2021 CCPC 华为云挑战赛
试题分析

比赛链接：https://acm.hdu.edu.cn/contests/contest_show.php?cid=1029

1001. 对象存储调度问题

注意到数据对象的大小都是 2 的整数次幂，贪心地把这些数据对象从大到小去往分条里存就是对的。一个做法是每次取剩下最大的数据对象，然后用堆来维护分条，每次取剩余空间最大的一个分条并将该数据对象存进去，如果存在存不进去的情况则无解，否则就有解。

时间复杂度：$O(n \log n + m \log m)$

1002. 卷业务模型分析

做法很多。一个做法是用最小二乘法分别算出 $\{(A_{1,1}, B_1), (A_{1,2}, B_2), \cdots, (A_{1,m}, B_m)\}$ 和 $\{(A_{2,1}, B_1), (A_{2,2}, B_2), \cdots, (A_{2,m}, B_m)\}$ 这两个点列的回归直线 $y = \widehat{k}x + \widehat{b}$ 以及离差平方和之类的衡量误差的数值，取误差小的作为答案。

时间复杂度：$O(m)$

1003. CDN 流量调度问题

首先 $\lceil \frac{a_i}{k_i} \rceil$ 只有 $O(\sqrt{a_i})$ 种不同的结果，所以每条网络线路只有 $O(\sqrt{a_i})$ 种有用的 CDN 节点个数，因为如果增加个数但不能降低时间的话就不必去加了。

用 $dp_{i,j}$ 表示考虑前 i 条网络线路，使用了 j 个额外的 CDN 节点带来的最大优化量。转移时考虑每条网络线路的增量，至多有 $O(\sqrt{a_i})$ 种有意义的增量，最后每次 $O(m)$ 转移即可。

时间复杂度：$O(nm\sqrt{a_i})$

1004. 银河清浅夜纵横

假设星星都放左侧的方案数为 $f(n)$，则答案是 $2f(n)$。

将夜空逆时针旋转 45°，由于走最短路，所以只能向下或向右。每颗星星需要被检查两次，向下、向右一共 $2n$ 段，所以最多只能确认 n 颗星星，且此时 n 颗星星都不同行不同列。

问题相当于每行每列选一个星星后，在每个星星头顶选一条网格线，使得网格线高度从左到右不增加，如下分析：

- 当 $n=1$ 时，有唯一的格局，$f(1)=1$
- 否则点击最后一行的星星，炸掉一行一列，则得到一个 $(n-1)\times(n-1)$ 的格局
- 对于每个 $(n-1)\times(n-1)$ 的格局，枚举被炸掉的列和这一列选的网格线位置，则从最左边、网格线放最上面开始，每次要么右移一列，要么网格线下降一格，因为如果右边一列的格线比自己低，则不能右移，只能降线；如果右边一列的格线和自己一样，则不能降线，只能右移，且右移后格线恰好在合法的最高位置。因此一共有 $1+n-1+n-1=2n-1$ 种 $n\times n$ 的格局对应这个 $(n-1)\times(n-1)$ 格局
- 故 $f(n)=(2n-1)f(n-1)=(2n-1)!!$，答案为 $2(2n-1)!!$

算 $(2n-1)!! \pmod{2^{64}}$ 是一个经典问题，有一种基于分块的做法，详见文末附录。

1005. 重叠的子串

对字符串 s 建立 SAM。考虑每个询问中的串 t，考虑其在 s 中的所有 endpos。如果其对应 endpos 集合中，最邻近的点对距离小于 $|t|$，则说明该子串在 s 中出现了重叠。

因此，可以在后缀连接的树上预处理出每个点的 endpos 集合的最近距离，然后对于每组询问，倍增找到该子串在树上的对应结点，查询即可。预处理最近点对距离可以用 *set* 启发式合并维护 endpos 集合。

复杂度 $O(n\log^2 n+q\log n)$

1006. 仓颉造数

首先可知，对于任意已有的 2^k 个数，两两求平均数，再把平均数两两求平均数，k 轮后可以得到它们的和除以 2^k。

不妨设 a,b 互质（不互质的话同时除以它们的 gcd 即可）：

- 若 $\frac{a}{b},\frac{b}{a}$ 满足 $a+b=2^k\,(k\in\{1,2,\cdots\})$，则 $\dfrac{a\cdot\frac{b}{a}+b\cdot\frac{a}{b}}{2^k}=1$，是可以合成 1 的
- 否则，考虑对奇质数 p，若 $a+b\equiv 0 \pmod p$，则 $a\equiv -b, ab^{-1}\equiv -1 \pmod p$
  对任意两个 $\equiv -1 \pmod p$ 的数，平均数 $\equiv -1$，调和平均也 $\equiv -1$，所以生成的数始终 $\equiv -1 \pmod p$，不能为 1

综上，能合成 $1 \Leftrightarrow a+b=2^k\,(k\in\{1,2,\cdots\})$，且此时 2^k-1 步一定能合成 1。

1007. 加点

随机基本是过不了的，所以考虑正经一点的做法：

设原来的点为 A 集，新选的点为 B 集，那么三点共线共分为 AAA，AAB，ABB，BBB 四种。首先第一种是不存在的，然后可以弄一个凸包作为 B 的候选点集，那么 BBB 也不存在，弄凸包的方法大概是找到所有的 $|i|, |j|$ 不超过一个阈值（标程是 68，主要是控制坐标绝对值上限不超过 10^5），然后 $\gcd(i,j) = 1$ 的 (i,j) 数对，然后把这些极角排序一下，然后依次增量相连就可以搞出一个坐标比较小，且点也比较多（大概有 11000 个点）的凸包了。

- 考虑 AAB，可以 $O(n^2)$ 枚举每条线，然后在凸包上三分可以求出是否与凸包有交点，有的话就把这个交点打个标记，这部分是 $O(n^2 \log m)$ 的。
- 考虑 ABB，$O(n)$ 枚举每个给定的点，分三种情况：
  1. 这个点在凸包外，那么找到这个点与凸包的两个切点，注意到枚举的给定点与凸包上的点所在直线与凸包的另一个交点是朝着相反方向转的，可以 $O(m)$ 线性扫描所有可能的直线
  2. 这个点在凸包边上，那么这个点只会影响到所在的这条边
  3. 这个点在凸包内，那么就围着凸包转一圈，注意到枚举的给定点与凸包上的点所在直线与凸包的另一个交点也是朝着相同方向转的，也是 $O(m)$ 的

  然后如果发现有两个凸包上的点所在直线经过枚举的点，就随便给其中的一个打个标记。综上，这部分的复杂度是 $O(nm)$ 的。

因为数据随机，所以被标记的点一般不会太多（最多就几百个），所以基本上是可以通过的。亲测甚至写错一些细节也是能过的。。

时间复杂度：$O(n^2 \log m + nm)$

附录. 计算 $(2n-1)!! \pmod{2^{64}}$

设 $(2n-1) = 2^{17}q + r$，$0 \le r < 2^{17}$，考虑：

$$\begin{aligned}
&((2k+1)2^{16} - (2^{16}-1))((2k+1)2^{16} - (2^{16}-3)) \cdots ((2k+1)2^{16} + (2^{16}-1)) \\
=& ((2k+1)^2 2^{32} - (2^{16}-1)^2) \cdots ((2k+1)^2 2^{32} - (2^{16} - (2^{16}-1))^2) \\
\equiv& A + (2k+1)^2 2^{32} B \pmod{2^{64}}
\end{aligned}$$

其中可得 $A = \prod\limits_{i=1}^{2^{15}} (2i-1)^2, B = -\sum\limits_{i=1}^{2^{15}} \frac{A}{(2i-1)^2}$

令 $Q = \prod\limits_{i=1}^{2^{16}q} (2i-1) \equiv \prod\limits_{k=0}^{q-1} (A + (2k+1)^2 2^{32} B) \equiv A^q + A^{q-1} 2^{32} B \sum\limits_{k=0}^{q-1} (2k+1)^2 \pmod{2^{64}}$

又 $\sum\limits_{k=0}^{q-1} (2k+1)^2 = 4\sum\limits_{k=0}^{q-1} k^2 + 4\sum\limits_{k=0}^{q-1} k + q = \frac{q(2q-1)(2q+1)}{3}$

故 $Q \equiv A^{q-1}(A + 2^{32} B \frac{q(2q-1)(2q+1)}{3}) \pmod{2^{64}}$

剩下 $\prod\limits_{i=2^{16}q+1}^{n} (2i-1)$ 暴力计算即可，该结果再乘上 Q 即为最后的答案。