $n-j$ 条任意连边 , 可以和 $0-0$ 边相连 , 也可以连成自环 , 或者和其他的链相连 .

然后容斥求出恰好 i 个环的方案数就行了.

然后把 $0->x$ 和 $x->0$ 的自身构成环的方案数卷起来 , 得到两种边内部构成 i 环的方案数 $g[i]$.

剩下的就是考虑 $0->x$ 和 $x->0$ 边构成的环 , 因为这两种链合并需要借助 $0-0$ 边 , 于是可以和 $0-0$ 边一起考虑.

只需要把 g 和$0-0$ 内部构成环的方案数卷起来就行了 , 这样就把 $0-0$ 边内部形成的环和 $0->x$ 和 $x->0$ 合并的方案同时考虑完了 .

复杂度 $O(n^2)$.