CS2109S Tutorial 2

AY 25/26 Sem 1 — github/omgeta

A. 1. States are all unique permutations of the original array

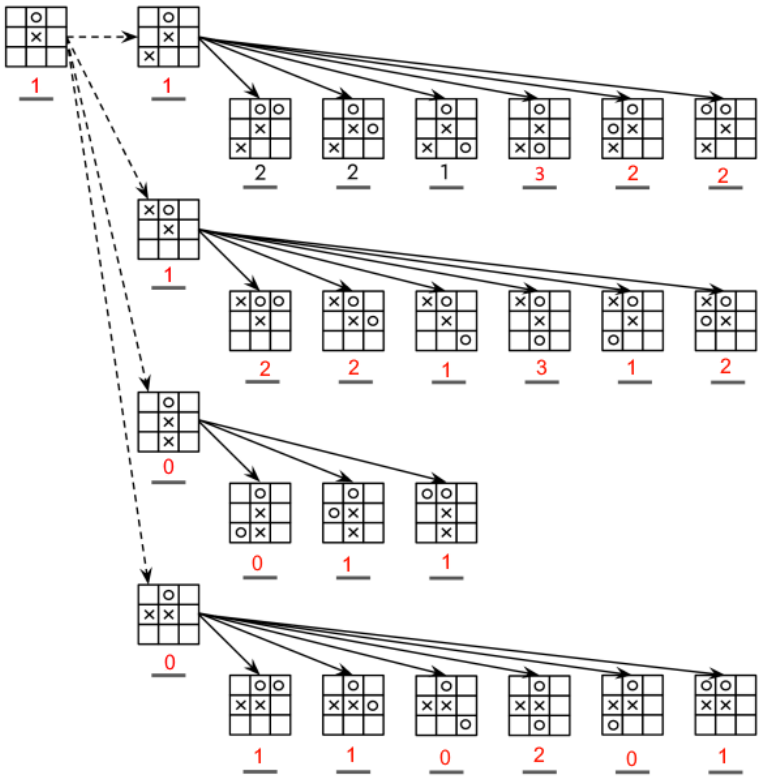
2. Initial: original array A
   Goal: sorted ascending array A'

3. Use Inversion Count $I(s) = \#\{i < j : a_i > a_j\}$. Goal reached when $I(s) = 0$

4. Successors are adjacent swaps of out-of-order neighbour elements, such that each swap changes $I(s) \pm 1$

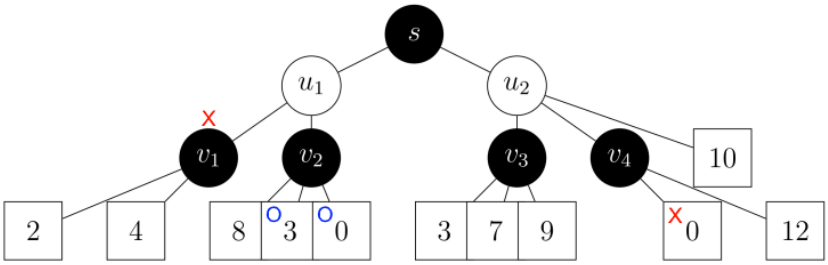
5. No; state space is too large for feasible informed search

B. 1.



2. Cache results of visited states so that if revisted later, we already know the result. Once a line contains both X and O, it can never contribute to the score anymore so it doesn't have to be reevaluated deeper.

C.



D. 1. $A < 9$

2. $X \geq 11$