

Link

<https://atcoder.jp/contests/agc032/tasks>

Summary

Task	Score	Difficulty	Tag
A	400	易	模拟
B	700	易	构造
C	800	中	构造、数据结构
D	1000	中	DP
E	1200	难	思维、推导、单调性、二分

Details

A - Limited Insertion

题意:一开始有一个空的数组 a , 然后对这个空数据进行了 N 次操作, 第 i 次操作选择一个位置 $j(j \leq i)$ 插入值为 j 的数。给你一个长度为 N 的数组 b , 问你是否存在一组操作可以得到数组 b , 如果存在输出任一组操作, 不存在输出 -1 。

解法:由于每次都是在第 j 个位置插入 j , 可以倒着考虑, 最后一次必定存在一个数 x 它的下标也是 x , 若不存在则是无解。把这个数抽掉后是一个规模缩小1的子问题, 可以迭代求解下去。如果有多个位置符合要求, 不失一般性考虑 $i \leq j$, 抽走 i 会使得 j 往前一位而使其变为不符合要求, 若抽走 j 由于 i 在 j 的前面, 不会受到影响, 所以显然应该抽走最大的那个。

B - Balanced Neighbors

题意: 给你一个数 $N(3 \leq N \leq 100)$, 让你构造一张简单连通图, 使得每个点的邻居编号之和都相同。

解法: 如果是一张完全图, 那么每个点的 $sum_i = \sum_{j=1, j \neq i}^n = n * (n + 1) / 2 - i$

注意到 $1 + n == 2 + (n - 1) == 3 + (n - 2) == i + (n - i + 1) == n + 1$

因此如果对于第 i 个点去掉与 $n - i + 1$ 的边则每个点的

$sum_i = n * (n + 1) / 2 - i - (n - i + 1) = n * (n + 1) / 2 - (n + 1)$, 为定值。

注意根据 N 的奇偶性来构造就可以了

C - Three Circuits

题意: 给你一张有 $N(1 \leq N \leq 10^5)$ 个点 $M(1 \leq M \leq 10^5)$ 条边的简单无向连通图, 问是否可以将这张图划分为三个环(可以重复经过点但不能重复经过边), 只输出是否即可。

解法: 题目中的环可以重复经过点, 其实就是欧拉回路。如果可以划分为三个环, 那么可以将三个环拼接成一个大环, 因此首先所有点的度数都要为偶数, 即存在欧拉回路。先考虑如何判断一张存在欧拉回路的图能否划分为两个环, 可以求出任一欧拉路径(即回路去掉最后回到开头的那一步), 如果欧拉路径中有一个点出现了两次, 假设下标为 i 和 j , 则说明 i 到 j 这个子区间是一个环。

判断能否划分为三个环可以考虑先删去一个环后判断剩下的图能否划分为两个环，先求出一个欧拉路径序列，假设长度为 len ，倍长后任一段长度为 $len + 1$ 的都是原图的一个合法欧拉回路序列，我们可以在倍长后的序列上枚举删去的环，具体地，对于每个 $i(1 \leq i \leq len)$ 找到离 $a[i]$ 最近的下一个值为 $a[i]$ 的位置 pos ，将 $[i, pos]$ 这段形成的环删去，接下来只需查询 $[pos + 1, i + len]$ 这段区间内是否有出现至少两次的数，一种做法是看 $\min_{j=pos+1}^{i+len} nextpos[a[j]] \leq i + len$ 是否成立，即区间中是否有一个数的下一个出现位置依旧在区间内，用支持查区间最小值的数据结构维护即可。

D - Rotation Sort

题意：给你一个长度为 $N(1 \leq N \leq 5000)$ 的排列，每次操作可以选择一段区间 $[L, R]$ ，花费 A 将这段区间里的数循环左移，或者花费 B 将这段区间里的数循环右移，可以进行无限次操作，问将序列变为升序（1到 N ）的最小花费。

解法：首先可以直接看成每个数确定往左或往右移动，或者留在原位，即假设一个数的原位置是 pos ，最终的位置是 pos' ，若 $pos < pos'$ 是往右移动了，其余两种情况类似，并且一个位置里可以放多个值（可以看作一些盒子，每个盒子放一些数，盒子内的顺序可以随意定，从一个盒子移到另一个盒子需要花费）。定义 $dp[i][j]$ 表示已经考虑了前 i 小权值的位置，且 i 的位置是 j 的最小花费，显然有 $j \leq i$ ，分别从小到大枚举 i 枚举 j 。

$$dp[i][j] = \begin{cases} \min_{k=0}^i dp[i-1][j] + B & j < pos[i] // \text{表示左移} \\ \min_{k=0}^i dp[i-1][j] + A & j > pos[i] // \text{表示右移} \\ \min_{k=0}^i dp[i-1][j] & j == pos[i] // \text{表示不动} \end{cases}$$

记录前缀最小值后根据 j 与 $pos[i]$ 的大小关系进行转移即可。

最终 $ans = \min_{i=0}^n dp[n][i]$ ，注意转移要从0开始。

E - Modulo Pairing

题意：给一个正整数 $M(1 \leq M \leq 10^9)$ ，给你 $2N(1 \leq N \leq 10^5)$ 个数 $a_1, a_2, \dots, a_{2n}(0 \leq a_i < M)$ ，将 $2N$ 个数分成 N 对，假设一对数为 X 和 Y ，则这一对的 $val = (X + Y) \bmod M$ ，问所有对的 val 值的最大值最小是多少。

解法：可以证明最优情况一定是将数从小到大排序后，存在一个分界点，左边部分的点首尾依次匹配，并且每对的和都不超过 M ，右边部分的点首尾依次匹配，并且每对的和都大于等于 M 。

具体地，可以分别证明如下7种情况来证明上述结论：

- $a_i \leq a_x \leq a_j \leq a_y, a_i + a_j < M, a_x + a_y < M, (a_i + a_j, a_x + a_y)$ 不优于 $(a_i + a_y, a_x + a_j)$
- $a_i \leq a_x \leq a_j \leq a_y, a_i + a_j \geq M, a_x + a_y \geq M, (a_i + a_j, a_x + a_y)$ 不优于 $(a_i + a_y, a_x + a_j)$
- $a_i \leq a_j \leq a_x \leq a_y, a_x + a_y < M, (a_i + a_j, a_x + a_y)$ 不优于 $(a_i + a_y, a_j + a_x)$
- $a_i \leq a_j \leq a_x \leq a_y, a_i + a_j \geq M, (a_i + a_j, a_x + a_y)$ 不优于 $(a_i + a_y, a_j + a_x)$
- $a_i \leq a_x \leq a_j \leq a_y, a_x + a_y \geq M, a_i + a_j < M, (a_i + a_j, a_x + a_y)$ 不优于 $(a_i + a_x, a_j + a_y)$
- $a_i \leq a_x \leq a_y \leq a_j, a_x + a_y \geq M, a_i + a_j < M, (a_i + a_j, a_x + a_y)$ 不优于 $(a_i + a_x, a_y + a_j)$
- $a_i \leq a_x \leq a_y \leq a_j, a_x + a_y < M, a_i + a_j \geq M, (a_i + a_j, a_x + a_y)$ 不优于 $(a_i + a_x, a_y + a_j)$

然后可以发现，当分界点越往左时，可能的答案越小，因此我们可以二分这个分解点，合法的最左分解点所对应的就是最优解。并且注意二分如果是左部分不合法则应该往左二分，如果是右部分不合法应该往右二分。