第2日　Birthday　解説

Solution

やらなければならないことは, どの子供も移動する椅子の数の最大数を最小にするような座席表を見つけることである. まず, 最終的な座席表には二つの可能性があることに注意する. たとえば, 順列 $(1, 2, 3)$ が与えられているとき, 子供 1 は, 子供 2 の左隣にも右隣にもなり得る. 前者を反時計回りの座席表, 後者を時計回りの座席表と呼ぶことにする. どちらの場合も計算は同じようにできるので, 時計回りの座席表だけを考えることにする. 生徒は両方の場合を考え, 小さくなる方の結果を選ばなければならない.

Simple solution

第一のアイディアはすべての可能な席替えのシミュレーションを実行することだろう. 一番の子供の位置を固定しよう. 与えられた順列を使って, すべての子供たちの最終位置 (と移動距離と) を, 時間計算量 $O(n)$ で, 計算できる. 一番の子供の可能な位置すべてにこのステップを実行しなければならないから, 全体の時間計算量は $O(n^2)$ である.

Optimal solution

もう少し良い解がある. 与えられた子供たちの順列を (p_i) で表すことにしよう. 子供 p_1 の最終位置が f である子供たちの最終的な座席表を考える. この座席表のように座るためには, 子供たちの何人か (すべてかもしれない) は移動しなければならない. 子供 i の移動を数 d_i^f で表現することにする. d_i^f は, $\left|d_i^f\right|$ が移動距離 (移動する椅子の数) で, 子供が時計回りに移動するとき正, 反時計回りに移動するとき負であるような数である. さらに, 子供たちは常に距離が小さい方の方向に (もし両方向の距離が等しいときは時計回りに) 移動することを選択すると仮定する. したがって, $1 - \lceil \frac{n}{2} \rceil \leq d_i^f \leq \lfloor \frac{n}{2} \rfloor$ である.

$S_f = \left\{ d_i^f : i = 1, 2, \ldots, n \right\}$ とする. S_f を考えている席替えの別の表現として扱う. この表現が得られれば,

$$R_f = \max\left(-\min\left(S_f\right), \max\left(S_f\right)\right)$$

という式を用いて, 移動距離の最大値を計算することができる. d_i^f の値は, 与えられた順列 (p_i) と f に依存する. それらは,

$$a = (f + i - 1 - p_i) \mod n \text{ として } d_{p_i}^f = \min\left(a, n - a\right)$$

という式で決まる. さらに, 与えられた表現 S_f から, “シフトすること” により簡単に S_{f+1} を計算できる (つまり, $x < \lfloor \frac{n}{2} \rfloor$ ければ x を $x + 1$ に, ちょうど $\lfloor \frac{n}{2} \rfloor$ のときは $1 - \lceil \frac{n}{2} \rceil$ に置き換える).

f のとりえる値すべてに対しての結果 (R_f) の最小数を知りたい. どんな表現 S_f も元になる表現, それを S_0 としよう, のシフトしたものになっていることに注意しよう. C によって, $\left\{1 - \lceil \frac{n}{2} \rceil, \ldots, \lfloor \frac{n}{2} \rfloor\right\}$ のなかで, S_0 の中には現れない数で, (mod n で考えて) 連続する数の個数の最大数を表すことにする.

結果は, $\lfloor \frac{n-C}{2} \rfloor$ である. これが時間計算量 $O(n)$ のアルゴリズムである.