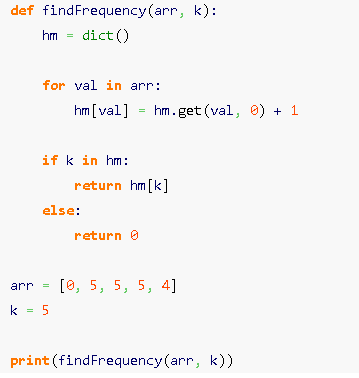
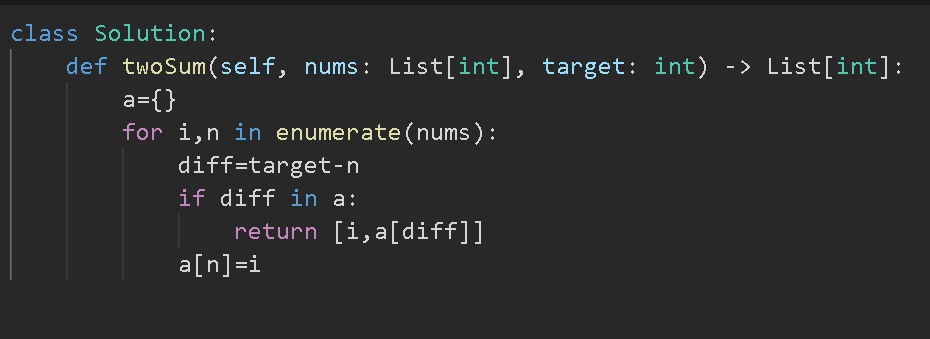
Question link :

<https://www.geeksforgeeks.org/find-frequency-number-array/>



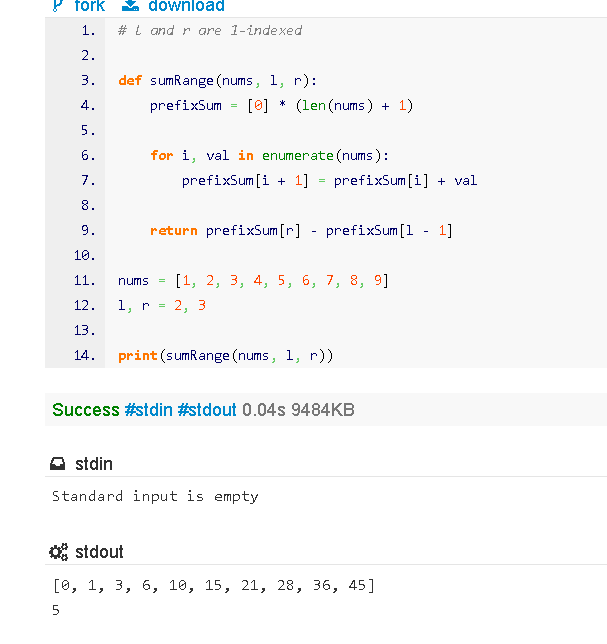
Question link:

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



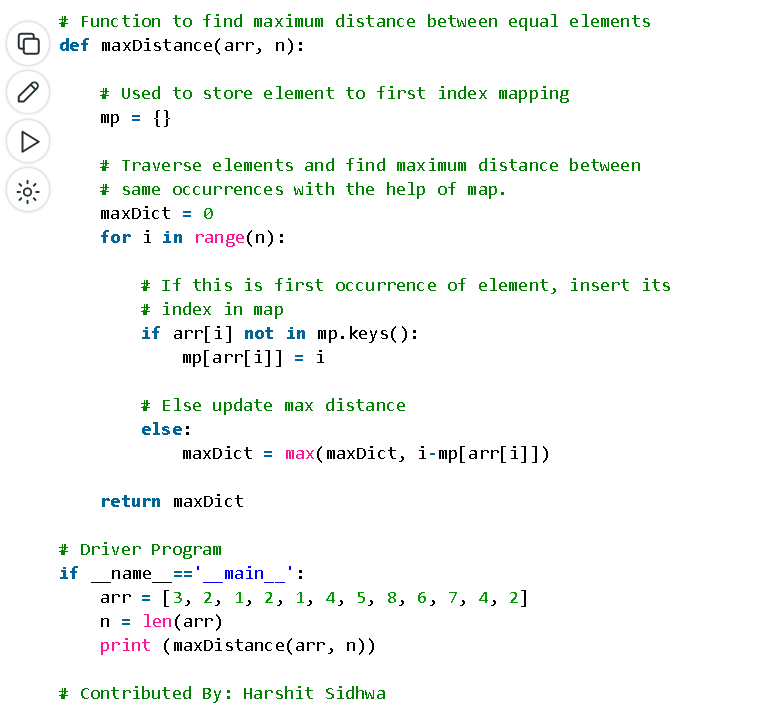
Question link:

<https://www.spoj.com/problems/CSUMQ/>



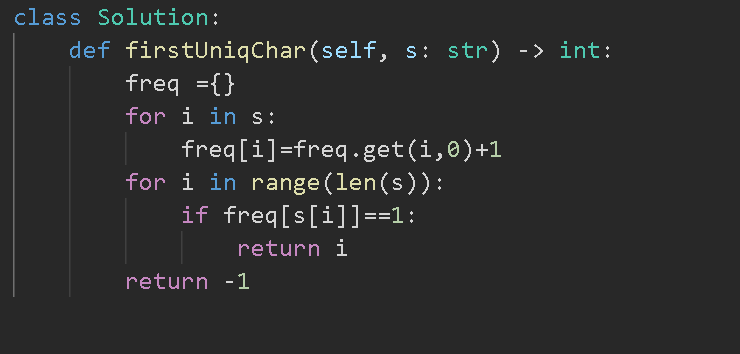
Question link:

<https://www.geeksforgeeks.org/maximum-distance-two-occurrences-element-array/>



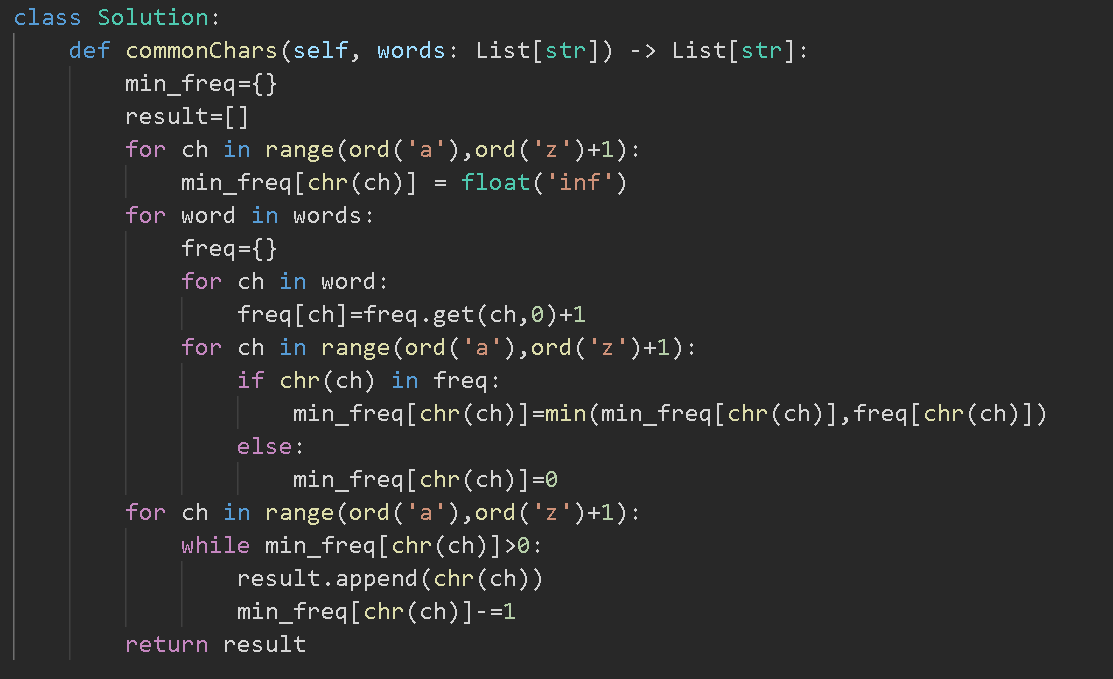
Question link:

<https://leetcode.com/problems/first-unique-character-in-a-string/>



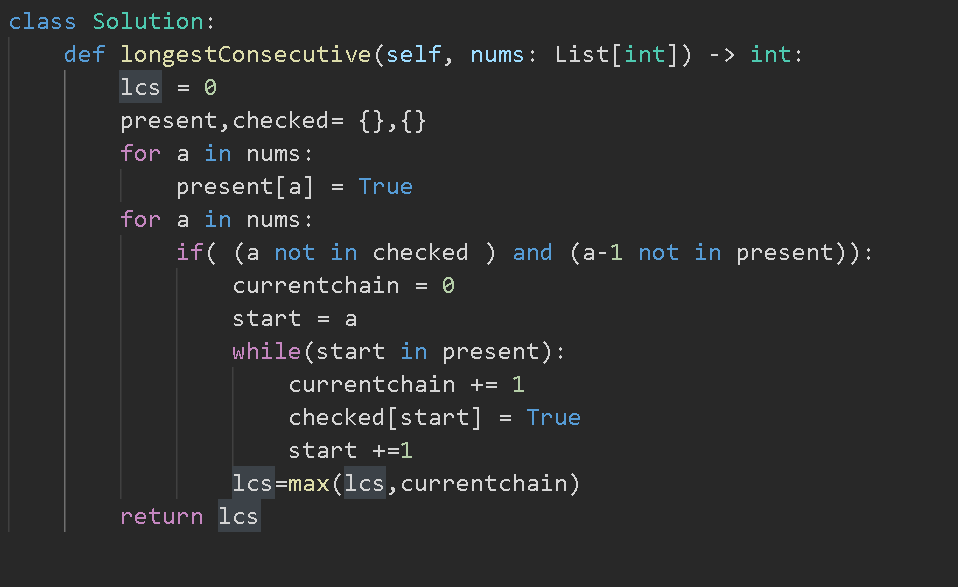
Question link:

<https://leetcode.com/problems/find-common-characters/>



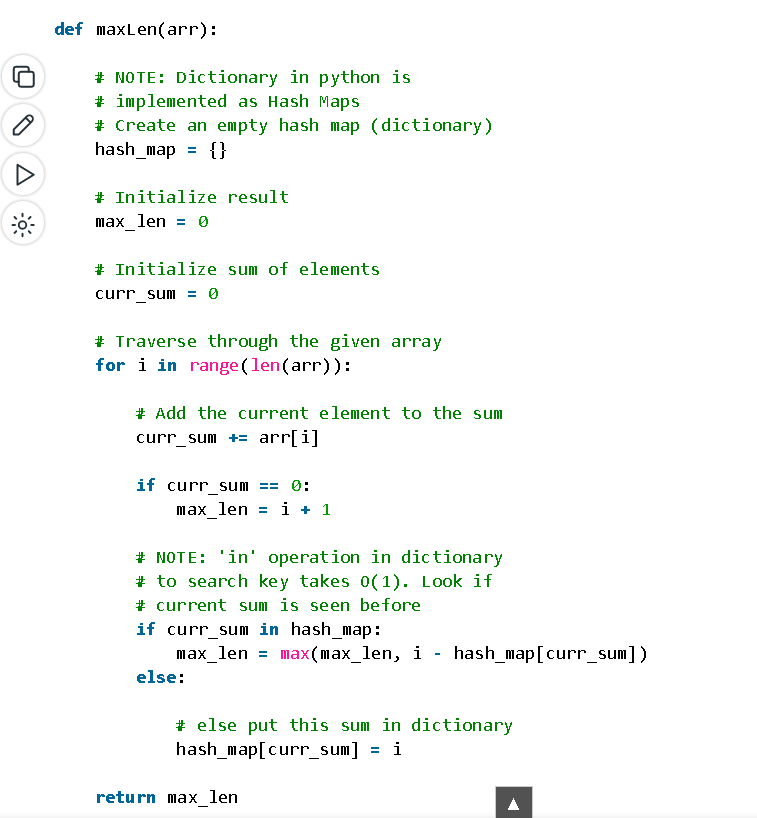
Question link:

<https://leetcode.com/problems/longest-consecutive-sequence/>



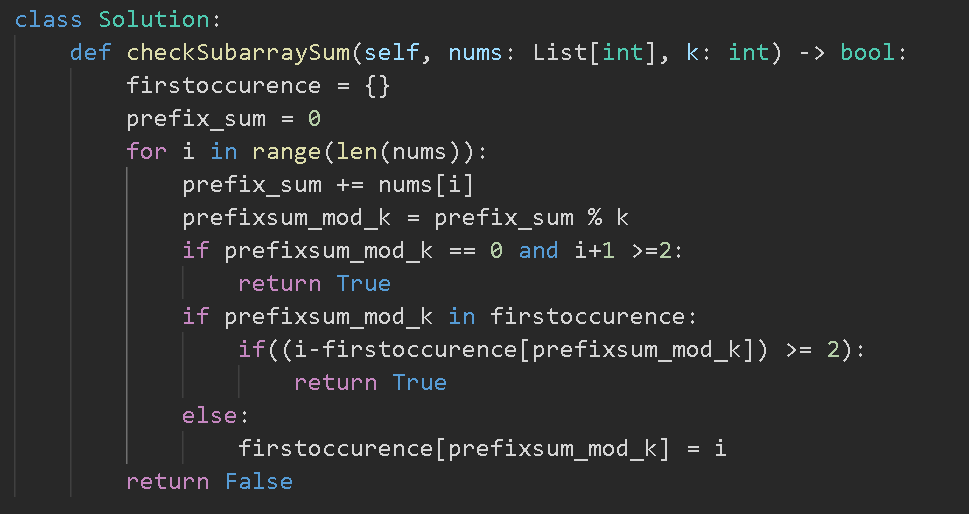
Question link:

<https://www.geeksforgeeks.org/find-the-largest-subarray-with-0-sum/>



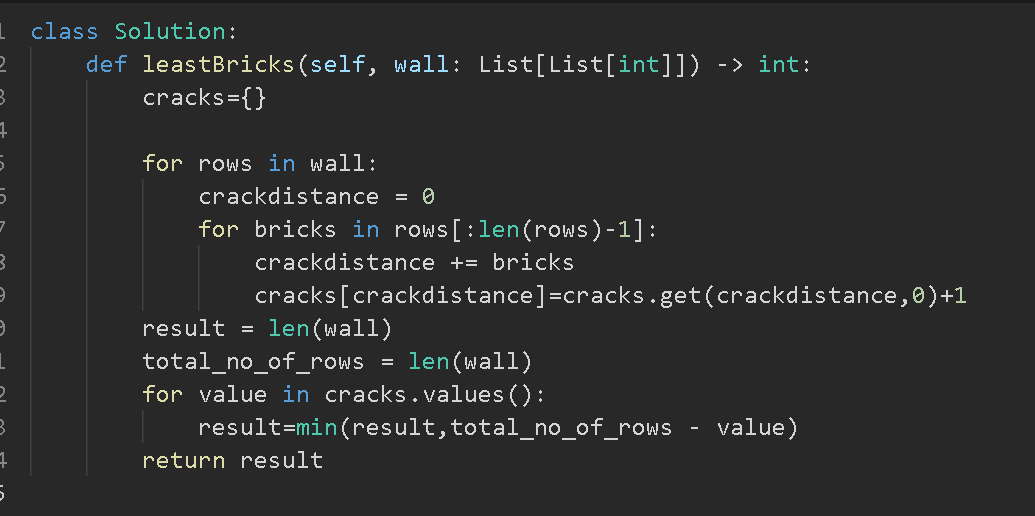
Question link:

<https://leetcode.com/problems/continuous-subarray-sum/description/>



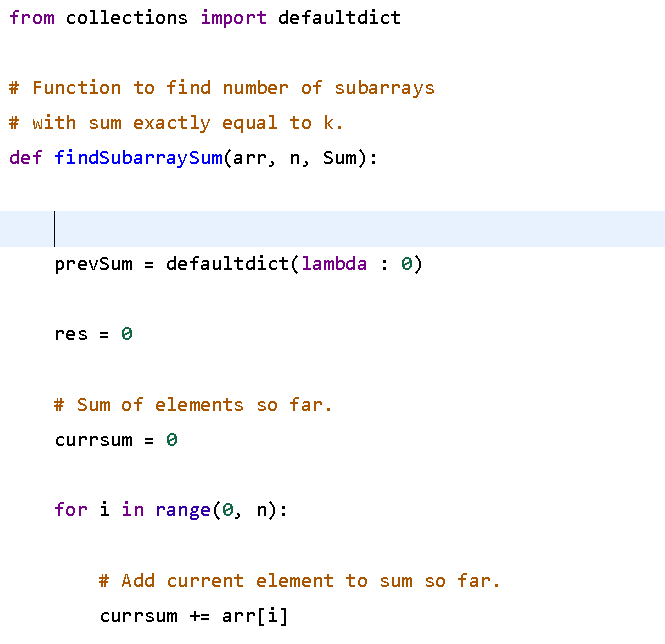
Questionlink:

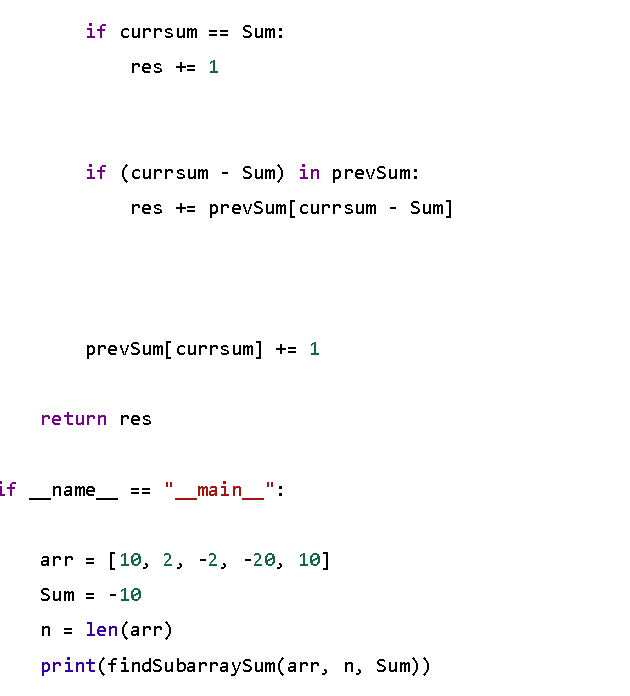
<https://leetcode.com/problems/brick-wall/description/>



Question link:

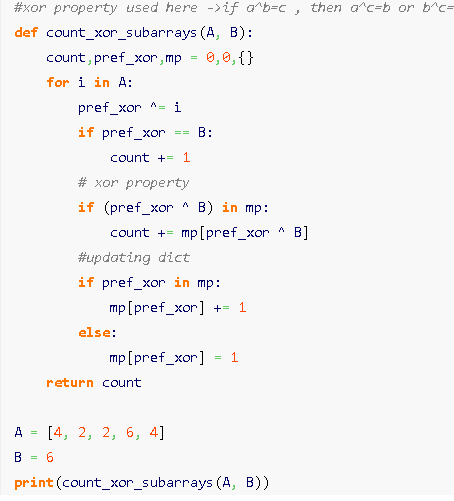
<https://www.geeksforgeeks.org/number-subarrays-sum-exactly-equal-k/>





Question link:

<https://www.geeksforgeeks.org/count-number-subarrays-given-xor/>



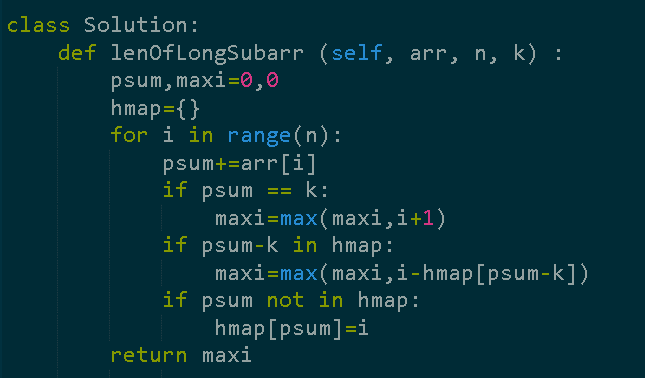
Question link:

<https://www.geeksforgeeks.org/smallest-subarray-with-sum-k-from-an-array/>

see the note for code

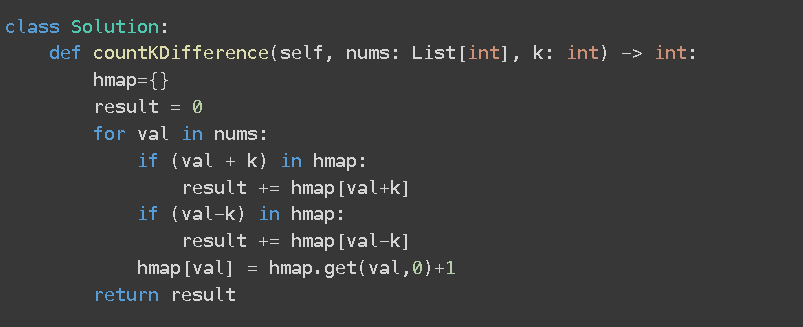
Question link:

<https://practice.geeksforgeeks.org/problems/longest-sub-array-with-sum-k0809/1?utm_source=gfg&utm_medium=article&utm_campaign=bottom_sticky_on_artic>



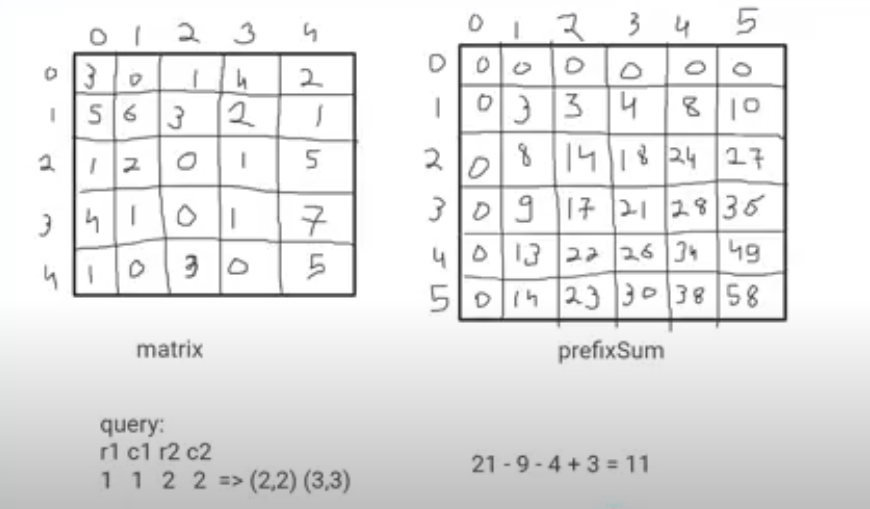
Question link:

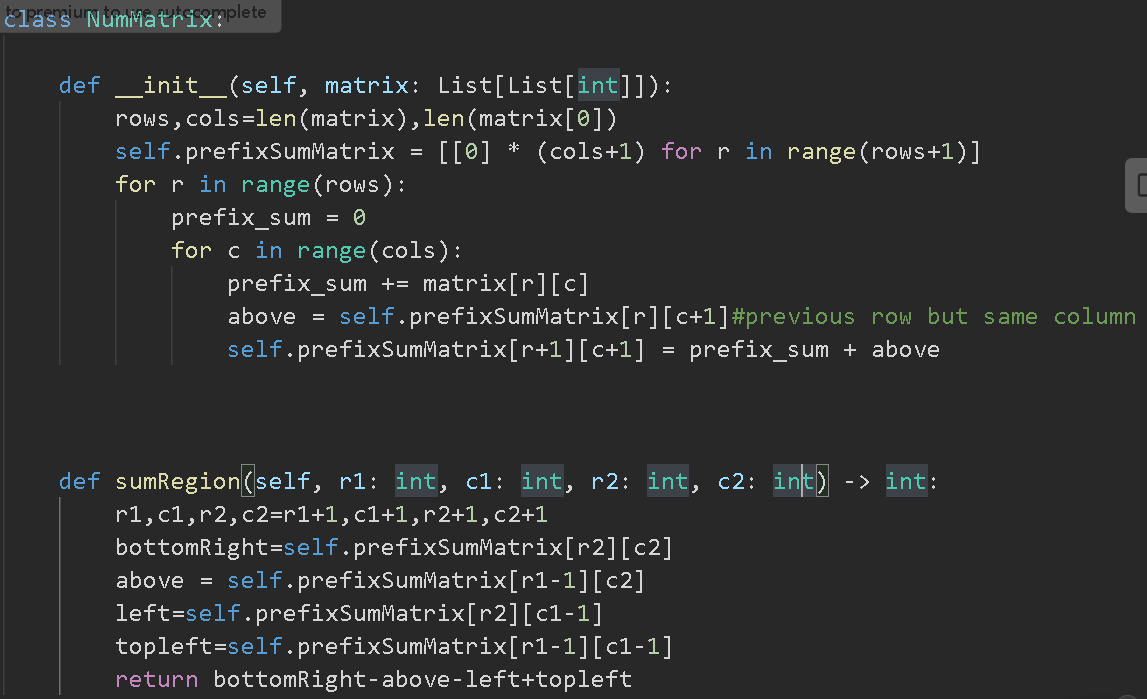
<https://leetcode.com/problems/count-number-of-pairs-with-absolute-difference-k/submissions/919396377/>



Question link:

<https://leetcode.com/problems/range-sum-query-2d-immutable/description/>





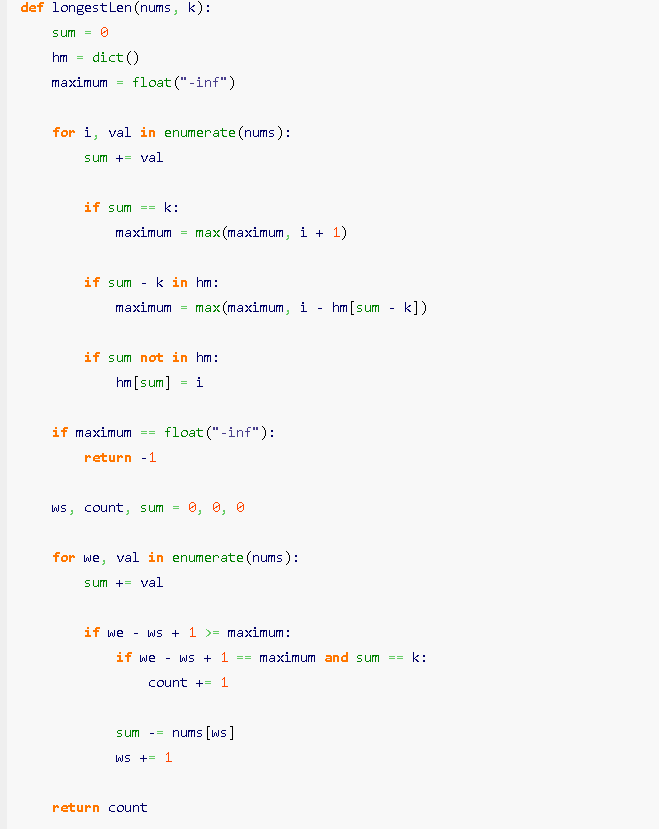
**Q. Find the count. of subarray of** longest **length whose sum==k**

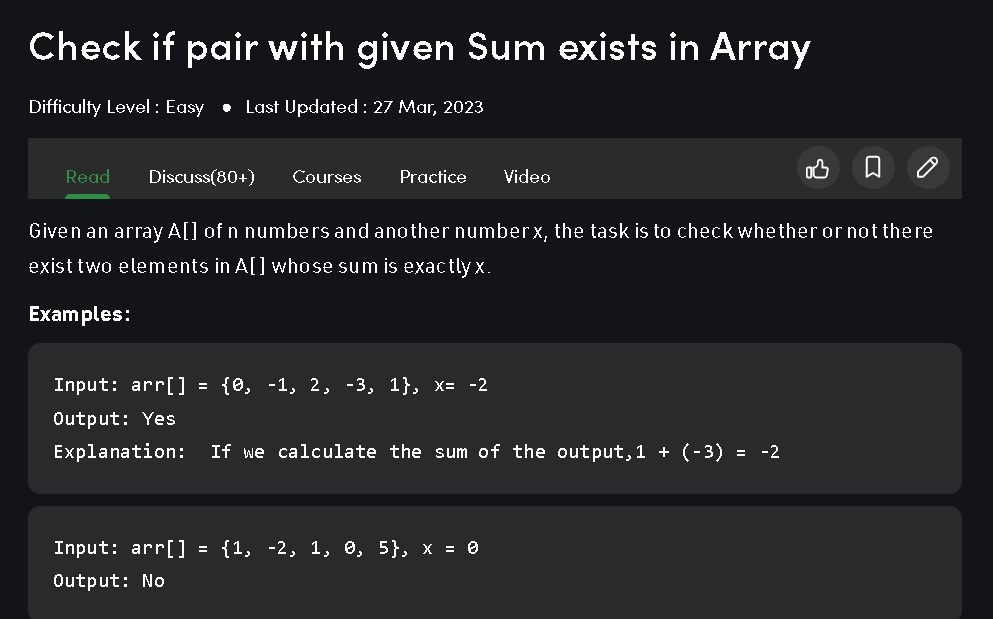
**A = {10,5,2,7,1,9,8,7}   k = 15**

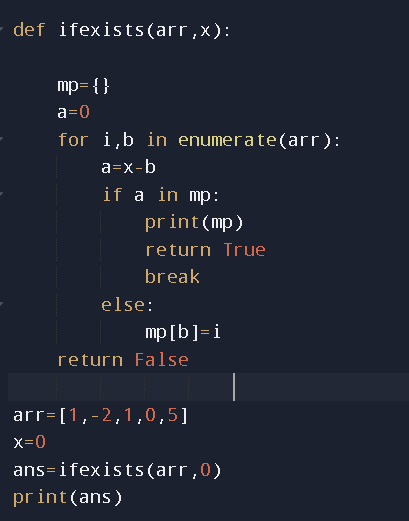
**O/p: 4**

Eg. subarrays having sum =15 , {10,5} , {5,2,7,1} , {8,7}

Longest length is 4

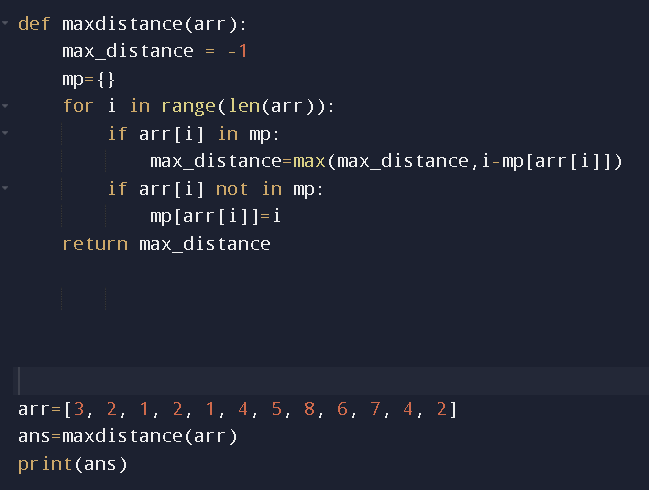


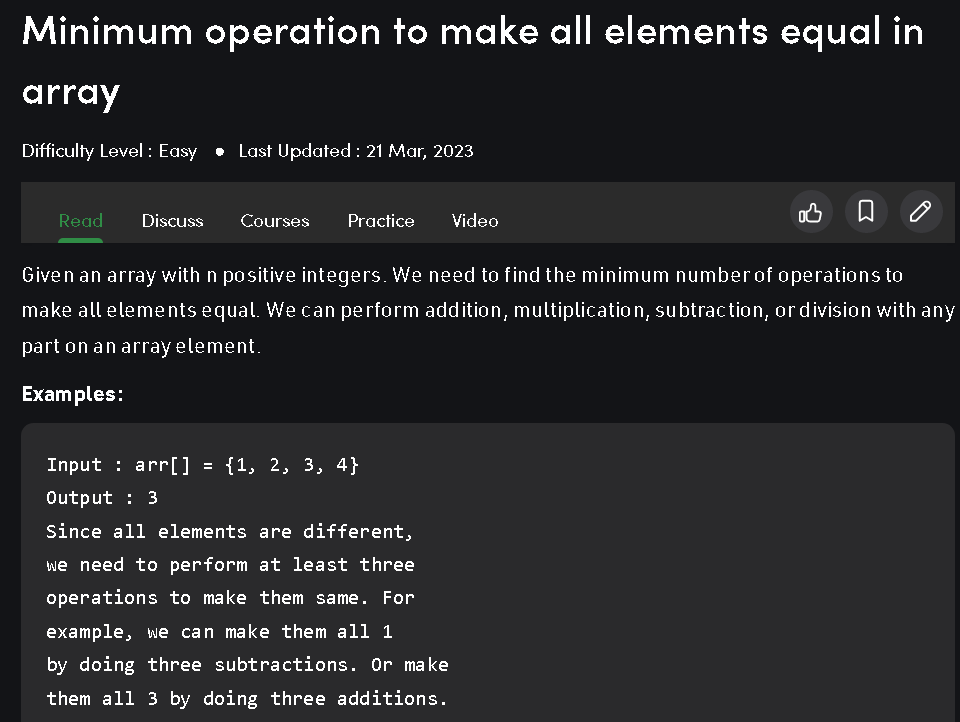


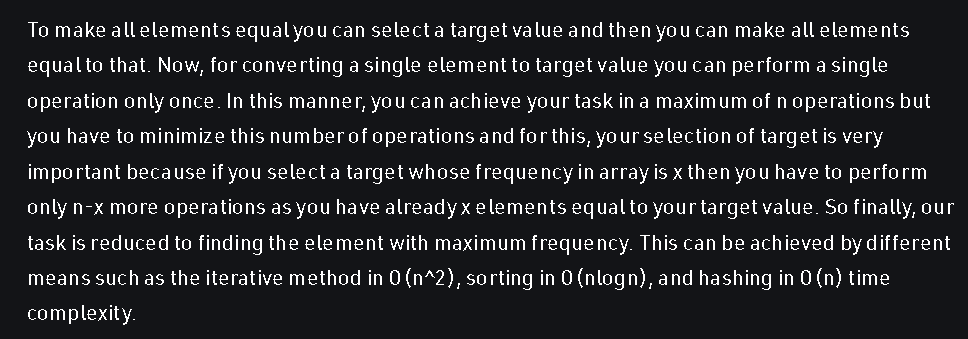


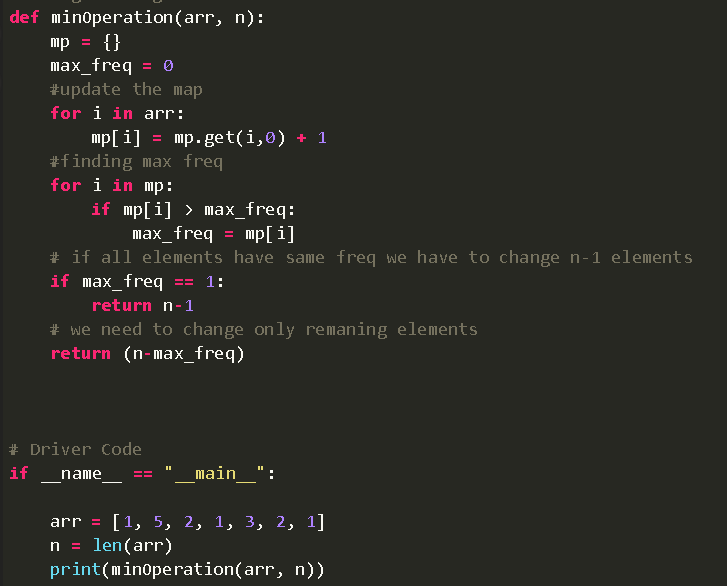
Problem link:

<https://www.geeksforgeeks.org/maximum-distance-two-occurrences-element-array/>





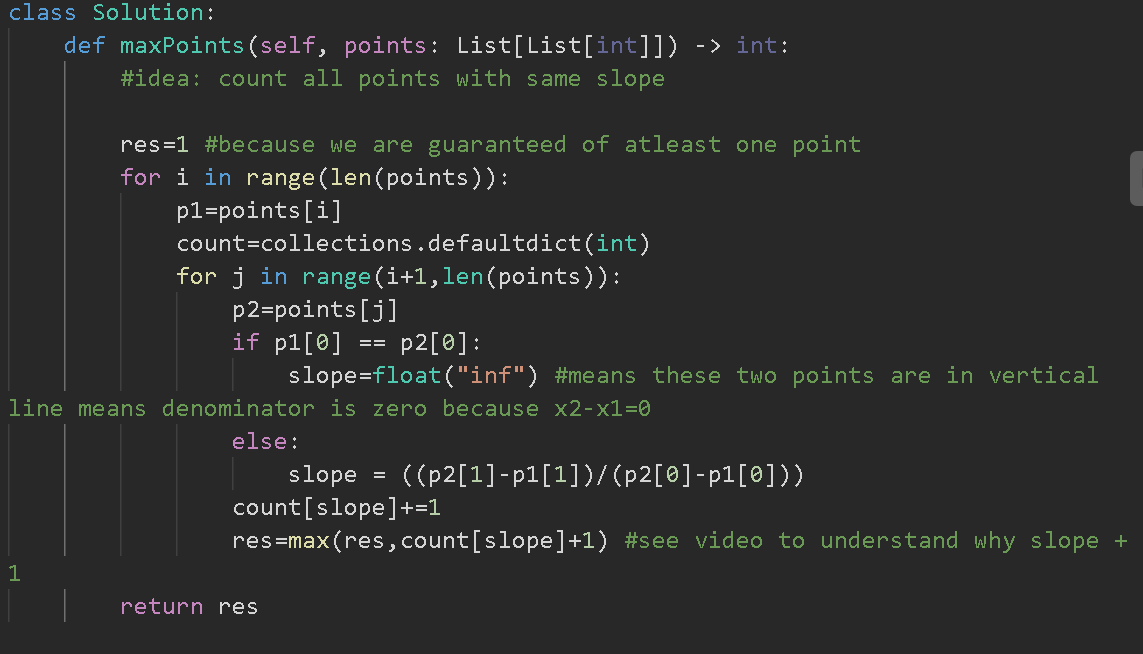




Max\_freq because we need to find the minimum number of ways

Problem link:

<https://leetcode.com/problems/max-points-on-a-line/description/>



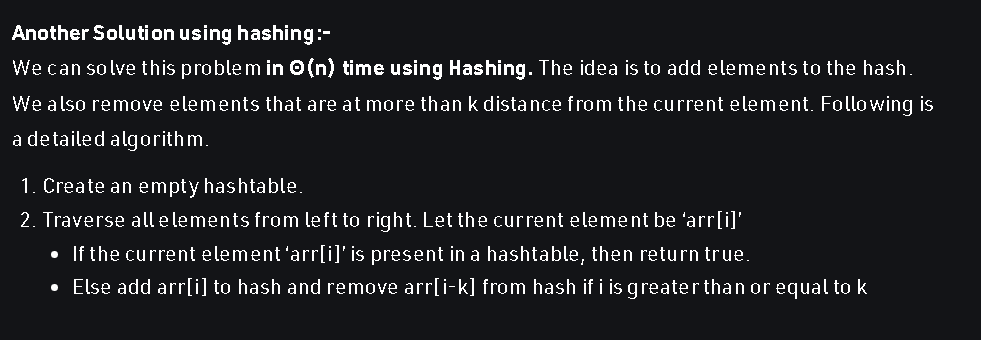
count[slope]+1 because we are counting the number of points in the questions

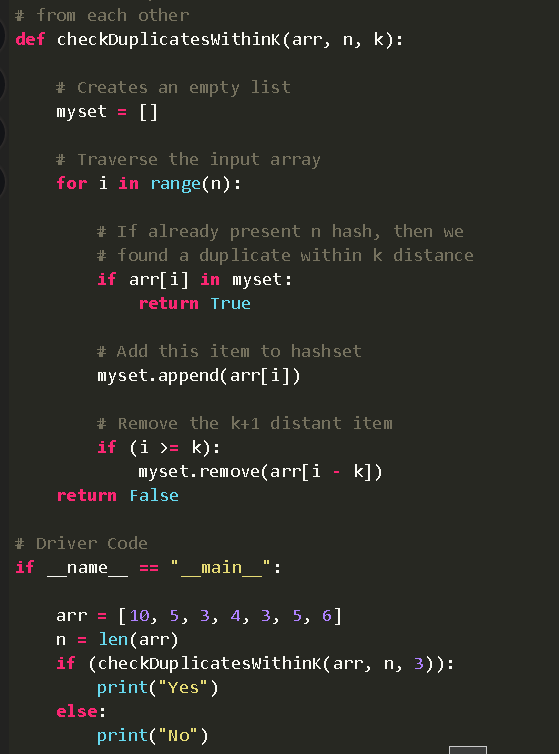
and we also know that

no. of points = frequency of the slope + 1

Problem link:

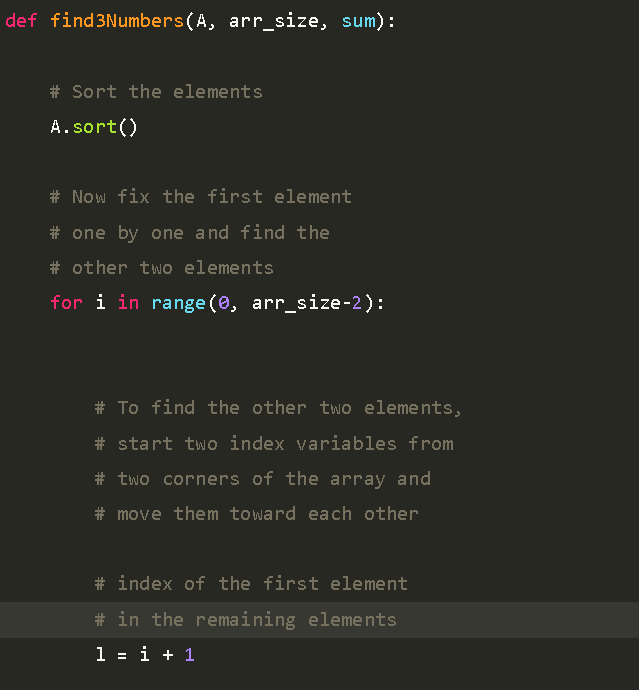
<https://www.geeksforgeeks.org/check-given-array-contains-duplicate-elements-within-k-distance/?ref=lbp>





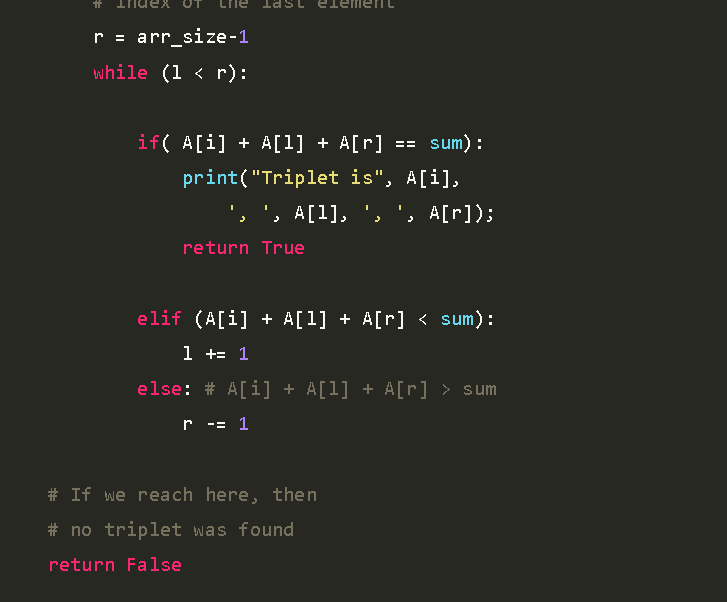
Problem link:

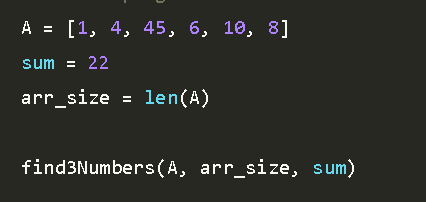
<https://www.geeksforgeeks.org/find-a-triplet-that-sum-to-a-given-value/>

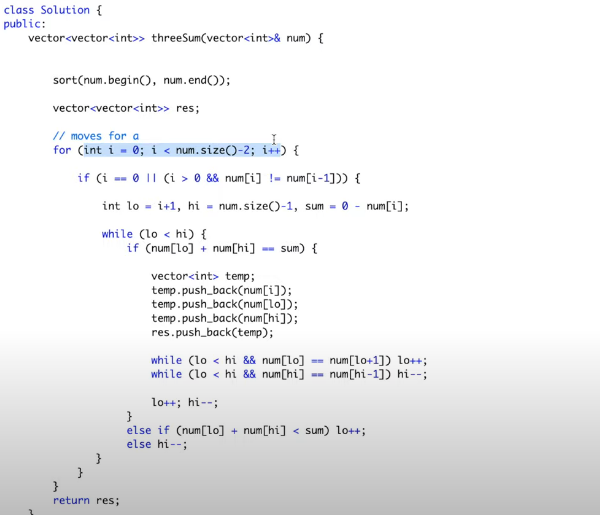


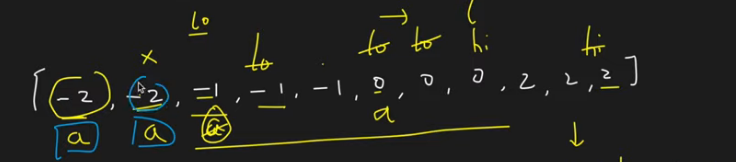
Arr\_size -2 in the loop because the first number of the triplet cannot be in the last two elements of the array , because third number is not possible

if the first element starts with the last two elements.





****

****

**Nums[i]!=nums[i-1] to avoid duplicates**

Google OA

-> <https://www.desiqna.in/13378/google-oa-sde2-march-2022>

We are given array of size N ; find the quadruplets such that a[i] < a[j] < a[k] < a[l] and i<j<k<l ;

N<=1000

Understanding :-

A = [ 1 2 3 4 5 ]

Q -> {1 2 3 4}

{1 2 3 5}

{2 3 4 5}

{1 3 4 5}

{1 2 4 5}

Answer = 5

Brute - Force :- O(n^4) ; iterate over all the quadruplets using 4 nested for loops.

Pseudo Code : -  <https://ideone.com/mFajjY>

C++

Java

Python

Little Better :- Lets fix the index k and analyze ;

i<j<k<l

K = 5

-> you are at a[k=5] ; what do you need to find to get the valid pairs such that k is always fixed at 5

[b1 b2 b3 b4 b5 b6 b7 b8 b9 b10] ; N = 10

You are at K = 5 ?

-> you need to find all elements a[l] such that a[k]<a[l] ; l will lie in the range (k+1,n)

-> hence in O(N) time you can calculate all indices l such that k < l and a[k] < a[l] for a particular K = 5

-> lets call this number ar right.

-> what else is there to find ?

-> now for each particular k ; you need to find pair(i,j) such that i < j ; a[i]<a[j]<a[k] ;

You can simply do this by iterating all the pairs in (1,k-1) -> lets call this number as left\_pairs.O(N^2)

-> so the number of valid quadruplets for a particular k is left\_pairs\*right

-> For each index k :- O(N^2 + N)

-> BUT THERE ALL N POSSIBILITIES OF K : - O(N\*(N^2 + N)) = O(N^3+N2) ; N^3>>>>>N^2 ;

TC :- O(N^3)

Pseudo Code :- <https://ideone.com/ABV6qn>