



1. 矩阵游戏

算法一：对于50%的数据：送分，直接一个一个乘，时间复杂度 $O(KN)$ 。

算法二：对于80%的数据：如果我们不一个一个乘，将第 i 行的和乘 x ，第 j 列的和乘 y ，所计算出的结果与正解不同的地方仅仅是 (i, j) 这一个元素。而这样的数不足 K^2 个。所以把这些元素单独计算一遍就可以了，时间复杂度 $O(K^2)$ 。

算法三：对于100%的数据：时间复杂度 $O(M+N)$ ，具体看标程吧。

2. 跳房子

算法一：对于20%的数据：模拟，直接走 K 步，时间复杂度 $O(K)$

算法二：对于40%的数据：走 $M*N$ 步内必有一个循环节。直接走，找循环节，时间复杂度 $O(M*N)$

算法三：对于100%的数据：利用分块思想，一行为一块。用一个数组记录第一列第 i 行走 M 步到达的位置 $jump[i]$ 。在模拟过程中只要一行一行的走，不足一行再一步一步走，按行找循环节，时间复杂度 $O(M+N)$ 。

更改操作：对于每个更改的单元格 (x, y) ，我们回溯到第一列，找到第一列要更新的区间，更新 $jump[i]$ 。因为第一列到 (x, y) 的行是一个连续区间，在找的过程中，只需记录区间上下边界。复杂度为 $O(M+N)$ 。

3. 优美序列

算法三：对于100%的数据：

分治法，离线处理。假设现在处理的询问都包含在 $[L, R]$ 中，设 $mid = (L+R)/2$ 。然后将包含在 $[L, mid]$, $[mid+1, R]$ 的区间分治处理。剩下的就是包含 $[mid, mid+1]$ [mid, mid+1]的询问，然后找出包含 $[mid, mid+1]$ [mid, mid+1]的所有优美区间，用这些优美区间更新询问的答案。

时间复杂度 $O(n(\log n)^2)$ 。