# Lab Assignment 9 (Divide and Conquer Approach)

## 1. Merge k Sorted Lists

https://leetcode.com/problems/merge-k-sorted-lists/

You are given an array of `k` linked-lists `lists`, each linked-list is sorted in ascending order.

*Merge all the linked-lists into one sorted linked-list and return it.*

**Example 1:**

```
Input: lists = [[1,4,5],[1,3,4],[2,6]]

Output: [1,1,2,3,4,4,5,6]

Explanation: The linked-lists are:

[

  1->4->5,

  1->3->4,

  2->6

]

merging them into one sorted list:

1->1->2->3->4->4->5->6
```

## 2. Maximum Subarray

https://leetcode.com/problems/maximum-subarray/

Given an integer array `nums`, find the contiguous subarray (containing at least one number) which has the largest sum and return *its sum*.

**Follow up:** If you have figured out the `O(n)` solution, try coding another solution using the **divide and conquer** approach, which is more subtle.

**Example 1:**

```
Input: nums = [-2,1,-3,4,-1,2,1,-5,4]

Output: 6

Explanation: [4,-1,2,1] has the largest sum = 6.
```

**Example 2:**

```
Input: nums = [1]

Output: 1
```

### 3. Beautiful Array

https://leetcode.com/problems/beautiful-array/

For some fixed `N`, an array `A` is *beautiful* if it is a permutation of the integers `1, 2, ..., N`, such that:

For every `i < j`, there is **no** `k` with `i < k < j` such that `A[k] * 2 = A[i] + A[j]`.

Given `N`, return **any** beautiful array `A`. (It is guaranteed that one exists.)

**Example 1:**

```
Input: 4

Output: [2,1,4,3]
```

**Example 2:**

```
Input: 5

Output: [3,1,2,5,4]
```

**Note:**

- `1 <= N <= 1000`

### 4. K Closest Points to Origin

https://leetcode.com/problems/k-closest-points-to-origin/

We have a list of `points` on the plane. Find the `K` closest points to the origin `(0, 0)`.

(Here, the distance between two points on a plane is the Euclidean distance.)

You may return the answer in any order. The answer is guaranteed to be unique (except for the order that it is in.)

**Example 1:**

```
Input: points = [[1,3],[-2,2]], K = 1

Output: [[-2,2]]

Explanation:

The distance between (1, 3) and the origin is sqrt(10).

The distance between (-2, 2) and the origin is sqrt(8).

Since sqrt(8) < sqrt(10), (-2, 2) is closer to the origin.

We only want the closest K = 1 points from the origin, so the answer is just [[-
2,2]].
```

**Example 2:**

```
Input: points = [[3,3],[5,-1],[-2,4]], K = 2

Output: [[3,3],[-2,4]]

(The answer [[-2,4],[3,3]] would also be accepted.)
```

**Note:**

1. `1 <= K <= points.length <= 10000`
2. `-10000 < points[i][0] < 10000`

### 5. Different Ways to Add Parentheses

https://leetcode.com/problems/different-ways-to-add-parentheses/

Given a string of numbers and operators, return all possible results from computing all the different possible ways to group numbers and operators. The valid operators are `+`, `-` and `*`.

**Example 1:**

```
Input: "2-1-1"

Output: [0, 2]

Explanation:

((2-1)-1) = 0

(2-(1-1)) = 2
```

**Example 2:**

```
Input: "2*3-4*5"
```

**Output:** [-34, -14, -10, -10, 10]

**Explanation:**

(2*(3-(4*5))) = -34

((2*3)-(4*5)) = -14

((2*(3-4))*5) = -10

(2*((3-4)*5)) = -10

(((2*3)-4)*5) = 10