

1. 若干组物品或数字取值，并且取值的和有上界的问题，可以考虑背包或生成函数，消序可以用指数生成函数做。
2. 当答案下界至少有 $1/2$ 或以上个，可以考虑随机思想。
3. 如果存在某种贪心决策，一次只选择最大或最小，可以用区间dp出答案。
4. 涉及到几种方案里只选择一个，或者有两个值选择后不如只选其中一个，可以考虑分组背包，dp题想不太出就想想看背包。
5. 如果 $n\log n$ 复杂度能过，但是想不到怎么来的，可以想想看调和级数复杂度。
6. $O(1)$ 做区间修改可以通过维护一阶差分数组，算答案的时候做一阶差分的前缀和。
7. 字符串的题如果能够确定答案长度很小，写if else不如直接暴力搜答案。
8. dp数组太大时，可以考虑是否可以滚动。
9. 树上异或问题，考虑按位分离讨论。
10. query数据范围有限，比如几千个的时候，可以暴力打表存答案，不会T。
11. 如果求解区间过大，但是元素之间有前后递推关系，则可以分块进行打表，然后剩下部分暴力递推。（分块打表）
12. 数学题或树上问题可以考虑算贡献。
13. 对一个数字进行某种操作，如果有固定步长的概念，那么一个步长d内，相邻位数之间没有关系， i 只和所有 $i+j*d$ 有关。
14. 步长为2考虑分奇偶讨论。
15. 超大数质因数分解用pollard_rho，同时fac要离散化
16. 连续数字的异或值，每4位00, 01, 10, 11异或值是0
17. 非此即彼，可以考虑做减法来数数。
18. 需要开60个数组存01两种状态，可以使用状压。
19. 卡特兰数数列的前几项为：1, 1, 2, 5, 14, 42, 132, 429, 1430, 4862, ...
20. 看到整除关系，可以考虑调和级数复杂度算法。
21. 二分图需要考虑的问题：1.是否有多个连通分量，2.是否是二分图
22. 方案总数有限可以暴力。
23. dp数组的设计，其实就是把你需要的所有历史信息全部交代，就可以了
24. 能够被 $1-n$ 整除，就是能够被 $1-n$ 的整体lcm整除
25. 在一些对区间有所限制的问题当中，我们可以通过维护合法区间的左右边界。通过区间移动找出所有合法的区间，最后找到最终的答案。（尺取法/双指针/滑动窗口）