Two Sum

Description

Hints

Submissions

Discuss

Solution

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]

Example 3:

Input: nums = [3,3], target = 6

Output: [0,1]

Constraints:

2 <= nums.length <= 104

-109 <= nums[i] <= 109

-109 <= target <= 109

Only one valid answer exists.

Follow-up: Can you come up with an algorithm that is less than O(n2) time complexity?

Seen this question in a real interview before?

Subscribe to see which companies asked this question.

Related Topics

Similar Questions

3Sum4SumTwo Sum II – Input Array Is SortedTwo Sum III – Data structure designSubarray Sum Equals KTwo Sum IV – Input is a BSTTwo Sum Less Than KMax Number of K-Sum PairsCount Good MealsCount Number of Pairs With Absolute Difference KNumber of Pairs of Strings With Concatenation Equal to TargetFind All K-Distant Indices in an ArrayFirst Letter to Appear TwiceNumber of Excellent PairsNumber of Arithmetic TripletsNode With Highest Edge ScoreCheck Distances Between Same LettersFind Subarrays With Equal SumLargest Positive Integer That Exists With Its NegativeNumber of Distinct AveragesCount Pairs Whose Sum is Less than Target

Python3

1

Class Solution:

2

3

Def twoSum(self, nums, target):

4

5

# Create a hashmap to store the indices of elements

6

7

Num\_indices = {}

8

9

# Iterate through the array

10

11

For I, num in enumerate(nums):

12

13

# Calculate the complement needed to reach the target

14

15

Complement = target – num

16

17

# Check if the complement is already in the hashmap

18

19

If complement in num\_indices:

20

21

# If found, return the indices of the two numbers

22

23

Return [num\_indices[complement], i]

24

25

# Otherwise, add the current number and its index to the hashmap

26

27

Num\_indices[num] = i

28

29

# If no solution is found, return an empty list (this should not happen based on the problem statement)

30

31

Return []

32

33

Nums1 = [2, 7, 11, 15]

34

35

Target1 = 9

36

37

Sol = Solution()

38

39

Print(sol.twoSum(nums1, target1))

40

41

Nums2 = [3, 2, 4]

42

43

Target2 = 6

44

45

Print(sol.twoSum(nums2, target2))

46

47

Nums3 = [3, 3]

48

49

Target3 = 6

50

51

Print(sol.twoSum(nums3, target3))

52

Custom Testcase( Contribute )

Run Code: Finished

×

Run Code Result:

Your input

[2,7,11,15]

9

Your stdout

[0, 1]

[1, 2]

[0, 1]

Your answer

[0,1]

Expected answer

[0,1]