T1：

算法一：

直接暴力算出交点，排序。期望得分 10 分

算法二：

算交点前，二分出一个时间，使得前 m 次超越的时间均小于它。

二分时，将二分的值带入直线求出走的路程，将路程排序，每一条直线原来的位置减去现在的位置就是它超过了多少条直线减去被超过的数量，将超过了一定数量的直线的贡献加起来，这个数是小于实际值的，但是也能够二分出一个优秀的时间，总超越数小于 m 即可。

然后将满足条件的交点丢入小根堆中，弹出前 m 个。排序求 k 大。

期望得分 30 分

算法三：

在算法二的基础上，运用二分套二分求 k 大。

期望得分 50 分

算法四：

求超越与被超越，用堆维护当前时间下每只脚之前和之后的脚，因为一定是与最近的脚相遇(可能被后面的超过，也可能超越前面的)，超过之后，更新一下，用堆调整即可。答案用堆存，保证正确性。用分数类来表示时间，避免精度误差。

排序求 k 大

期望得分 50 分

算法五：

结合算法三与算法四，期望得分 100 分。

T2:

注意到模数为 1e9+9，那么根号 5 在该模数下存在两个数使得平方后取模等于 5。

$$Fib(n)=\frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^n-\left(\frac{1-\sqrt{5}}{2}\right)^n\right]$$

带入后交换和号，用二项式定理展开计算，式子最后的一部分等比数列求和即可，

注意讨论公比为 1 的情况。

第二个式子先莫比乌斯反演，再用杜教筛即可。

T3.

算法一：

暴力枚举，期望得分 20

算法二：

最小生成树计数考虑按边长排序后用矩阵树定理，具体可以参考 BZOJ1016

复杂度应该是 O(N^5)

算法三：

对于 a[i]=0 的数据，运用 cayley 定理即可

算法四：

我们可以按照最高位的值，将点分为两种：S(为 0 的)，T(为 1 的)

有个结论，连边方式肯定是，S 构成一棵树，T 构成一棵树，S 到 T 连一条边

证明：假设 S,T 之间有多条边，随便删去一条，然后 S 变成两个联通块或者 T 变成两个联通块，之后从 S,T 内部选边，边权的最高位肯定为 0，代价肯定比较小。

于是我们可以利用字母树，贪心寻找 S 到 T 最小的边，然后分别递归下去

时间复杂度 O(nlog^2n)