《心跳排列图》命题报告

重庆市巴蜀中学 罗思远

November 9, 2022

Contents

1 题目大意

1.1 题目描述

在本题中，一切序列的下标从 1 开始。

你有一个算法竞赛机器人，每分钟心跳 n 次，第 i 次心跳的强度为 a_i。这里，$a_1 \sim a_n$ 恰为 $1 \sim n$ 的一个排列。

设 A_i 为序列 a 删除第 i 个元素后得到的序列，即 $A_i = [a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_n]$。

对于元素互不相同的序列 p，设 $G(p)$ 为一个无向图，有 $|p|$ 个点，编号为 $1 \sim |p|$。对于每对正整数 $1 \le i < j \le |p|$，若 $\forall k \in [i, j] \cap \mathbb{Z}$，都有 $p_k \in [\min(p_i, p_j), \max(p_i, p_j)]$，则 $G(p)$ 中 i 号点和 j 号点有一条边。设 $F(p)$ 为 $G(p)$ 中 1 号点到 $|p|$ 号点的最短路长度，这里一条路径长度定义为其边数。

设 $f(a) = [F(A_1), F(A_2), \ldots, F(A_n)]$。

给定长度为 n 的序列 $[b_1, \ldots, b_n]$，请你求出任意一个 $1 \sim n$ 的排列 a，使得 $f(a) = b$。保证有解。

在某些子任务中，算法竞赛机器人小 G 会给你一些“提示”：设 $G_0 = G(a)$，设 $path_0$ 为 G_0 中某条 1 到 n 的最短路经过的点构成的集合，设 $path_j$ 为 $G(A_j)$ 中某条起始点到结束点的最短路经过的点构成的集合（为了方便，这里给出的 $path_j$ 中点的编号仍然沿用原图中点的编号）。则小 G 有可能会额外告诉你所有 $path_j$（包括 $path_0$），也有可能只告诉你 $path_0$，也有可能不给你提示。

1.2 数据范围

多组数据，$n \le 10^5$，所有数据中 n 的和不超过 5×10^5。

1.3 时空限制

时间限制 1s，空间限制 2GB。

1.4 部分分

- 子任务 1（7 分）$T \le 250, n \le 7$。
- 子任务 2（5 分）$b_i = 1$。
- 子任务 3（10 分）$n \ge 90000$，保证存在一组解满足 $a_1 = 1, a_n = n$。
- 子任务 4（7 分）$n \ge 90000$，保证存在一组解满足 $a_2 = 1, a_{n-1} = n$。
- 子任务 5（15 分）$n \le 100, \sum n^3 \le 3 \times 10^6$，存在所有 $path_j$ 的提示。
- 子任务 6（15 分）$n \le 100, \sum n^3 \le 3 \times 10^6$，存在 $path_0$ 的提示。
- 子任务 7（15 分）$n = 100, T = 3$，共 5 个测试点，输入生成方式是随机一个 a 再求出 $f(a)$ 作为输入。
- 子任务 8（25 分）$n \le 100, \sum n^3 \le 3 \times 10^6$。依赖子任务 $1, 5, 6, 7$。
- 子任务 9（1 分）无特殊限制。依赖子任务 $1 \sim 8$。

2 解法分析

2.1 初步观察

设 p 是我们要求的排列，a 是输入的序列。定义 $G(p)$ 中，1 到 n 的最短路经过的点集为 $path$，将 $path$ 中的点称为**关键点**。

首先，我们观察 $path$ 的结构特点。容易发现，可以用以下方法计算 $F(p)$：定义 $dis(l,r)$ 表示 $G(p)$ 中 l 号点与 r 号点间的最短路，最开始调用 $dis(1,n)$；在调用 $dis(l,r)$ $(l \neq r)$ 时，先找出 p 中 $[l,r]$ 区间内的最小值与最大值位置，分别设为 m 与 M；不妨假设 $m < M$。若 $m = l$ 且 $M = r$ 则返回 1；否则，返回 $dis(l,m) + dis(M,r) + 1$。

同时，我们还有另一种求 $F(p)$ 的方法：从 1 开始往 n 移动，若当前在 x 号点，则找到最大的 y 使得 $G(p)$ 中 x,y 之间有边，移动到 y 号点。不停移动直到 $x = n$。

通过上述两种算法，我们不难确认以下事实，这些事实很容易根据算法过程得到验证，故在此略去证明。

引理 2.1.1. *删去非关键点得到新的序列 p'，有 $F(p) = F(p')$。*

引理 2.1.2. *删去关键点后得到新的序列 p'，有 $F(p') \geq F(p) - 2$。*

引理 2.1.3. *若将 (i, p_i) 这 n 个点画在平面直角坐标系上，并连接在 $path$ 中相邻的点，得到的图形如下：*

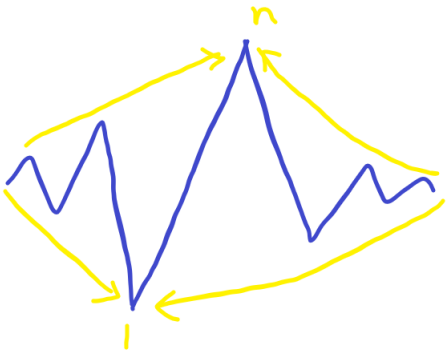

Figure 1: $path$ 的形状

当然，图形也可能与我们画出的相反，1 处于 n 右边，但显然总可以假设 1 处于 n 左边，因为将整张图上下反转，不改变 $F(A_i)$。我们称连接 $1, n$ 的斜线为**大斜杠**。我们称 $path$ 中，比与其相邻的位置均**高**（即纵坐标更大）的位置为**峰**，否则称为**谷**。称纵坐标为 1 的位置**谷底**，称纵坐标为 n 的位置**峰顶**。

以下，我们提到**位置** i 时，如果同时提到了横坐标或纵坐标，实际所指是平面直角坐标系上 (i, p_i) 这个点。

引理 2.1.4. *$path$ 中处于大斜杠**左边**（即横坐标更小的一边）的位置中，峰的纵坐标从左到右递增，谷的纵坐标从左到右递减，在大斜杠右边恰好相反。*

有了上述初步观察，我们便可开始解决问题了。

2.2 准备工作

以上，我们针对 $path$ 本身的结构做出了一些观察，并且也对 $F(a)$ 中数的下界作出了估计。自然的想法是对 $G(A_i)$ 中的新最短路，设为 $path'_i$，的结构也进行类似的观察，并找出删去一个点后 $path$ 变化情况的基本规律。

称原本不在 $path$ 中，但删去某个位置 i 后存在于 $path'_i$ 中的元素 x 为**半关键点**，同时称 i **管辖** x。称 $path$ 中 i 位置的前驱位置为 $pre(i)$，i 位置的后继位置为 $next(i)$。称 i 管辖的点集为 H_i。

引理 2.2.1. *若 $i \in path, a_i \geq F(p) - 1$，则删去位置 i 后，**总可以认为** $pre(i)$ 和 $next(i)$（若存在）都仍处于 $path'_i$ 中。*

Proof. 不妨假设 i 在谷底左侧或就是谷底。证明考虑下图：

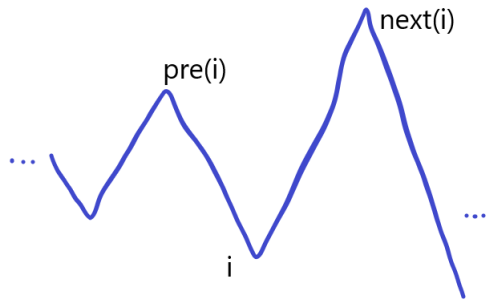

Figure 2: $pre(i), i$ 与 $next(i)$

由上述求 $path$ 的算法流程可知，删去 i 后，$next(i)$ 一定仍属于最短路。若 $pre(i)$ 不属于最短路，通过上面第二种求 $path$ 的算法，我们知道 $pre(pre(i))$ 一定仍属于最短路；考虑从其开始往右移动的过程，若直接移动到 $next(i)$，则新的最短路长度为 $F(p)-2$，与前提矛盾；若不直接移动到 $next(i)$ 也不移动到 $pre(i)$，只会是下述情况：

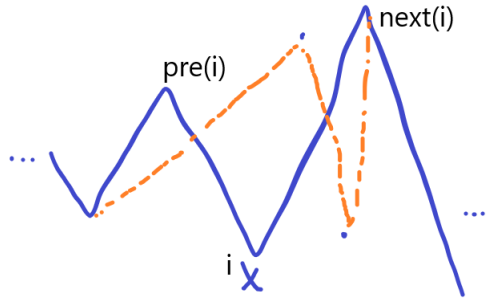

Figure 3: 删去 i 后的变化

注意，上图中在删去 i 后只画出了一个新峰和一个新谷，但其实可能有多个新峰和新谷。在只有一个新峰一个新谷时，我们作下述调整（按绿色箭头方向改变新峰谷的纵坐标）：

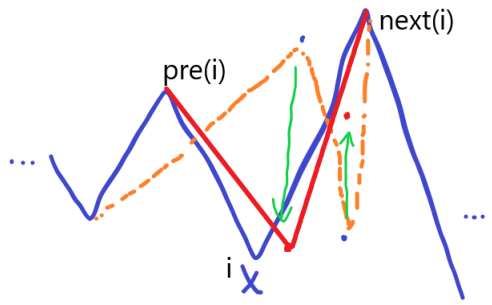

Figure 4: 删去 i 后的变化

显然，该调整不改变 a 数组，同时使 $pre(i)$ 存在于了新最短路上。

在有多个新峰和新谷时调整类似，这里略去。
若 $pre(pre(i))$ 不存在即 $pre(i)=1$，上述结论显然仍正确。 □

推论 1. *若 $i\in path, a_i\geq F(p)-1$，则删去位置 i 后，**总可以认为**若 $pre(i)$ 和 $next(i)$ 存在且不为 1 或 n，则它们若原来是峰，现在仍为峰；若原来是谷，现在仍为谷。*

Proof. 通过上述第二种计算 $path$ 的方式容易证明该推论。 □

引理 2.2.2. *设位置 i 处于谷底左侧且属于 $path$ 且 $pre(i)$ 存在，则 i 管辖的元素横坐标一定在 $(pre(i),next(i))$，且至多有一个元素的横坐标在 $(pre(i),i)$。*

Proof. 引理的前半部分证明略，这里证明后半部分：假设至少有两个元素的横坐标在 $(pre(i),i)$，因为 $path$ 上一定是峰谷交替，所以其中至少有一个峰和一个谷。不妨假设 i 是谷，由引理 2.2.1，$pre(i)$ 仍存在于最短路上且 $pre(i)$ 是峰，考虑属于 $(pre(i),i)$ 中的新峰位置 x，由引理 2.2.1 知道删去 i 后 $pre(i)$ 仍存在于最短路中；由引理 2.1.3 推出 $p_x>p_{pre(i)}$。然而在原 $path$ 中 i 和 $pre(i)$ 相邻，这说明在 $G(p)$ 中 i 和 $pre(i)$ 之间有边，因此 $p_x\in[p_i,p_{pre(i)}]$，矛盾！ □

引理 2.2.3. *设 $i\in path, i\neq 1, i\neq 2, i\neq n-1, i\neq n$，则 $|a_i-F(p)|$ 是偶数。*

Proof. 通过推论 1 我们知道，如果 $pre(i)$ 和 $next(i)$ 均不为 $1,n$，则它们都存在于 $path_i'$ 中，根据峰谷位置在 $path$ 中位置的奇偶性容易导出上述结论。

如果 $pre(i)$ 和 $next(i)$ 中某一者或两者均为 1 或 n：若 $pre(i)=1$，不妨假设 1 为峰，i 为谷，因为 $i\neq 2$，所以 $1<2<i$，所以 $p_2\in[p_i,p_1]$，因此删去 i 后，由于 p_2 的存在，1 仍然为峰。同理，若 $next(i)$ 为 n，删去 i 后也不会改变其峰谷性。所以上述结论仍成立。 □

引理 2.2.4. *设 $i\in path$，若 $pre(i)=1$ 或 $next(i)=n$，则 $a_i\neq F(p)-2$。*

Proof. 根据计算 $path$ 的第一种方式容易证明该结论。 □

引理 2.2.5. *设 $i\in path$，i 处于谷底左侧或就是谷底，$pre(i)>1$，$a_{pre_i}>F(p)$，则 $a_i\geq F(p)$。在 i 处于谷右侧时，将 $pre(i)$ 改为 $next(i)$，$pre(i)>1$ 改为 $next(i)<n$，结论仍然成立。*

Proof. 我们只证明 i 处于谷左侧或就是谷底的情况。不妨设 i 是谷。

因为 $a_{pre_i}>F(p)$，所以 $|H_{pre(i)}|\geq 2$，所以 H_{pre_i} 中至少有两个点，即其中至少有一个峰和一个谷，设其中的谷为 x。根据 $path'_{pre(i)}$ 的结构，我们知道 x 纵坐标比 $pre(pre(i))$ 低，这要求 $x>pre(i)$。假设 $a_i<F(p)$，根据引理 2.2.3 和引理 2.1.2 我们知道 $a_i=F(p)-2$，这要求 $pre(pre(i))$ 在删去 i 后能直接到达 $next(i)$，这又要求 x 纵坐标比 $pre(pre(i))$ 高，矛盾。 □

推论 2. *设 $i\in path$，i 处于谷底左侧或就是谷底，$pre(i)>1$，$a_{pre_i}>F(p)$，则 $|H_i|\geq 1$。*

引理 2.2.6. *设 $i\in path$，i 处于谷底左侧或就是谷底，$pre(i)>1$，$a_{pre_i}>F(p)$，则**总可以认为** $|H_i\cap H_{pre(i)}|=1$。在 i 处于谷右侧时类似的结论也成立。我们称这种情况为“共用”。*

Proof. 我们只证明 i 处于谷左侧或就是谷底的情况。不妨设 i 是谷。

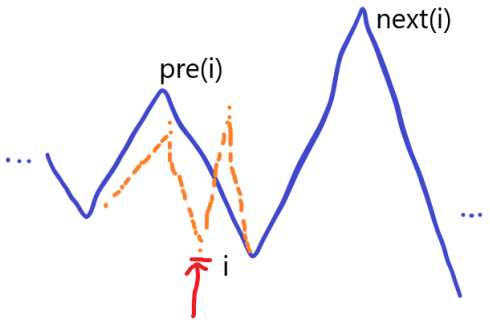

Figure 5: 删去 $pre(i)$ 后的变化

考虑 $pre(i)$ 管辖的最靠右的谷 x，我们证明总可以认为 $x \in H_i$（显然，只要 $H_i \cap H_{pre(i)}$ 非空，其中的元素必定是 x）。

若 $|H_i \cap H_{pre(i)}| \neq 1$，考虑 i 管辖的最靠左的谷 y，根据引理 2.2.2 可以推出 $|H_i \cap H_{pre(i)}| \leq 1$，因此 $|H_i \cap H_{pre(i)}| = 0$，所以 $y > x$。这时，将 i 管辖的所有元素 $[y, x_1, x_2, \ldots]$ 在这个序列中整体左移一位，峰谷情况不变，并且将 y 移到 x 位置上；对于 i 管辖的最后一个元素，它是谷，左移后其原来所在的位置作出下图所示的调整即可（最右侧绿色箭头）。图中黄色线条是 $H(pre(i))$ 间的连边，黄色箭头所指是 x，粉色线条是 $H(i)$ 原来的连边，绿色线条是调整后的连边。容易发现调整不改变数组 a。

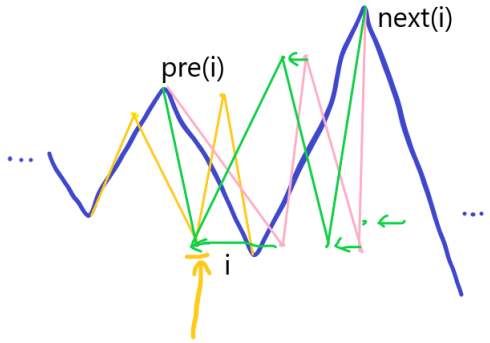

Figure 6: 调整

□

实际上，到这里我们已经具备了做出本题的所有必要观察了。接下来考虑几个特殊情况。首先是边界上的非法情况：

情况 2.2.7. *以下情况不可能存在：$F(p) = 2$，$path = \{1, 2, n\}$，$p_1 = 1$，$p_2 = n$，$a_2 = 1$；或 $F(p) = 2$，$path = \{1, n-1, n\}$，$p_n - 1 = 1$，$p_n = n$，$a_{n-1} = 1$。*

Proof. 只证 $path = \{1, 2, n\}$ 的情况。考虑 p_3，因为 $p_2 = n$ 必定是峰，所以 p_n 是谷，$p_n < p_3 < p_2$；因为 $a_2 = 1$，所以删去 2 后新的最短路直接变为 $1 \to n$，因为 $p_1 = 1$ 必定是谷，所以 $p_1 < p_3 < p_n$，矛盾！ □

情况 2.2.8. *记 $path$ 中前 3 小的位置是 $1, p, q$，以下情况不可能存在：$q > p + 1, a_q = F(p) - 2, a_p = F(p) - 1$。*

Proof. 考虑 (q, p) 中的元素，通过类似上面引理的推导可以导出矛盾。 □

情况 2.2.9. *若 $a_1 \geq F(p)$，则类似引理 2.2.6，我们总可以认为 $|H_1 \cap H_{next(1)}| = 1$。对于 n 也有类似的情况。*

Proof. 证明与引理 2.2.6 类似。可以发现，这种情况其实表明引理 2.2.6 在边界上会具有更松的条件。 □

其次是大斜杠内部的情况：

情况 2.2.10. *设 P 和 Q 分别是谷底和峰顶。若 $|H_P| = 1$ 且 $|H_Q| = 1$，可能出现下面的情况：*

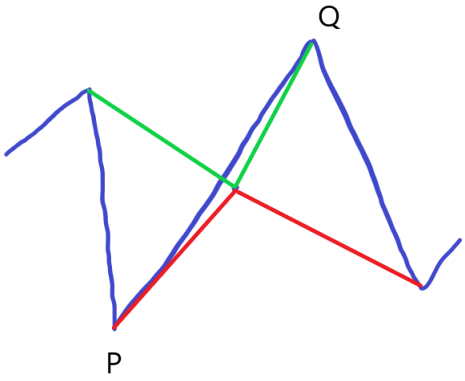

Figure 7: 一种特殊情况

这里绿线表示 H_Q，红线表示 H_P。我们称这种情况为“P,Q 共用一个位置”。

情况 2.2.11. 设 P 和 Q 分别是谷底和峰顶。若 $|H_P| \geq 2$ 且 $|H_Q| \geq 2$，可能出现下面的情况：

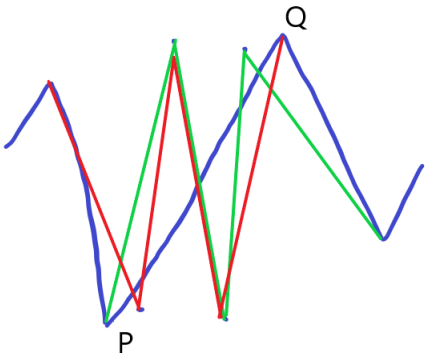

Figure 8: 一种特殊情况

这里绿线表示 H_Q，红线表示 H_P。我们称这种情况为“P,Q 共用两个位置”。

在上述”共用两个位置“的情况下，如果共用的两个位置中靠左的一个纵坐标小于 P 之前的峰，则可能出现另一种“P,Q 共用一个位置”的情况。

同时容易发现，大斜杠两端至多共用两个位置。

2.3 第一步：DP 确定关键点

通过上面的讨论，我们着手设计算法。

引理 2.3.1. 设 $d_0 = \min_{1\leq i\leq n} a_i$。则 $F(p)$ 只会是 d_0, d_0+1, d_0+2 中的一个。

Proof. 若至少存在一个位置不属于 $path$，显然成立。若所有位置都属于 $path$，这时 $F(p) = n-1$，上面又已经证明 $a_i \geq F(p) - 2 = n-3$，所以仍然成立。 □

由此，我们可以先枚举 $F(p)$，即不删任何元素时的最短路长度：它只有 $O(1)$ 种情况。不妨令 $D = F(p)$。

根据引理 2.1.1，所有 $a_i \neq D$ 的 i 都一定是关键点，$1, n$ 当然也一定是关键点。然而，还有一些我们不能确定位置的关键点，它们删去后的最短路长度仍是 D，因此混迹于非关键点之中无法识别。将暂未确定的关键点称作**隐式关键点**，已经确定的称为**显式关键点**。

不过，$a_i = D$ 的关键点 i 一定恰好管辖一个点，同时，容易证明总可以认为其管辖点要么与它自己相邻，要么与另一个 $a_i > D$ 的关键位置（或是 $a_i = D$ 的边界位置）共用。因此，关于隐式关键位置对整个问题的影响，我们很大程度上只关心它们的数量，而不关心它们的具体位置。具体地，对隐式关键点的限制可以描述如下：

- 如果只考虑显式关键点，关键点的个数 u 小于 $D+1$，则需要至少 $D+1-u$ 个隐式关键点来“凑数”。
- 如果只考虑显式关键点，出现了引理 2.2.4、引理 2.2.5、情况 2.2.8 之中断言不可能存在的情况，我们就需要在违反上面引理的位置中间插入至少一个隐式关键点来修正。

如果在第二条限制里，需要的关键点个数比 $D+1-u$ 还多，则直接返回当前的 D 无解。否则，可以发现，上述限制都在限制隐式关键点的数量下界，因此隐式关键点其实“越多越好”：如果有一种 K 个隐式关键点的情况，任意删去一个不会违反第二条限制的关键点就得到了 $K-1$ 个关键点的情况。

若我们有一段连续的长度为 len 的区间，其中没有显式关键点且元素没有被任何 i 管辖且也不存在引理 2.2.6 中的共用，根据上面的讨论，其中至多可以放下 $\lfloor \frac{len}{2} \rfloor$ 个隐式关键点。

如果两端存在共用元素，对式子的改变等价于让 len 加上了 1（因为这意味着与可共用元素相邻的隐式关键点只消耗一个位置而非两个位置了）。除了“共用”会影响 len，在引理 2.2.2 中提到了“至多有一个元素横坐标在左边”，而到底是恰好有一个元素在左边还是没有元素在左边依赖于我们自身的决策。为了处理这样的决策，我们考虑用动态规划。

设 $dpl(i,0/1)$ 表示从左到右 dp 到第 i 个显式关键点，其中第 i 个显式关键点如果有管辖元素的话，它的管辖元素中可以自由移动的（称这个元素为**自由元素**）放在左边 / 右边，此时在第 i 个显式关键点之前至多有多少个隐式关键点。$dpr(i,0/1)$ 类似，但是从右往左做，因为在大斜杠两侧的处理方式是有差别的。

考虑枚举大斜杠的位置 P,Q ($p_P=1, p_Q=n$)，并讨论情况 2.2.10 和情况 2.2.11 来合并 $dpl(Q,0/1)$ 和 $dpr(Q,0/1)$（这里 P,Q 不一定是显式关键点，因此可能需要额外执行一次从 P 上一个显式关键点到 P 的转移，Q 同理）。如果合并成功（满足上面的两条限制），就可以直接进入第二步了。

枚举 P,Q 时，我们需要保证 (P,Q) 区间内不存在显式关键点。如果 P,Q 之一是显式关键点，这样的 (P,Q) 只有 $O(n)$ 对，否则，根据情况 2.2.10，容易推出 $Q-P\le 2$，其它情况都是无用的。所以这时 (P,Q) 也只有 $O(n)$ 对。因此这步的总时间复杂度是 $O(n)$。

如果得到了合法 P,Q，记录下 dp 转移并倒推，即可找到每个自由元素放在左边还是右边，进而得到所有可以放置隐式关键点的段。得到这些段之后，优先在违反上面第二条限制的段里面插入隐式关键点，确保没有违反限制的段后，剩下的隐式关键点可以随意放置。

2.4 第二步：拓扑排序求出最终方案

这部分相比于第一步要简单、机械很多：通过上述 DP，已经还原出哪些点应当处于 $path$ 里，也很容易确定 i 管辖的所有点，接着根据引理 2.1.3 和引理 2.1.4 确定关键点之间，半关键点之间，以及关键点和半关键点相互间必要的大小关系，连边成为一个有向图，在有向图上运行拓扑排序算法即可求出一种合法解。

定理 2.4.1. *建立的有向图一定存在拓扑序。*

Proof. 前述的引理中的调整，保证了只要有解，就一定存在满足我们得到的拓扑序要求的解，所以建立的有向图一定存在拓扑序。 □

由于前述 DP 是单次 $O(n)$ 的，最初要枚举 $O(1)$ 种 $F(p)$ 的取值，拓扑排序算法也是 $O(n)$ 的，故本题在 $O(n)$ 时间复杂度内得到解决。

3 针对部分分的算法

3.1 子任务 1

使用 $O(n! \times poly(n))$ 的算法枚举全排列并求出 $F(A_i)$，可以通过子任务 1。

3.2 子任务 2

直接输出 $1, 2, \ldots, n$ 可以通过子任务 2。

3.3 子任务 3,4

通过并不麻烦的讨论，可以对子任务 $3, 4$ 的情形直接作出构造。

3.4 子任务 5

只要选手作出了“初步观察”一节内对最短路的纵坐标满足的大小关系的观察，再根据此大小关系拓扑排序，就可以通过子任务 5。

3.5 子任务 6

如果选手没有发现可以动态规划得到本来属于 $path$ 的位置，但已经发现确定位置后贪心放置半关键点的方式，并想到了拓扑排序求出最终答案，仍可通过子任务 6。

3.6 子任务 7

在 $i = 1, 2, n-1, n$ 以及大斜杠处存在分类讨论，如果选手漏掉了一些情况，仍可通过数据完全随机的子任务 7。

3.7 子任务 8

如果选手暴力枚举所有可能的大斜杠位置，并每次重新使用 DP 与拓扑排序判断是否合法，会使复杂度达到 $O(n^2)$ 或 $O(n^3)$，但只要正确性无误仍可通过子任务 $5, 6, 7, 8$。

4 命题灵感

本题的思想来源于《心跳》一题（题目链接：https://www.luogu.com.cn/problem/P8554）。在此题中，我们需要对一个序列删去一个位置后得到序列的前缀最大值个数序列计数。我在思考这道题以及尝试拓展的过程中，发现其“可做”的关键有两处：一是“前缀最大值”这种对象本身便于描述结构；二是删去一个位置后，对整个序列的影响是局部的，几乎不会互相影响。我自然地想到，是否存在一种新的组合对象，其本身结构比较容易描述，删去一个位置后的影响也“几乎”是局部的、不难处理的，但是具有更灵活、更多样的有趣性质。经过思考，我发现 Codeforces 上的《Permutation Graph》一题（题目链接：https://codeforces.com/problemset/problem/1696/D）给出了一种有趣的构造，因此将两题结合命制了本题。实际上，本题的名字《心跳排列图》正是两者名字的拼接。

解决本题不需要任何超过 NOIP 普及组难度的知识点，问题最终解决只用到了动态规划和拓扑排序两种算法。尽管若要通过本题需要较为繁琐的分类讨论和代码实现，笔者仍然设置了大量基本不需要分类讨论即可获得的部分分，其意即希望只要分析出需要求的对象的结构特点的选手，都能获得不少的分数。总体而言，本题能区分出写了暴力的选手、会分析性质的选手，以及分析出了性质，且能认真讨论出所有情况的选手。

5 参考文献与致谢

- 《心跳》题目描述，https://www.luogu.com.cn/problem/P8554
- 《心跳》解题报告，李羿辰，https://www.luogu.com.cn/blog/dottle/xin-tiao
- 《Permutation Graph》题目描述，https://codeforces.com/problemset/problem/1696/D
- 《Permutation Graph》解题报告，罗思远，https://codeforces.com/blog/entry/103479

集训队员郭雨豪和常瑞年参与了本题的验题与本文审稿工作，在此感谢。