

1 Apple Picking

Suppose without loss of generality you have an array of apples, granny smith, pink lady, red delicious etc, and you wish to find the type of apple that comprises over 50% of the array. As apples do not have an order, you cannot sort the array or perform a comparison. Devise a divide and conquer algorithm to solve this problem.

2 Box Nesting

Suppose you have a sequence of boxes each with a width and depth, (w_i, d_i) for $i = 1, \dots, n$. Devise an algorithm to determine the maximum number of boxes that can be nested inside each other, where a box can be nested inside another if and only if both its width and depth are strictly smaller than those of the other box.¹

3 k -Selection Percentile

Suppose you have a black box which given an array of integers $A[1..n]$ returns an element $a \in A[1..n]$ such that at least $n/10$ of the elements in $A[1..n]$ are less than or equal to a and at least $n/10$ of the elements in $A[1..n]$ are greater than or equal to a . Devise an algorithm to find the k -th smallest element in $A[1..n]$ using this black box. What is its runtime?

4 Middle DAG

Suppose you have a directed acyclic graph (DAG), $G = (V, E)$, devise an algorithm which finds a vertex $v \in V$ such that at most half of other vertices in V are reachable from v and at most half of the other vertices in V can reach v .

¹Hint: LIS