set

subtask1：暴力模拟

可以证明(x的第k对应点)等价于((x的第k-1对应点)的第2对应点)

由此联想到模型的转换： x的第2对应点 -> fa[x] 注意到可能不存在第2对应点，因此这是一个森林，新建超级根结点R，不存在第2对应点的都连到R x,y的最优公共对应点 -> lca(x,y)==R ? -1 : (lca(x,y)==x || lca(x,y)==y) ? fa[lca(x,y)] : lca(x,y)

因此这个操作2在一定程度上等价于查询x到y路径上的点权和（不等价的边角情况需要特判） 而操作1等价于修改x和其所有儿子的点权

subtask2：树剖直接实现

暴力树剖显然被菊花图卡到生活不能自理，但是只要冷静分析一波就能解决这个问题 把原树复制一份，新树的结点x的权值（开始全为0）是结点x的儿子增加的值 真实权值和=原树的路径和+2*新树的路径和-新树上x和y的权值+新树上lca(x,y)的权值+新树上fa[lca(x,y)]的权值

实际上这两棵树长得一模一样，所以并不需要真的开两棵树，只需要记两种值就行了；当然也可以把原树的权值和静态维护，只对新树用树剖

修改操作$O(\log n)$，查询操作$O(\log^2 n)$，有一定常数

all：树剖，修改操作通过标记实现

blue

一句话题解（90pts）：蓝超巨星（**b**lue **s**uper**g**iant **s**tar）

我们可以尝试先确定答案的界

注意到映射是双射，这表明将映射表化成一张图的话它将由若干个不相交的简单环组成

循环左移变换的周期显然是n，映射变换的周期应该是环长的lcm

那么最终答案的周期是两个周期的lcm

写个搜索可知在字符集大小为26的情况下映射变换的周期最大为1260

即答案的界大致是$1260n$

通过一些预处理，任意次数的递进操作可以在$O(n)$时间实现

subtask1：暴力枚举答案

subtask3：留给乱搞大师，暴力应该是卡掉了

对于S=T的情形，答案一定是两个周期的lcm的因数（最小正周期是任意周期的因数）

由此，暴力求解这一情形的时间复杂度为$O(n \times \sigma(1260n))$

~~要相信它是跑的过去的~~

subtask2：枚举lcm的因数一一验证

注意到变换存在逆变换，可以应用BSGS求解：方程即为$f^x(S) = T$，令$t = \sqrt{1260n}$，则$f^{ut-v}(S) = T, 0 < u \leq t, 0 \leq v < t$，等价于$f^{ut}(S) = f^v(T)$

故BSGS的时间复杂度为$O(n\sqrt{1260n})$

subtask4：BSGS

两种变换本质上都可以通过同余方程来表达（$ax \equiv \text{起始位置差} \mod n$，$bx \equiv \text{环上位置差} \mod \text{环长}$）

同时可以发现，如果我们确定了起始位置差，那么应当满足两个条件：1.S和T对应的位置所在的环相同；2.对于每个环，环上所有的位置差相同。

条件2可以对每个环独立差分来表达，具体而言就是对每个位置与其同一个环上的前驱做差分

那么按照这两个条件对两个pair<int,int>的串做匹配就可以求出所有可行位置

对于每个可行位置，解由26+1（每个环1个+起始位置1个）个同余方程组成的方程组即是答案

匹配次数最多是$O(n)$的，故时间复杂度$O(27n\log n)$

注意S=T的情况应该输出正整数解

all：KMP匹配，对命中位置解同余方程组

secret

考虑序列 K 中某个元素的质因数分解式，可知其不在质数序列 S 中的项没有意义，在 S 中 的项的指数如果大于 1 也没有意义。简略证明：gcd、lcm、对质数的整除性这三个要素都 只关心某些质因子的存在性，而不关心具体存在几个。

subtask1：直接搜每个元素的质因子组成

subtask2：元素的质因子组成不全相同是不优的（调整法），贪心

subtask3：状压DP等乱搞

在上面结论的基础上可以进一步发现组成同一个元素的不同质因子之间也没有什么联系。

基于此，我们实际上是在对 10n 个元素分别决策（取/不取），并希望最小化代价；而决策只有两种，相当于将 10n 个元素划分成两个集合。

考虑最小割。选择代价和弃用代价分别是属于两个集合要付出的代价，差异代价是某两个特定元素不属于同一个集合要付出的代价。这是一个经典模型，没见过的可以参见 POJ 3469。

但最小割的优化目标是和最小，不能处理积最小的情况。

70%（结合前2个subtask为82，结合前3个subtask为88）：如上所述跑最小割

对输入的代价数据全部取对数后再运用上述做法即可。输出答案的时候要用 exp 之类的函数乘方回去。

all：取对数后最小割

---

end.

这套题的准备时间十分仓促（~4days），如有错漏之处请指正，谢谢~
