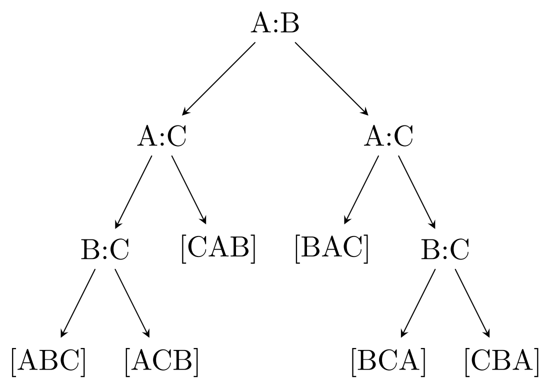
Sorting Complexity

Every comparison-based sort has a corresponding decision tree. Consider the following decision tree for an arbitrary sorting algorithm to sort a 3-element list `[A, B, C]`.



At every point in the tree, a comparison-based sort by definition will compare 2 elements. The leaves in the tree are all possible orderings for the list. For a decision tree to represent all permutations of the array, there need to be $n!$ number of leaves. For a binary decision tree there is an upper bound on the number of leaves possible:

$$L \leq 2^H \tag{1}$$

Substituting $L = n!$ into the above inequality and using Stirling's Formula to expand the factorial gives

$$H \geq n \lg n + O(n) \tag{2}$$

Because the height of the tree is the number of comparisons done, the optimal (lower-bound) number of comparisons for a comparison-based sort is $\Theta(n \lg n)$.