T1：

不难发现，就是计算区间内颜色种数，并判断是否存在一种颜色出现的位置等距。

我们把询问离线，按左端点排序，每次移动左端点，重新设置左端点的颜色 c 对答案的影响（右端点在什么范围内会包含 c，以及在什么范围内 c 出现的位置不等距）。重新设置需要撤销之前的，并添加现在的贡献，这个用线段树/树状数组。

T2：

F[i][j]表示 n=i，p=j 时的答案，F[k][1]=k!。

考虑在 n 的排列中插入元素 n+1，原有集合保持不变，而带有 n+1 的集合，可以通过 n+1 的位置计算。$O(n*(n-k)*p)$。

然后 O(n)的转移可以前缀和优化到 O(1)。$O((n-k)*p)$。

对于 p，我们 O(n)算出上界，p 超出上界直接 puts(“0”)，而上界最大是$\frac{1}{6}n^2$（此时 k 为$\frac{1}{3}n$）。$O(n^3)$

然而，非 0 的状态并不多，$k=\frac{1}{5}n$时，状态数最大，仅有$\frac{7}{150}n^3$个。

T3：

例如，我们想算出每条对角线被染过色的格子数，我们不妨计算出未被染色的格子，这个可以通过行列染色情况 fft。

这样我们可以算出每条对角线上，没染过色，没染过绿色，没染过红色的格子数，然后容斥一下。