1. A porcupine number is a prime number whose last digit is 9 and the next prime number that follows it also ends with the digit 9. For example, 139 is a porcupine number because:

   a. it is a prime

   b. it ends in a 9

   c. The next prime number after it is 149 which also ends in 9.

   Write a method named `findPorcupineNumber` which takes an integer argument n and returns the first porcupine number that is greater than n. So `findPorcupineNumber(0)` would return 139 (because 139 happens to be the first porcupine number) and so would `findPorcupineNumber(138)`. But `findPorcupineNumber(139)` would return 409 which is the second porcupine number.

   #procupine_number

2. 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.

   #two_sum

Input	Output
nums =[2,7,11,15], target =9	[0,1], because nums[0]+nums[1]==9
nums = [2,3,4], target = 6	[0,2]
nums=[3,3], target = 6	[0,1]

3. Write a function named `mostFrequent()` that given an array, finds the most frequent element in it. If there are multiple elements that appear maximum number of times, returns any one of them.

   #most_frequent

4. Given an array of integers, write a function that finds the pair in array whose sum is closest to 0. If there are more than one pair, return any pair.

   For e.g., [4,2, -1,3,-2,-3] → [2,-2]

   #closest_to_zero

5. Write a function, that given two arrays representing two non-negative integers with digits, adds the two numbers, and return it as an array.

   #add_two_numbers

Input	Output	Explanation
[3,4,2], [4,6,5]	[7,0,8]	342+465=807
[1,1], [1,2,3]	[1,3,4]	11+123=134
[1,1], [9,9]	[1,1,0]	11+99=110

6. Given two binary strings, return their sum (also a binary string).

   The input strings are both non-empty and contains only characters 1 and 0.

   #add_binary

Input	Output
a="11", b="1"	"100"
a="1010", b ="1011"	"10101"

7. Given a positive integer, return its corresponding column title as appear in an Excel Sheet.

   For example:

   1 →A, 2→B, 3→C,…,26→z,27→AB, →28->AB, …

   #excel_sheet_column_title

Input	Output
1	A
28	AB
701	ZY

8. Given a column title as appear in an excel sheet, return its corresponding column number.

   #excel_sheet_column_number

Input	Output
A	1
AB	28
ZY	701

9. A strobogrammatic number is a number that looks the same when rotated 180 degree (looked upside down)

   Write a function to determine if a number is strobogrammatic. The number is represented as string.

   #strobogrammatic_number

Input	Output
69	true
101	true
88	true
868	false

10. Given an array containing n distinct numbers taken from 0,1,2,…,n, find the one that is missing from the array.

    #missing_number

Input	Output
[3,0,1]	2
[9,6,4,2,3,5,7,0,1]	8

11. Given a sorted array and a target value, return the index if the target is found. If not, return the index where it would be if it were inserted in order.

    You may assume no duplicates in the array.

    #search_insert_position

Input	Output
[1,3,5,6], 5	2
[1,3,5,6], 2	1
[1,3,5,6], 7	4
[1,3,5,6], 0	0

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

    #maxinum_subarray

Input	Output	Explanation
[-2,1,-3,4,-1,2,1,-5,4]	6	[4,-1,2,1]

13. Given an array of integers and an integer k, find out whether there are two distinct indices i and j in the array such that nums[i] = nums[j] and the absolute difference between i and j is at most k.

    #contains_duplicate_ii

Input	Output
nums=[1,2,3,1], k=3	true
nums=[1,0,1,1], k=1	true
nums= [1,2,3,1,2,3], k=2	false

14. Given a list of words and two words word1 and word2, return the shortest distance between these two words in the list.

    #shortest_word_distance

    Assume the words ["practice", "makes", "perfect", "coding", "makes]

Input	Output
word1 = "coding", word2 = "practice"	3
word1 = "makes", word2= "coding"	1

Note: You may assume that word1 does not equal to word2, and word1 and word2 are both in the list.

15. Given an array nums, write a function to move all 0's to the end of it while maintaining the relative order of the non-zero elements.

    #move_zeroes

    Example:

    Input: [0, 1, 0, 12, 3]

    Output: [1,3,12,0,0]

    Note: you must do this in-place without making a copy of the array.

16. Write a function that takes a string as input and reverse only the vowels of a string.

    # reverse_vowels_of_a_string

Input	Output
"hello"	"holle"
"leetcode"	"leotcede"

17. Write a function summaryRanges, such that given a sorted integer array nums, returns the smallest sorted list of ranges that cover all the numbers in the array exactly. That is, each element of nums is covered by exactly one of the ranges, and there is no integer x such that x is in one of the ranges but not in nums.

    Each range [a,b] in the list should be output as:

    "a->b" if a != b

    "a" if a == b

    #summary_ranges

Input	Output
[0,1,2,4,5,7]	["0->2","4->5","7"]
[0,2,3,4,6,8,9]	["0","2->4","6","8->9"]
[]	[]
[-1]	["-1"]
[0]	["0"]

18. For two strings s and t, we say "t divides s" if and only if s = t + ... + t (t concatenated with itself 1 or more times)

    Given two strings str1 and str2, return the largest string x such that x divides both str1 and str2.

    #greatest_common_divisor_of_strings

Input	Output
str1= "ABCABC", str2="ABC"	"ABC"
str1="ABABAB", str2="ABAB"	"ABAB"
str1="ABCDEF", str2="ABC"	""