1　种植(plant.c/cpp/pas)

前 100%的数据:

非常裸的状压 DP，大家应该都会做，PPT 上面也讲过差不多的题（放置国王），这里就不讲了，具体可以参考代码。

2　计数(count.c/cpp/pas)

前 100%的数据:

转化成求[0,R]上各位数字之和能整除原数的数个数。

整除的一般套路是在状态中记录对模数取模的值，但是发现在 dp 过程中并不能确定最后的各位数字之和是多少，所以模数难以确定。

但是发现各位数字之和至多 9*18，非常地小。所以考虑枚举模数 mod，在 dp 过程中记录对 mod 取模的值，即 dp[i][j][k][0/1]表示处理完前 i 位，各位和是 j，对 mod 取模的值是 k，是否压上界的方案数。最终统计答案时仅将 dp[num][mod][0][0/1] 统计到答案中，其中 num 是 R 的位数。

3 **棋盘**(chess.c/cpp/pas)

这道题题解之前用 LaTeX 写过，这里直接用截图方便一些。

1 chess

1.1 20% 的数据

可以发现，第 i 列和第 $i+kn$ 列的棋子数是一样的，于是可以暴力枚举前 $n*n$ 棋子的放法。假设第 i 列放了 t 个棋子，则再乘上 $C(n,t)^{(m-i)/n}$ 即可。

1.2 另外 20% 的数据

由于 $m=n$，考虑直接 dp，$dp[i][j]$ 表示考虑前 i 列，放置了 j 枚棋子。转移时，$dp[i][j]=\Sigma dp[i-1][j-t]*C(n,t)$ 即可。

答案为 $dp[n][C]$。

1.3 另外 20% 的数据

考虑在上面 dp 的转移过程中，将第 n 列后面的列考虑进来。

也就是说，$dp[i][j]=\Sigma dp[i-1][j-t]*C(n,t)^{(m-i)/n+1}$，答案同样为 $dp[n][C]$，复杂度为 $O(n^4\log n)$。

1.4 100% 的数据

预处理 $C(n,t)^{(m-i)/n+1}$，可以少掉一个 log，能通过全部数据。

4　树(tree.c/cpp/pas)

100% 的数据:

注意到一个性质：假设存在一个大小为 k 的联通块，其中含有 a 个黑点，存在另一个大小为 k 的联通块，其中含有 b 个黑点。那么对于任意 a<=i<=b，一定都存在一个大小为 k 的联通块，其中含有 i 个黑点。证明：对于含有 a 个黑点的联通块，我们每次从联通块中删去一个点，再加入一个新的点，形成一个新的大小为 k 的联通块，那么黑点个数至多改变 1；而通过这样的过程我们能够从含有 a 个黑点的联通块逐渐变成含有 b 个黑点的联通块，所以 a 到 b 中任意一个数均会出现。

知道这个性质以后，就是一个简单的树上背包。f[i][j]表示以 i 为根，大小为 j 的联通块（根一定在联通块中）黑点数的最小值；g[i][j]表示以 i 为根，大小为 j 的联通块（根一定在联通块中）黑点数的最大值。

那么即 f[a][j] = min(f[a][j], f[a][j-k]+f[b][k])，g[a][j] = max(g[a][j], g[a][j-k]+g[b][k])。

枚举的时候注意 j 从大到小枚举（应该类似于 01 背包而不是完全背包），然后枚举 k 的时候要注意上下界，确定上下界的准则就是考虑上枚举的黑点数一定不会超过结点总数。具体可以参考标程。

计算完毕之后，可以再用 minn[i]和 maxn[i]分别表示整棵树中，大小为 i 的联通块，含有的黑点的最少数目和最大数目。

于是处理询问的时候，对于询问 x, y，即询问是否存在大小为 x 的联通块，包含 y 个黑点。即判断 y 是否大于等于 minn[x]并且小于等于 maxn[x]即可。