

D: ReverseSort

問題: 藪
解答作成: 大坂、藪
解説: 藪

問題概要

- ▶ 1～Nまでの数字を用いた順列がある
 - ▶ ある区間を逆順にする操作を元に昇順に並び変えたい
 - ▶ 必要な最小操作回数は？
-
- $1 \leq N \leq 10$

解法(1 / 3)

- ▶ 両側幅優先探索
- ▶ N個の数字をソートするとき、必要な操作回数は高々N-1回
 - ソートされていない一番小さい数値を正しい位置にもってくるような操作を繰り返す

解法(2/3)

- ▶ $N=10$ の時、最悪でも9回の施行でソートが完了する
- ▶ $[1, 2, \dots, N]$ から4回の施行で辿り付ける順列と、 $[A_1, A_2, \dots, A_N]$ から4回の施行で辿り付ける順列をそれぞれBFSで計算する
 - 1回の施行で $N^2/2$ 通りの施行の仕方があるので4回だと $N^8/8$ 通り
 - 1回の施行には $O(N)$
 - よって全体では $O(N^9)$ だが、定数がとても小さいので十分間に合う

解法(3 / 3)

- ▶ 二つの共通部分が見つかったら、その2つの和の最小値が答え
- ▶ 共通部分が見つからなかったら答えは9
- ▶ 全体の計算回数はBFSの計算回数と同じ

ジャッジ解

- ▶ 藪 C++(82行)
- ▶ 大坂 Java(119行)

解答状況

- ▶ First Acceptance
 - Onsite: agitsune (51分)
 - All: agitsune (51分)
- ▶ First Submission
 - Onsite:C_hokudai(38分)
 - All: C_hokudai(38分)
- ▶ AC/Submission
 - 8/39 (21%)