

Spring 2024

## CS 412 (Algorithms: Design and Analysis)

### Weekly Challenge 02: Sorting

Announced: Friday, January 19, 2024.

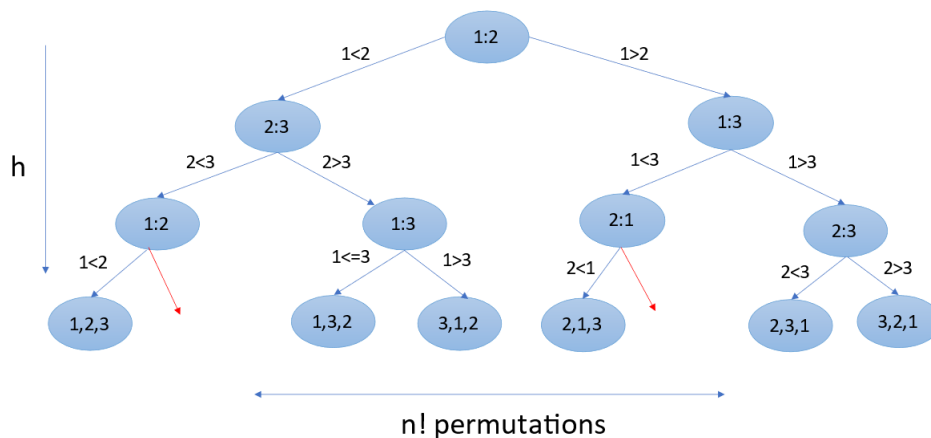
Deadline: Friday, January 26, 2024 (11:59 pm PKT).

Total marks: 1.

Instructions: Submit **individually** your solution as a PDF with the file name as your *studentID.pdf*; typeset in LaTeX. You must submit your solution on Canvas.

- (1 point) A decision tree is a binary tree that represents the comparisons between elements that are performed by a particular sorting algorithm operating on an input of a given size. Control, data movement, and all other aspects of the algorithm are ignored. The leaf nodes represent all possible permutations (arrangements) of the input array. For example, the following is a decision tree of the bubble sort algorithm for  $n = 3$ , where  $n$  is the total number of elements in an array.

### Decision tree (Bubble Sort)



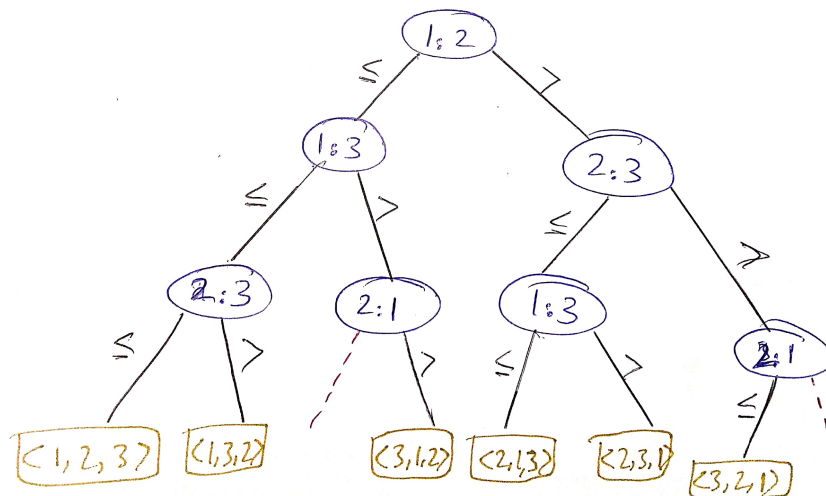
Take  $n = 3$  and compare the efficiency of selection sort and insertion sort, in terms of the number of comparisons, with the help of a decision tree when the array is:

1. Fully sorted
2. Fully unsorted

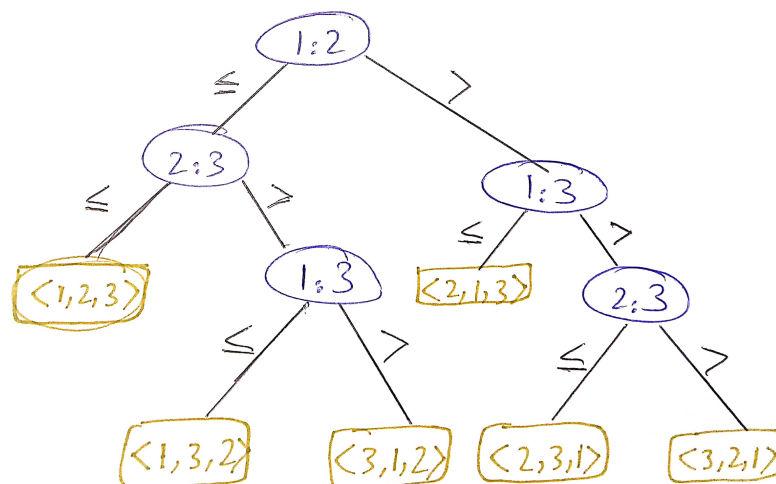
Please draw the decision tree for both algorithms and give your analysis in three to four lines.

**Solution:**

## Selection Sort



## Insertion Sort



When we look at it, Selection sort took 3 comparisons in both best and worst cases, while insertion sort took 3 comparisons in the fully unsorted one, while 2 comparisons in the sorted one. This aligns with the idea that the worst and best case complexity of Selection sort is  $O(n^2)$  and the worst case complexity of Insertion Sort is  $O(n^2)$  and best case is  $O(n)$