T1

线性筛出 $\varphi(i), i^p$，然后线性求逆元即可。复杂度 $O(n\log_n p)$。

T2

考虑从大到小插入每个点，于是只需要知道它有多少个位置可以插入（即“空位个数”s），如果当前加入的点没有被钦定是叶结点，那么它不管插入在哪里都会让 s 恰好增加 1；如果当前插入的点被钦定了是叶结点，那么它不管插入在哪里都会让 s 恰好减少 1。

于是会发现，从大到小插入时每个点能被插入的位置数量都是固定的，乘起来即可。复杂度 $O(n)$。

T3

考虑一个暴力的 dp：$f(i,S)$ 表示当前加入了 i 条边，形成 SCC 的大小构成的集合为 S 时的方案数（S 事实上是个 n 的划分数）。

考虑转移。如果当前这条边被加在某个 SCC 内部了，那么集合不会变；如果被加在外部了，可以暴力枚举缩起来了哪几个 SCC，然后转移。

注意到转移的条件是什么。对于加在 SCC 内部的情况，你需要让当前存在边的空位。即 $i <$ 形成 S 中 SCC 所需要的至少的边 $+$ 添加尽量多的边使得 SCC 集合不变化。对于加在外部的情况，假设缩起了 k 个 SCC，就需要 $i+1 \ge k+$形成 S 中 SCC 所需要的至少的边。

于是会发现上面似乎还少了一维状态 j 表示形成 S 至少需要的边（这是因为当前状态下形成 S 至少需要的边不一定就是 S 中的环数，可能有某个 SCC 是通过若干个更小的 SCC 连接而来的）。

而我们会发现，第二种转移和 S 已经毫无关系了（只和 SCC 的个数有关），第一种转移需要让能够添加的边更多，那自然的方案就是除了一个 SCC 的大小为 $n-|S|+1$ 以外，剩下 $|S|-1$ 个 SCC 的大小都是 1。于是可以设计更简单的转移：

$f(i,j,k)$ 表示加入了 i 条边，形成当前状态的 SCC 至少需要 j 条边，有一个大小为 k 和 $n-k$ 个大小为 1 的 SCC 时的方案数。

注意到第二维是 $O(n)$ 的，因此总状态数是 $O(n^4)$ 的。暴力转移可能需要枚举当前缩起来了多少个 SCC，但是这可以前缀和优化掉。

于是总复杂度 $O(Tn^4)$。

T4

设所有数字的异或和为 s，那么显然平局当且仅当 $s=0$，于是接下来考虑 $s\ne 0$ 的情况。

找到 s 的最高位，设为 2^p，不难发现先后手的胜负一定是在这一位上被区分的，于是就完全被转化成了 $0\le a_i\le 1$，且一定有奇数个 1 的问题。

注意到，若 n 为偶数，则要么奇数位上有奇数个 1，要么偶数位上有奇数个 1，并且先手可以任意控制后手选的位置的奇偶性，于是先手一定可以拿到奇数个 1，先手必胜。

若 n 为奇数，且两段都是 0，那么先手不管取哪一个都会转化成 n 为偶数且有奇数个 1 的情况，先手必败。

于是先手第一步必然会取两端中的某个 1，然后不管后手取哪一个，先手都必须取和它一样的元素，否则就会转化成必败态。

于是贪心即可，先把两端相等的全部删掉，然后剩下的位置必须形如 001111...（即相邻奇数位偶数位必须相同）先手才有可能胜，否则后手一定必胜。

此外，如果 1 的个数不是 4 的倍数，先手会和后手把数字全取完，由于先手必须每次都和后手取相同的元素，因此先手会取奇数个 1，再加上最先开始的 1，总共取了偶数个 1，输了。于是还需要满足此时 1 的个数是 4 的倍数，否则全是后手胜。

于是扫一遍即可，复杂度 $O(\sum n)$。