Leetcode -> 448.

=> Find all numbers disappeared in an Array.

Algorithm:-

1. Iterate through the array.
2. for each elements nums[i], mark the element at index abs(nums[i])-1 as -ve.
3. Iterate through the array again.
4. For each +ve element, its corresponding index plus one is a missing no.
5. Store the missing numbers in a seperate array
6. Return the array containing the missing numbers.

Program:-

```
int * findDisappearedNumbers(int *nums, int numsSize
                             int * returnSize){
    int *disappered;
    int *count = (int *)malloc(numsSize * sizeof(int))
    for(i=0; i<numsize; i++){
        counts(nums[i]-1)++;
    }
    int missingCount=0;
    for(i=0; i<numsSize; i++){
        if(counts[i]==0){
            missingCount++;
        }
    }
    disappeared = (int*)malloc(missingCount * sizeof(int
    *returnSize = missingCount;
```

```
    int index = 0;
    for (i = 0; i < numsSize ; i++) {
        if (counts[i] == 0) {
            disappeared[index++] = i+1;
        }
    }
    free (counts);
    return disappeared;
}
```

O/P :-

nums = [4, 3, 2, 7, 8, 2, 3, 1]
O/P : [5, 6]
Expected : [5, 6].

nums : [1, 1]
O/P : [2]
Expected : [2].

9/5/24