**tree**：

首先不难发现，如果C:\Users\Lenovo\AppData\Local\Temp\SGPicFaceTpBq\9528\1419A71C.png某一步是朝着t走的，那么一定不会走回来了；否则，C:\Users\Lenovo\AppData\Local\Temp\SGPicFaceTpBq\9528\1419A71C.png一定会把整棵子树走完再回到当前点。

走到一个s到t路径上的点时，C:\Users\Lenovo\AppData\Local\Temp\SGPicFaceTpBq\9528\1419A71C.png会随机选择一个儿子走下去，这等价于随机一个儿子们的排列，然后按这个顺序走。

那么s到t的路径上所有点显然一定会走到，以s为根时t子树中的点显然走不到，而其它点都有1/2的概率会走到。

时间复杂度O(nlogn+m)。

**prime：**

n较小时，我们可以直接用线性筛/埃氏筛法求出每个数的最小质因数。

我们考虑进行容斥。对于每个质数x，我们需要求出1~n/x中不被比x小的质数整除的数的个数。一种简单的思路是，对于x<=k的情况，我们进行常见的枚举子集容斥；对于x>k的情况，n/x较小，我们就在n/k的范围内进行线性筛/埃氏筛法。

注意到进行子集容斥时，枚举子集后贡献形如(-1)i(n/S)，而n/S只有O(sqrt(n))种取值，我们可以对这个进行记忆化。

时间复杂度O(n3/4/sqrt(logn))

**path：**

这是课件里的例题。

点分治统计树上的情况，然后单独考虑经过剩下那条边的答案。

在环上按顺序枚举一个端点，用树状数组维护另一个端点到这条边的距离。

时间复杂度O(nlogn)