

Codeforces Round #538(Div.2) solution

A - Got Any Grapes?

简单模拟即可。

B - Yet Another Array Partitioning Task

易得最优策略是去前 $m \times k$ 个最大的，很显然可以取到，然后就模拟即可。

C - Trailing Loves (or L'oeufs?)

对于求 $n!$ 在 k 进制下有几个 0，可以说在 $n!$ 中可以被多少个 x 整除。

所以容易想出对于 k 的质因子 p ，其中有 num_1 个， $n!$ 中有 num_2 个，输出结果为所有 p 中 $\min\{\text{num}_2 - \text{num}_1\}$ ，所以说算法已经显然。质因子分解 k ，然后暴力枚举对于 k 的质因子 p 有几个 num_2 ，短除即可，然后取 \min 。

时间复杂度 $O(\sqrt{k} \log n)$

D - Flood Fill

考验对于选手的阅读能力。

简单的 dp ， $f(i, j)$ 为使得在 i, j 位置中所有颜色是相等的。

然后简单转移即可。

E - Arithmetic Progression

第一次写交互。

两种操作，一种是告诉你有没有元素 $> x$ ，另一种是告诉你打乱等差数列中第 a_i 项是多少。

对于第 1 种操作容易想到二分，即为求在 a 序列中的最大值。

对于第 2 种操作：

观察等差数列 $\{a_1, a_2, \dots, a_n\}$ ，对于任意 $k (2 \leq k \leq n)$ ，则 $a_k - a_{k-1} = d$ ，其中 d 为恒值。而对于打乱序列 a ，所对应的 $|a_k - a_{k-1}| = d_k$ ， d_k 为 d 的倍数，且 $\gcd(d_2, d_3, \dots, d_n) = d$ 。

通过上述可得发现 rand 30 下位置，通过以此去得到 30 个 a_i ，去做 \gcd 就可以得到公差 k 。

因为二分 30 次肯定能得到最大值，而 30 次随机 a_i 对于得到错误的 d 的几率为 1.86185×10^{-9} ，详情见 <https://codeforces.com/blog/entry/65136>。所以有 rp 的就能 AC。

F - Please, another Queries on Array?

其实一眼就可以看出线段树吧。

关于欧拉函数，可以想得与质因子有关。又因为 $a_i, x \leq 300$ ，打个暴力发现质数只有 62 个，所以说可以在线段树中每个节点存储次质数出没出现过。然后再乘除一下即可。

因为细节很多所以代码先坑着，有时间再补。