

2023-04-08 - Handout – Arrays

Q1. Two Sum

Link: <https://leetcode.com/problems/two-sum/>

Given an array of integers `nums` and an integer `target`, return indices of the two numbers such that they add up to `target`. You may assume that each input would have exactly one solution, and you may not use the same element twice. You can return the answer in any order.

Example 1:

Input: `nums = [2,7,11,15]`, `target = 9`

Output: `[0,1]`

Explanation: Because `nums[0] + nums[1] == 9`, we return `[0, 1]`.

Example 2:

Input: `nums = [3,2,4]`, `target = 6`

Output: `[1,2]`

Q2. Combinations Sum

Link: <https://leetcode.com/problems/combination-sum/>

Given an array of distinct integers `candidates` and a target integer `target`, return a list of all unique combinations of `candidates` where the chosen numbers sum to `target`. You may return the combinations in any order.

The same number may be chosen from `candidates` an unlimited number of times. Two combinations are unique if the frequency of at least one of the chosen numbers is different.

The test cases are generated such that the number of unique combinations that sum up to `target` is less than 150 combinations for the given input.

Example 1:

Input: `candidates = [2,3,6,7]`, `target = 7`

Output: `[[2,2,3],[7]]`

Explanation: 2 and 3 are candidates, and $2 + 2 + 3 = 7$. Note that 2 can be used multiple times. 7 is a candidate, and $7 = 7$. These are the only two combinations.

Example 2:

Input: `candidates = [2,3,5]`, `target = 8`

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

Q3. Subsets

Link: <https://leetcode.com/problems/subsets/>

Given an integer array `nums` of unique elements, return all possible subsets (the power set). The solution set must not contain duplicate subsets. Return the solution in any order.

Example 1:

Input: `nums = [1,2,3]`

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

Example 2:

Input: `nums = [0]`

Output: `[[],[0]]`

Q4. Median of Two Sorted Arrays

Link: <https://leetcode.com/problems/median-of-two-sorted-arrays/description/>

Given two sorted arrays `nums1` and `nums2` of size `m` and `n` respectively, return the median of the two sorted arrays. The overall run time complexity should be $O(\log(m+n))$.

Input: `nums1 = [1,3]`, `nums2 = [2]`

Output: 2.00000

Explanation: merged array = `[1,2,3]` and median is 2.

Input: `nums1 = [1,2]`, `nums2 = [3,4]`

Output: 2.50000

Explanation: merged array = `[1,2,3,4]` and median is $(2 + 3) / 2 = 2.5$.