T1

先去除所有黑色的行和列，做这些操作不会变化。

剩下每种操作都会发生变化，设现在的答案为 $n+m+1$。但是这样会计数重复，如果有一个格子满足同行同列都只有这一个格子，则答案需要减 1。

T2

$S \le 10^6$ 做法：可以看做手上有 $[0,S]$ 这个区间，每次加一个 w_i, v_i 会把 $[0,S]$ 分为 $[0,w_i-1]$ 和 $[w_i,S]$，后者价值加上 v_i 后变成 $[0,S-w_i]$。于是模拟这个过程，我们手上有若干个区间 $[0,r]$，每次会加 w_i, v_i 都会把 $r \ge w_i$ 的那些区间砍一刀。显然砍的次数最多是 S 次，用 priority_queue 维护这些区间，复杂度 $S\log S$。

$S \le 10^{12}$ 做法（正解）：我们发现题目保证了 $w_i \le 10^6$。于是我们发现一个性质，取一个 $T < S-\max w$ 的 T 一定不够优，证明不难。于是我们一开始手上有一个区间 $[S-\max w, S]$，这个区间最多也是被砍 $\max w$ 次，模拟 算法二 的过程即可做到 1log，可以通过。

T3

考虑 $F(x)$ 的性质。

如果 $x=0$ 则 $F(x)=0$；否则，$F(x)$ 只和 $x \bmod (B-1)$ 有关。

如果 $x>0$ 且 $x \bmod (B-1)=0$ 则 $F(x)=B-1$；否则 $F(x)=x \bmod (B-1)$.

证明可以考虑 $x \bmod (B-1)=\sum B^k a_k \bmod (B-1)=\sum a_k \bmod (B-1)$.

考虑对于所有区间，计算出区间的和 $\mathrm{mod}(B-1)$ 以及区间中有哪些数的集合。

对于每次询问，我们想计算哪些区间是**不合法**的，枚举区间的和 y，**不合法** 的条件就是没有一个数在某个集合 T 内，T 可以由 S 计算得出。对于每一个 y，做一个高维前缀和就可以计算。

接下来要处理询问 $x=0, x=B-1$ 的情况，这需要一些分类讨论：

- $x=0$：只有 000000,0000x000 的情况可能符合。
- $x=B-1$：不能变成全 0，需要讨论一些**只能变成全 0** 的情况并减去。
  - 00000：需要能替换一个 $B-1$。
  - 单个字符 x：需要能替换一个 $B-1$。
  - 0000x000：需要有一个 $B-1$ 替换掉 x 或有一个 $B-1-x$ 来替换掉 0。
  - 大于两个数不为 0：这种情况不需要减去。

最终复杂度 $O(nB+B^2 2^B+mB)$，可以通过。

T4

这个问题看起来就非常诡异，我们考虑一些转化。

转化：
可以看做每次 swap 选中连续段的开头结尾。给 B 从左到右标号，只考虑 B 的移动，翻转 ABB 可以看做一个 B 移动了两格并跳过了另一个 B。翻转 AAB 可以看做一个 B 移动了两格。

可以看做初始的每个 B 和最后的每个 B 互相匹配。

特殊性质 A：若两个匹配的 B 之间距离为 d_i，则总代价为 $\sum(\lfloor d_i/2\rfloor\times(C+3)+(d_i \bmod 2)\times(C+2))$。简单 DP 即可。期望得分 48。

特殊性质 B(x)：枚举初始的每个 B 和最后每个 B 的匹配位置，这是一个全排列。此时 AAB 与 ABB 翻转代价不同，我们盲猜最终总代价只会多一个 B 排列的逆序对数。枚举全排列计算，发现这是对的。期望得分 48。结合性质 A 可以获得 68 分。

正解：根据上一行，此时的代价式子是 $\sum(\lfloor d_i/2\rfloor\times(C+4)+(d_i \bmod 2)\times(C+3))+\text{Inversions}$ 。于是也可以使用 DP 求解。

关于这个 DP 的详细说明：

首先把要匹配的 B 按照原来在奇数偶数位分成两组，最后一定是选一些终点**按顺序分配**给原来在奇数位的 B，选一些终点按顺序分配给原来在偶数位的 B。于是设 $f(i,j)$ 为前 i 个终点分配给了偶数位 j 个，枚举下一个给偶数还是奇数位，匹配的点是确定的。简单计算一下贡献即可。

复杂度 n^2，期望得分 100 分。