NOIp 2026 模拟赛第一场
参考题解

小鸥训练中心

目录

1 双网通行（traffic） 2
- 1.1 关键观察 . . . 2
- 1.2 正确性证明 . . . 2
- 1.3 复杂度 . . . 2
- 1.4 部分分 . . . 2

2 栏目排列（sheet） 2
- 2.1 转化为最大权 Hamilton 路 . . . 2
- 2.2 状态压缩动态规划 . . . 3
- 2.3 正确性证明 . . . 3
- 2.4 复杂度与实现 . . . 3
- 2.5 部分分 . . . 3

3 序列消除（erase） 3
- 3.1 基本递推 . . . 3
- 3.2 干净边界区间 . . . 4
- 3.3 回答询问 . . . 4
- 3.4 正确性证明 . . . 4
- 3.5 复杂度 . . . 5
- 3.6 部分分 . . . 5

4 停机检查（halt） 5
- 4.1 默认边与特殊事件 . . . 5
- 4.2 函数图树上的批量询问 . . . 5
- 4.3 环上的命中 . . . 6
- 4.4 事件图 . . . 6
- 4.5 正确性证明 . . . 6
- 4.6 复杂度与部分分 . . . 6

1 双网通行（traffic）

1.1 关键观察

同一交通系统内任意两个不同站点之间都可以用一次系统通行证到达，因此同系统公开线路不会进一步缩短距离。只需保留端点属于不同系统的公开线路。

记交通系统 t 的站点数为 s_t，站点 u 的跨系统公开线路度数为 deg_u，系统 t 中跨系统度数非零的站点数为 b_t。设 u 属于系统 t，另一个系统为 $o = t \operatorname{xor} 1$。

同系统内的距离和恒为 $s_t - 1$。对于另一系统中的站点 v：

- 若 u, v 之间有公开线路，则 $D(u, v) = 1$；
- 若二者中至少一个站点的跨系统度数非零，则 $D(u, v) = 2$；
- 否则必须先在 u 所在系统内移动到一个跨系统端点，经过跨系统线路，再在另一系统内移动，故 $D(u, v) = 3$。

题目保证全图在两类移动下连通，因此只要两个系统均非空，就一定存在跨系统公开线路。

于是答案为

$$\mathsf{ans}_u = \begin{cases} s_t - 1 + 2s_o - \mathsf{deg}_u, & \mathsf{deg}_u > 0, \\ s_t - 1 + 3s_o - b_o, & \mathsf{deg}_u = 0. \end{cases}$$

1.2 正确性证明

同系统的两个不同站点显然一步可达。对分属不同系统的 u, v，一步可达当且仅当存在公开线路 (u, v)。若不存在该线路但 $\mathsf{deg}_u > 0$，可以先由 u 沿跨系统线路到达另一系统，再使用通行证到达 v；若 $\mathsf{deg}_v > 0$，过程对称，因此距离为 2。若二者跨系统度数均为零，任何路径都必须先在 u 所在系统内到达某个跨系统端点，并在进入另一系统后从某个跨系统端点到达 v，至少需要三步；连通性又保证这样的三步路径存在。上述分类完整且互斥，逐类计数即得到公式。

1.3 复杂度

时间复杂度为 $\mathcal{O}(n + m)$，空间复杂度为 $\mathcal{O}(n)$。

1.4 部分分

在 $n \le 500$ 时，可以把同系统站点两两连边后使用 Floyd，复杂度为 $\mathcal{O}(n^3)$。特殊性质 A 中所有站点属于同一系统，所有答案均为 $n - 1$。

2 栏目排列（sheet）

2.1 转化为最大权 Hamilton 路

对一行中所有为 1 的栏目集合 S，若暂不考虑相邻合并，该行贡献 $|S|$ 个连续段。排列中每出现一对相邻且均属于 S 的栏目，连续段数恰好减少 1。

令

$$w_{u,v} = \#\{i : \text{第 } i \text{ 行的第 } u, v \text{ 个字符均为 } 1\}.$$

再令 C 为输入矩阵中 1 的总数。对于栏目排列 $p_1, \ldots, p_m$，总代价为

$$C - \sum_{i=1}^{m-1} w_{p_i, p_{i+1}}.$$

因此问题等价于在以栏目为点、边权为 w 的完全图中求最大权 Hamilton 路。

2.2 状态压缩动态规划

设

$$f_{S,v} = \text{恰好经过集合 } S \text{ 中所有点且终点为 } v \text{ 的路径最大边权和.}$$

初值为 $f_{\{v\},v} = 0$，转移为

$$f_{S,v} = \max_{u \in S \setminus \{v\}} \left(f_{S \setminus \{v\},u} + w_{u,v} \right).$$

最大可减去的相邻收益为 $\max_v f_{\{1,\dots,m\},v}$，用 C 减去它即为答案。

2.3 正确性证明

一行中的每个连续段都有唯一的第一个 1，所以段数等于该行 1 的个数减去排列中相邻 1 对数。对所有行求和后，每对相邻栏目 (u, v) 被共同标记的行数正是 $w_{u,v}$，从而得到目标函数。

动态规划的最后一步必然从某个 $u \in S \setminus \{v\}$ 走到 v；删除该步后得到状态 $f_{S \setminus \{v\},u}$。反之，任一该状态的最优路径接上 v 都是合法候选，故转移枚举且仅枚举所有 Hamilton 路。最终取所有终点最大值即得到最大相邻收益。

2.4 复杂度与实现

构造边权的复杂度为 $\mathcal{O}(nm^2)$，状态压缩动态规划的复杂度为 $\mathcal{O}(m^2 2^m)$，空间复杂度为 $\mathcal{O}(m2^m)$。所有状态值不超过 $nm \le 1.8 \times 10^6$，标算特意使用 32 位整数保存大 DP，以降低内存占用并改善常数。

2.5 部分分

$m \le 8$ 时可以枚举排列，但应先算出 w，使单个排列只需 $\mathcal{O}(m)$ 计算。特殊性质 A、B、C 中都存在一个排列使每个非空行恰有一个连续段，答案等于非空行数。将这些判断与 $m \le 15$ 的状态压缩 DP 合并，可通过测试点 $1 \sim 18$。

3 序列消除（erase）

3.1 基本递推

对区间 I 定义 $F(I)$ 为消除方案数，并约定空区间的 F 为 1。若第一次选择数值 x，设 x 在 I 中的出现位置依次为

$$p_1 < p_2 < \cdots < p_t.$$

删除所有 x 后得到 $t+1$ 个间隙。集合中的区间始终按从左到右的确定顺序处理，各间隙的选择序列可以独立决定，因此以 x 为第一次选择的方案数为

$$F([l, p_1 - 1]) \left(\prod_{i=1}^{t-1} F([p_i + 1, p_{i+1} - 1]) \right) F([p_t + 1, r]).$$

这里相邻书写表示乘法。对区间内不同的 x 求和即为 $F(I)$。

直接递归仍然太慢。下面利用值域大小 $K = 20$，把所有会反复出现的中间间隙预处理。

3.2 干净边界区间

若 $p < q$，且开区间 (p, q) 中既没有 a_p 也没有 a_q，定义

$$G(p, q) = F([p + 1, q - 1]).$$

固定右端点 q。对每种值 y，只有 y 在 q 之前的最后一次出现位置 p 可能成为这样的左端点；还需满足 p 不早于 a_q 的上一次出现位置。因此每个 q 至多需要保存 K 个 G 值，总状态数为 $\mathcal{O}(nK)$。

设某个合法 (p, q) 的内部出现过数值 z，其第一次和最后一次出现位置分别为 f_z, l_z。记

$$P_z(f_z, l_z) = \prod_{\substack{u,v \text{ 为区间内} \\ \text{相邻两次 } z \text{ 的出现}}} G(u, v).$$

按照第一次选择的递推，有

$$G(p, q) = \sum_{z \text{ 出现在 } (p,q)} G(p, f_z)\, P_z(f_z, l_z)\, G(l_z, q).$$

空区间的 G 为 1。

按 q 从小到大处理；对同一 q，按左端点 p 从大到小处理。此时 $G(p, f_z)$ 的右端点更小，已经处理；$G(l_z, q)$ 的左端点更大，也已经在当前 q 中处理。故递推没有环。

对每种值，按出现顺序维护相邻间隙 G 的前缀积。由于模意义下前缀积可能为零，标算同时维护零因子个数、非零因子前缀积及其逆元：区间内存在零因子时乘积为零，否则用前缀积之比得到 P_z。

3.3 回答询问

考虑询问 $[l, r]$。对每个在区间中出现的值 x，记第一次、最后一次出现位置为 f_x, l_x，并定义

$$L_x = F([l, f_x - 1]), \qquad R_x = F([l_x + 1, r]).$$

按 f_x 递增计算 L_x。在区间 $[l, f_x - 1]$ 中选择根 y 时，设 p_y 为 f_x 前最后一个 y，则

$$L_x = \sum_{f_y < f_x} L_y\, P_y(f_y, p_y)\, G(p_y, f_x).$$

所有右侧项对应的首次出现位置更早，已经求出。

对称地，按 l_x 递减计算 R_x。若 s_y 是 l_x 后第一个 y，则

$$R_x = \sum_{l_y > l_x} G(l_x, s_y)\, P_y(s_y, l_y)\, R_y.$$

最终再枚举第一次选择的值：

$$F([l, r]) = \sum_x L_x\, P_x(f_x, l_x)\, R_x.$$

3.4 正确性证明

基本递推来自第一次选择值的唯一分类；删除该值后得到的极长段互不相交，且后续处理顺序唯一，所以各段方案数相乘。

G 的递推只是把同一基本递推应用于 (p, q) 内部。由于内部没有端点对应的两种值，任一根 z 的首段、相邻 z 间隙、尾段分别恰为 $G(p, f_z)$、$P_z(f_z, l_z)$、$G(l_z, q)$，没有遗漏或重复。

计算 L_x 时，区间 $[l, f_x - 1]$ 内任一可能的根 y 都满足 $f_y < f_x$；其首段是 L_y，中间段乘积为 P_y，尾段满足干净边界条件，正是 $G(p_y, f_x)$。因此公式完整枚举该区间的根。R_x 对称成立。最后一个公式再次按整个询问区间的第一次选择值分类，所以得到且仅得到所有消除方案。

3.5 复杂度

预处理 G 需要 $\mathcal{O}(nK^2+n\log \mathrm{mod})$ 时间、$\mathcal{O}(nK)$ 空间；每次询问需要 $\mathcal{O}(K^2)$ 时间。这里 $K=20$，故总复杂度为

$$\mathcal{O}\big(nK^2+n\log \mathrm{mod}+qK^2\big),$$

空间复杂度为 $\mathcal{O}(nK)$。

3.6 部分分

长度不超过 8 时可直接按第一次选择递归，并将值按首次出现顺序离散化后记忆化；这同时覆盖测试点 $1\sim 6$。当区间只含一种数时答案为 1；只含两种不同数时第一次选择只有两种，选定后所有剩余段均为常值段，答案为 2。据此可处理单点、值域不超过 2 和全相同序列。

4 停机检查（halt）

4.1 默认边与特殊事件

忽略判断相等的分支时，每个状态 i 都沿默认边

$$i \longrightarrow e_i$$

移动，并使 x 增加 d_i。这些边构成一张函数图。

当在状态 i 命中 $x=a_i$ 时，执行特殊边后配置一定变成

$$(c_i, a_i+b_i),$$

它只由事件状态 i 决定，与此前历史无关。因此只需建立 N 个“特殊事件”组成的新函数图：对每个事件后的固定配置，寻找沿默认边遇到的下一个特殊事件、终止状态，或确认永远不会再遇到事件。

4.2 函数图树上的批量询问

删除默认函数图的环边，并把终止状态 $N+1$ 也视作一个根，可得到若干棵内向树。设 u 的默认父亲为 e_u，定义势能

$$g_{e_u}=g_u+d_u.$$

标算从根反向遍历，等价地令 $g_u=g_{e_u}-d_u$。

从配置 (v,x) 沿默认边走到祖先 u 时，变量值为

$$x+g_u-g_v.$$

于是恰在 u 命中特殊事件当且仅当

$$a_u-g_u=x-g_v.$$

左侧是节点固定键值，右侧是询问键值。将节点和询问按键值分组；在每组中按 DFS 序扫描，用栈维护当前点的所有候选祖先，栈顶就是离询问起点最近的命中点。这样可离线回答初始配置和所有事件后配置共 $N+1$ 次询问。

若树路径上没有命中点，走到终止根时直接得到终止步数；走到函数图环时继续处理环。

4.3 环上的命中

设某个环按默认边方向为 $c_0, c_1, \ldots, c_{L-1}$，定义

$$P_0 = 0, \qquad P_i = \sum_{j=0}^{i-1} d_{c_j}, \qquad S = \sum_{j=0}^{L-1} d_{c_j},$$

以及 $B_i = a_{c_i} - P_i$。

从环上位置 s、变量值 x 开始，令 $z = x - P_s$。第 k 个标准圈次到达位置 i 时命中当且仅当

$$B_i = z + kS.$$

若 $S = 0$，只需在所有 $B_i = z$ 的位置中寻找循环顺序上严格位于 s 之后的第一个位置；若不存在，则变量和状态以后完全重复，程序不终止。

若 $S \neq 0$，必须有 $B_i \equiv z \pmod{|S|}$。固定余数 r，令

$$q_i = \frac{B_i - r}{S}, \qquad V_i = i + Lq_i,$$

并对起点同样定义 $V_0 = s + L(z - r)/S$。命中所需步数恰为 $V_i - V_0$，所以只需在同余类内寻找最小的 $V_i > V_0$。对每个环按“余数、V”排序即可二分。中间量可能达到 10^{23}，标算使用 `__int128` 比较，步数只在加入答案时取模。

4.4 事件图

若事件 i 后的配置首先到达事件 j，就在事件图中连边 $i \to j$，边权为“执行事件 i 的一步”加上中间默认步数。若直接终止或永不再命中事件，则记录对应结果。

事件图仍是函数图。若路径进入事件环，某个事件后的完整配置会再次出现，故程序永不终止；否则沿边累加到终止节点即可。初始配置先通过同一批量询问找到首个事件，再接上事件图答案。

4.5 正确性证明

任意两次特殊事件之间，程序只会沿默认函数图运行，所以“默认段压缩为下一事件或结局”不会改变执行顺序。势能等式精确描述树路径上的变量变化，离线祖先查询选择的是路径上最先命中的状态。若没有树上命中，程序必然进入对应环；环公式枚举了每个状态的所有未来访问圈次，并选择严格为正的最小时间，因此得到环上的首次命中或正确判定永不命中。

特殊事件后的配置仅由事件编号决定，所以事件图中的同一节点代表完全相同的程序配置。事件图出现环当且仅当某配置重复，此后执行无限循环；无环路径最终到达终止结局。故算法对终止性和步数均正确。

4.6 复杂度与部分分

函数图分解、Euler 序和事件图处理为线性，排序与二分为 $\mathcal{O}(N \log N)$，空间复杂度为 $\mathcal{O}(N)$。

特殊性质 B 中答案恒为 1。性质 C 可固定 x 后在状态函数图上模拟；性质 D 的状态编号严格增加；性质 E 的下一状态与判断结果无关，这三类都可在线性时间内处理。朴素保存 (state, x) 配置只适合小规模，无法处理变量单调漂移但在极大步数后才命中的情况。