BUAA Summer Practice 3 简单图论

Problem A：BipartiteGraph

Tags：分治、并查集

Solution：

注意到每次删去一个点，对全图的影响很小，考虑采用分治算法：首先加入1~$n/2$的所有点，处理右半边（删去一个点的情况）；再恢复到初始情况，加入$n/2+1$~n的所有点，然后处理左半边。两个半边的问题与原问题同质、规模减半。因此$T(n)=2T(n/2)+F(n)$，只要恢复到原始状态并加入另一半点的复杂度F(n)足够好，总问题的复杂度就是可以接受的。

使用并查集处理加入点和边以及恢复的操作。由于需要判断是否是二分图，并查集上需要维护距离。恢复操作可采用可持久化或暴力恢复，暴力恢复即记录每次修改的位置、修改前后的值，在恢复时做相反操作即可。经测使用rope实现可持久化并查集会tle，暴力恢复+不做路径压缩+启发式合并可通过。

Problem B：Yu-Gi-Oh!

Tags：二分图最大匹配

Solution：

由于每次使用两类卡各一张进行合并，可建立二分图，边上权值表示合并两张卡可获得的收益。为表示不合并的情况，对于每张卡，在另一个类别中建立一个虚点，只与该卡连边，边权值为该卡不合并时带来的收益。建立二分图模型后，使用KM算法或网络流均可解决。

Problem C：Flight

Tags：树的性质

Solution：

注意到答案只可能为$-1,1,2,3$。设树的直径已经找到，为AB。出发点为x、y。

答案为1的情况：两点间距离大于等于d。

讨论需要2或3步的情况。由于AB为直径，A、B中一定有一个点距离出发点的距离最远。假设到x、y距离最远的点相同，不妨设为A，则x->A->y要么可行，答案为2；要么不可行，答案为-1。假设到x、y距离最远的点不同，不妨设x对应A、y对应B，则x->A->B->y要么可行，答案$<=3$，要么不可行，答案为-1。

考虑x、y分别对应A、B但需要两步的情况，此时存在中间点T，$d(x,T)>=d$ && $d(T,y)>=d$。设x、y间的路径与直径重叠部分的端点为X、Y，可以证明，该节点要么存在于X、Y节点之间的直径的分支上，要么存在于A或B，方法还是利用直径最长的性质，简单讨论可发现，如果中间点T不在上述位置上，则可以构造出比直径更长的路径。查询端点X、Y之间是否存在满足条件的点是简单RMQ问题。考虑T距离A比较近（T距离x比较近）的情况。设P为x到y路径的中点，不妨设T所在子树根节点为Q，则$d(T,y)<=d(T,x)<=d$，等价于$d(T,Q)+d(Q,X)+d(X,x)<=d$，等价于$d(T,Q)+d(Q,A)-d(A,X)+d(X,x)<=d$，等价于$d(T,Q)+d(Q,A)<=d+d(A,X)-d(X,x)$，该式左侧与$x,y$无关，右侧仅与$x,y,d$有关，因此每次询问是一个RMQ问题。至此完成了所有情况的讨论。

Problem D：Asa'sChess Problem

Tag：带上下界最小费用流

Solution：

每行、每列建点，注意到每次交换要么共行，要么共列，因此至多只会在行列之一上产生影响。在受影响的行（列）之间建边，不妨从1减少的一方到1增多的一方，上界为1、下界为0、代价为1。从源点S向每行每列连边，上界、下界都为该行初始1的个数，代价为0。从每行、每列向终点T连边，上下界为该行（列）1的个数的限制，代价为0。答案即为在满足上下界限制的条件下的最小费用流。

求解上下界最小费用流需要进行转化。另建超级源、汇SS、TT。对于有费用下界的边，从SS向终点连代价为该边代价、容量为下界的边；从起点向TT连代价为0、容量为下界的边。该边容量变为上界-下界。另外，连接原有源汇T、S，以保持流量平衡。在该图上进行普通最小费用流，最后检查从SS出发的各边是否满流，满流即为存在解。

E CRB and Graph (hdu 5409)

如果一条边删除后使得无向图不连通，那么这条边是桥边。

于是我们用tarjan算法求出桥边，把每个双连通分量缩点，会得到一棵树，树上的边都是桥边。此时我们只需在缩点的树上看这个问题即可。

树上删除一条边会把树分成两棵子树，我们记每个子树包含的点的最大编号分别为m_1 和m_2 ，则 $m_1 == n || m_2 == n$ ，容易想到如果m_1 不为n 则答案为$(m_1, m_1 + 1)$ ，m_2 同理。所以我们只需以包含n 的分量为根，求出每个子树的最大编号即可。

F King's Pilot (hdu 5644)

现在有n天， 每天需要p_i 个飞行员，一个飞行员正常只能飞行一次。最开始有k个飞行员。可以花费P天Q元来新训练一个飞行员。还有m个方案，可以让飞行员在飞行后T_j天花费S_j元重新飞行一次。问最小费用

最小费用流 首先从第i天连向汇点，费用为0，容量p_i。然后从源点连向第1天，费用为0，容量为k; 第i天向第$i+1$天连一条容量无限，费用0的边， 表示之前剩下的飞行员都可以之后使用。新训练的飞行员可以直接从源点连到P一条容量无限，费用Q的边。每个方案可以把点拆开，每天都会有p_i个飞行员飞过，可以从源点向i'连一条费用0， 容量P_i的边，再从i'连一条边到$i + T_j$，费用S_j。之后跑费用流即可

G Red-black Cobweb (cf 833D)

树上静态计数问题容易想到树分治。

难点在于此题的条件比较奇怪，要求路径上红边和黑边的差距不超过2倍，我们令红边数量为x ，黑边为y ，则条件为 $\frac{1}{2} <= \frac{x}{y} <= 2$。

我们记分治过程中根为u ，子树中点的信息用一个三元组(r, b, w) 表示，分别代表此点到u 路径上红边个数，黑边个数以及权值乘积。我们顺次处理每颗子树，在处理一颗子树中的某个点时需要访问之前处理过的子树的信息，令已处理过的某点信息为(R, B, W) ，当前点的信息为(r, b, w)，需要满足的条件为

$\frac{1}{2} <= \frac{r+R}{b+B} <= 2$

如果我们知道满足此条件的(R, B) 个数C 以及所有(R, B) 对应的W 的乘积，则当前点的答案为$(\prod W) * w^C$ ，那么问题转化成在线地询问以上信息。

把不等式右半边展开可以得到 $r - 2b <= 2B - R$ ，这个可以用树状数组维护，同理左半边也可维护，得到答案只需要用比值$>= \frac{1}{2}$ 的除以比值> 2 的即可，这里需要做一下逆元处理。

这样我们可以得到$\prod W$ 。得到C 只需要维护另一个操作为$"+"$ 的树状数组就好了。

H Tree (cf 468D)

给一棵树，求一个全排列p_i， 使得$\sum dis(i,p_i)$ 和最大， 并且找一个字典序最小的p_i。

问题转换：

$\sum dis(i,p_i) = \sum(dep_i + dep_{p_i} - 2*dep_{lca(i,p_i)}) = 2*\sum dep_i - 2*\sum dep_{lca(i,p_i)}$

即求一个匹配方案， 使得lca距离和最小。

考虑把重心当成根，则每一枝不会超过$n/2$个，所以可以为每一个节点找到不同的匹配点，使得lca都为根，即最后一项为0， 所以和最大即直接为$2*\sum dep_i$。下面考虑如何求字典序排列最小的方案。

首先对于每一个点来说，只需要选择一个不在当前这颗子树的节点即可。所以直接按照顺序来求$p[1]$~$p[n]$。

假设当前需要为i来找$p[i]$， 设所在子树为s，子树内有x个大于i的，即还有$x+1$个未进行匹配，即没有匹配对应的$p[i]$。而子树中还有t个未被匹配，即$p[1]$~$p[i-1]$中还没有出现这t个点。当前总共没被匹配的有$n-i+1$个，而子树内需要保证剩下的都有其他子树的节点来匹配，即$t \le n-i+1-(x+1)$，即$t+x <= n-i$。 所以只需要满足这个式子即可。所以维护一个$t+x$的集合，如果有等于$n-i$的，就必须选择一个该子树内的最小的点，如果没有则直接选取全局非自己子树内的点即可。注意根可以和任何节点匹配，包括自己。

I Transmigration tree (hdu 5436)

给一棵树，点上有权值，现在把根节点和所有叶节点连一条边，构成一个图，m个询问，u到v的路径中权值和最小的，且权值最大值最大的路径的权值和还有最大权值。

不考虑根连向叶节点的边，可以直接用树链剖分或者lca等计算；加上这个边以后，就会有更多情况：

- 直接从树上走
- u->叶节点->根->v
- v->叶节点->根->u
- u->叶节点->根->叶节点->v

所以需要维护每个点到叶节点的值，然后四种情况取最小即可。

坑点：点到叶节点不一定一直向下走，可以先向上走，然后再走到另一支的叶节点。

J Garden (cf 152E)

由于$k <= 7$，我们状压一下。令$dp[i][j][S]$ 表示走到坐标为(i,j) 这个格子，已覆盖的重要格子集合为S 的最小费用。转移是比较直接的：

(i,j) 转移到周围格子， $dp[i'][j'][S] = dp[i][j][S] + cost[i'][j']$

对于$S' \cap S = \phi$， $dp[i][j][S'|S] = dp[i][j][S'] + dp[i][j][S] - cost[i][j]$

但是直接转移是不行的，成环违背了dp 的限制。

如果把每个状态(i,j,S) 看成一个点，转移看成边，那么只需要在这张图上跑一边最短路就可以把所有状态算出来，点和边的数量都是$O(n*m*2^k)$ 的，可以接受。

K Birthday (cf 590E)

给n个字符串，让选出m个串，要求这m个串种不能有两个串有包含关系。求最大m以及这m个串的id

首先通过AC自动机等建图，求出包含关系。可以求出一个关系图。建图的时候考虑按照包含关系建图，即可以得到一个DAG，然后就是求最大独立集，可以直接拆点做二分图匹配来求。

Dilworth定理：**最小链覆盖数 = 最长反链长度**

所以转化为最小链覆盖数->最小路径覆盖

最长反链证明：http://vfleaking.blog.163.com/blog/static/1748076342012918105514527/

L The King's Problem (hdu 3861)

如果(u, v) 互相可达，那么它们在一个块里。直接强连通分量缩点，这个条件就不用考虑了。

考虑缩点以后的图是无环的，问题转化成把这个图分成最少几个块，使每个块是弱连通的。由于每个弱连通分量必定有一条路径可以作为它的生成树，那么问题转化为最小路径覆盖，直接求解即可。