Solution

dy0607

August 2, 2018

1 Lighthouse

题目实际上求的就是哈密顿回路的数量。由于m很小，考虑容斥。

枚举删除的边的某个子集S, 设f_S表示有多少条哈密顿回路**至少**包含S集合中的边，答案就是$\sum_S (-1)^{|S|} \times f_S$.

怎么算f_S呢？首先判掉$f_S = 0$的情况，这包括仅考虑S中的边时，某个点的度数大于2；以及出现了环，并且这个环大小不为n.

特判掉S本身就是一个哈密顿回路的情况; 假设S中的边构成了k条链，那么$f_S = 2^{k-1} \times (n - |S| - 1)!$。

证明考虑将一条链看成一个点，那么总共有$n-|S|$个点，其环排列方案数为$(n-|S|-1)!$；每条链都可以翻转，因此乘上2^k；又由于一条哈密顿回路对应了两个环排列（正反两个方向），还要除以2。

$O(2^m \times m)$

2 Miner

`Source: Codeforces Round #375 (Div2) E`

把原题改了下，改简单了。。

题意就是加上尽量少的边，使图中存在欧拉路径；也就是用尽量少的路径，覆盖所有边恰好一次。假设图中有k个连通块，第i个中有c_i个度数为奇数的点，那么答案为

$$\sum_{i=1}^{k} \max(1, \frac{c_i}{2}) - 1$$

构造方法也很简单，将度数为奇数的点任意配对连额外的边，然后每个连通块跑欧拉回路，额外连的边会将回路割成若干段路径，这些路径之间以及连通块之间用1操作跳即可。

$O(n+m)$

3 Revive

对于一些带平方的式子，把平方拆开有时会很有用，比如这道题（$Path(u, v)$表示u到v的路径上的边的编号集合）：

$$
\begin{aligned}
ans &= \sum_{u,v\in[1,n],u<v}(\sum_{e\in Path(u,v)} w_e)^2 \\
&= \sum_{u,v\in[1,n],u<v}(\sum_{e\in Path(u,v)} w_e^2 + \sum_{a,b\in Path(u,v),a<b} 2\times w_a\times w_b) \\
&= \sum_{e\in[2,n]} w_e^2\times \text{（经过e的路径数）} + \sum_{a,b\in[2,n],a<b} 2\times w_a\times w_b\times \text{（同时经过a, b两条边的路径数）}
\end{aligned}
$$

经过e的路径数就是e两边子树的$size$之积，主要是计算经过两条边的路径数。

3.1 Solution1

点分治，每次考虑所有路径经过重心P的边对的贡献，若i在以P为根时在整棵树上的$size$为$sz(P,i)$，那么经过a,b的路径数为$sz(a,i)\times sz(b,i)$. 记录重心的每个子树的$sz(P,i)\times w_i$的和就可以方便地计算了。

修改时需要把分治结构存下来，对于每个包括这条边的分治重心重新算一下其所属子树的贡献即可。$O((n+q)\log n)$.

3.2 Solution2

这个方法是写Solution时才发现的，并不知道速度如何。

枚举一条边a，讨论b的位置：在a的子树内；在a到根的路径上；以及其他情况。对于每种情况可以分别用线段树计算。

第二种情况会涉及链查询，速度比较慢，注意我们只需要单点修改，所以线段树上可以直接维护每个点到根的路径上的和；这样就变成区间修改和单点查询了。

复杂度为$O(n+q\log n)$，但修改常数较大。