Divide and Conquer

Design and analyze means:

- Give pseudocode for your algorithm.
- Prove that your algorithm is correct.
- Give the recurrence relation for the running time for your algorithm.
- Solve the recurrence relation.

1. Rank the following running times in order from fastest to slowest:
   - $17n$
   - n^3
   - 2^n
   - $n!$
   - 36
   - $\log_b(n)$
   - $48n \log_b(n)$
   - $367n^2 \log_b(n)$
   - n^n

2. Solve the following recurrence relations:
   - $T(n) = T(n/2) + O(n)$
   - $T(n) = T(n/2) + O(n^2)$
   - $T(n) = 3T(n/3) + O(1)$
   - $T(n) = T(n/3) + O(n)$

3. In justifying our matrix multiplication algorithm, we accepted the following property: If X and Y are $n \times n$ matrices and
   $X = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \qquad Y = \begin{bmatrix} E & F \\ G & H \end{bmatrix}$
   where A, B, C, D, E, F, G and H are $n/2$ by $n/2$ matrices, then the product XY can be expressed as:
   $XY = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} E & F \\ G & H \end{bmatrix} = \begin{bmatrix} AE + BG & AF + BH \\ CE + DG & CF + DH \end{bmatrix}$
   Prove this property.

4. Suppose that you are given a sorted list of distinct integers $\{a_1, a_2, \ldots a_n\}$. Design and analyze a divide-and-conquer algorithm that determines whether there exists an index i such that $a_i = i$. For example, in $\{-10, -4, 3, 41\}$, $a_3 = 3$, and in $\{4, 7, 19, 20\}$ there is no such i.

5. Suppose you are given 3^n marbles that look identical, with one special marble that weighs more than the other marbles. You are also given a balancing scale that takes two items (or sets of items) and compares their weights. Design and analyze a divide and conquer algorithm to find the heavy marble using the balancing scale at most n times.

6. Suppose that you are given an arithmetic progression with one element of the progression missing. Design and analyze a divide and conquer algorithm to find the missing number.

   (https://en.wikipedia.org/wiki/Arithmetic_progression)

   Example: Suppose you are given the following arithmetic progression with common difference 4: $\{3, 7, 11, 19, 23, 27, 31, 35, 39, 43\}$. The missing number is 15. Or if you are given the arithmetic progression with common difference 2: $\{0, 2, 4, 6, 8, 12, 14\}$, the missing number is 10.

7. Suppose that you are given an integer, x, and a sorted list of integers. Design and analyze a divide and conquer algorithm that counts the number of occurrences of x in the list.

   Example: Suppose $x = 4$ and the list is $\{1, 2, 2, 2, 4, 4, 12, 20, 20, 20\}$. Your algorithm should return 2.

8. Suppose that you are given a list of n elements. Design and analyze a divide and conquer algorithm to remove all duplicates from the list in time $O(n \log n)$.

9. A list A is said to have a *majority element* if more than half of its entries are the same. There is not necessarily an order on the list, so there can't be comparisons of the form "Is $A[i] \leq A[j]$?". However, in constant time, the question "Is $A[i] = A[j]$?" can be answered. Given an array, design a divide and conquer algorithm to determine if there is a majority element, and, if so, to find that element. Your algorithm should run in time $O(n \log n)$.

10. Suppose we are given a list of n numbers representing stock prices on a single day. We want to find a pair $(buy, sell)$ with $buy \leq sell$ such that if we bought the stock on *buy* day and sold the stock on *sell* day, we would maximize our profit.

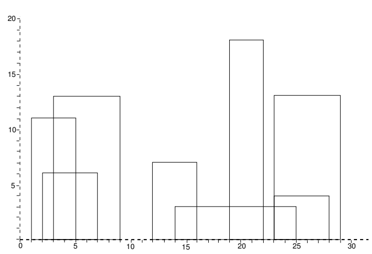
    Design and analyze a divide and conquer algorithm that finds the optimal $(buy, sell)$ pair.

11. In this problem we find the closest pair of points in the Euclidean plane. Suppose you are given a set of n points in the plane. The goal is to find the two points that are the closest. Recall that the distance between two points, $a = (a_x, a_y)$ and $b = (b_x, b_y)$ is $\sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$.

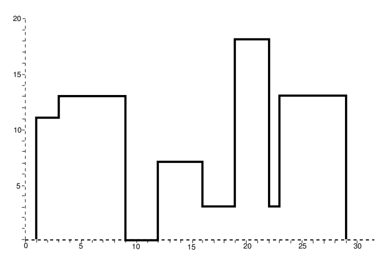
    Design and analyze a divide and conquer algorithm to solve this problem.

12. We will compute *skylines* in this problem. A building, B_i, is given as a triplet (L_i, H_i, R_i) where L_i and R_i denote the left and right x-coordinates of the building and H_i represents the height. A *skyline* of a set of buildings is a list of x coordinates and the heights connecting them arranged in order from left to right.

    Example: Given: $\{(3, 13, 9), (1, 11, 5), (12, 7, 16), (14, 3, 25), (19, 18, 22), (2, 6, 7), (23, 13, 29), (23, 4, 28)\}$, the skyline is: $\{1, 11, 3, 13, 9, 0, 12, 7, 16, 3, 19, 18, 22, 3, 23, 13, 29, 0\}$



buildings



skyline

    Design and analyze a divide-and-conquer algorithm to compute the skyline for a list of n buildings.

13. You are given a list of numbers. Your goal is to return the sum of the contiguous sublist of numbers that has the largest sum.

    For example, if you are given the list $\{1, -4, 3, 7, -5, 6, -9, 5\}$, your algorithm should return 11.

    Design and analyze a divide and conquer algorithm to solve this problem.

14. The *Towers of Hanoi* puzzle is defined as follows: You are given a stack of n disks arranged from largest on the bottom to smallest on top, placed on a rod, together with two empty rods. The Towers of Hanoi puzzle asks for the minimum number of moves required to move the stack from one rod to another, where moves are allowed only if they place smaller disks on top of larger disks. Design and analyze a "reduce-by-one-and-conquer" algorithm to solve this problem.

15. Suppose that you are given a stack of n pancakes of different sizes. You want to sort the pancakes so that smaller pancakes are on top of larger pancakes. The only operation that you can perform is a *flip*: insert a spatula under the top k pancakes and flip them all over. Design and analyze a "reduce-by-one-and-conquer" algorithm to sort the pancake stack.