

AfterShip Software Engineer > chungchi300@hotmail.com

Email: chungchi300@hotmail.com

Test Name: AfterShip Software Engineer

Taken On: 19 Apr 2018 14:56:20 HKT

Time Taken: 59 min 55 sec/ 60 min

Invited by: Teddy

Invited on: 18 Apr 2018 15:19:40 HKT

Tags Score: Apr 5/5

Algorithms 174/185



scored in **AfterShip Software Engineer** in
59 min 55 sec on 19
Apr 2018 14:56:20 HKT

Arrays 119/125 Binary Search 69/75 Brute Force 50/50 Classic 119/125 Complexity 0/5 Core CS 5/10 Core Skills 169/175 Data Structures 124/130 Easy 125/130 Lists 5/5 Medium 69/75 Problem Solving 169/175 REST API 10/10 Rest 5/5 Role Based 20/20 Sets 5/5 Stacks 50/50

Candidate Feedback:

Show input for pass that don't pass

Recruiter/Team Comments:

No Comments.

	Question Description	Time Taken	Score	Status
Q1	Braces > Coding	7 min 37 sec	50/ 50	\odot
Q2	Count Duplicates > Coding	23 min 8 sec	50/ 50	\odot
Q3	Distinct Pairs > Coding	20 min	69/ 75	\odot
Q4	Complexity of the Code Snippet > Multiple Choice	55 sec	0/5	\otimes
Q5	Properties of Data Structures > Multiple Choice	55 sec	5/ 5	\odot
Q6	HTTP Methods: Partially Modifying Resources > Multiple Choice	20 sec	5/ 5	\odot
Q7	HTTP Verbs > Multiple Choice	29 sec	5/ 5	\odot
Q8	REST: HTTP Status Codes > Multiple Choice	1 min 11 sec	5/ 5	\odot
Q9	APIs: Components Functionality Testing > Multiple Choice	1 sec	5/ 5	\odot



QUESTION DESCRIPTION

Score 50

You are designing a compiler for a C++ program and need to check that braces in any given file are balanced.

Braces in a string are considered to be balanced if the following criteria are met:

- All braces must be closed. Braces come in pairs of the form (), {} and []. The left brace opens the pair, and the right one closes it.
- In any set of nested braces, the braces between any pair must be closed.

For example, [{}] is a valid grouping of braces but [}]{} is not.

Function Description

Complete the function *braces* in the editor below. The function must return an array of strings where the string at each index *i* denotes whether or not the braces were balanced in a *valuesi*. The array should consist of strings "YES" or "NO" aligned with their indexes in *values*.

braces has the following parameter(s):

 $values[values_0,...values_{n-1}]$: an array of strings to analyze

Constraints

- $1 \le n \le 15$
- $1 \le \text{length of } values_i \le 100$
- It is guaranteed that each values; consists of (,), {, }, [, and] only.

▶ Input Format For Custom Testing

▼ Sample Case 0

Sample Input For Custom Testing

```
2
{}[]()
{[}]}
```

Sample Output

```
YES
NO
```

Explanation

 $values_0$: {}[]() meets the criteria for a balanced string, so index 0 in our return array should contain the string YES.

 $values_1$: {[}]} contains unmatched braces between a matched pair in the substrings [}, {[}, and [}], so index 1 in our return array should contain the string NO.

CANDIDATE ANSWER

$\label{language used: JavaScript (Node.js)} Language \ used: \textbf{JavaScript (Node.js)}$

```
1 class Stack {
    constructor() {
      this.elements = [];
 4 }
 5 push(element) {
 6
      this.elements.push(element);
 8
    pop(element) {
 9
      return this.elements.pop(element);
 10 }
 11 peek() {
12
      return this.elements[this.elements.length - 1];
13
     isEmpty() {
14
15
      return this.elements.length == 0 ? true : false;
16
17 }
18 function isBalanced(previousChar, currentChar) {
if (previousChar == '{' && currentChar == '}') {
```

```
ii (previouseilui -- į aa curreileeliui -- ) / į
20
     return true:
21
if (previousChar == '[' && currentChar == ']') {
23
24
    if (previousChar == '(' && currentChar == ')') {
25
26
27
28
29
    return false;
30 }
31 function isValueBalanced(value) {
32 const stack = new Stack();
    const chars = value.split(");
34
for (let i = 0; i < chars.length; i++) {
36
     if (chars[i] == '{' || chars[i] == '[' || chars[i] == '(') {
37
      stack.push(chars[i]);
38
     } else {
39
       let previousChar = stack.pop();
40
41
       if (!isBalanced(previousChar, chars[i])) {
42
        return false;
43
44
     }
45 }
46 //still have unmatched element in stack element
47 if (!stack.isEmpty()) {
48
     return false;
49
    }
    return true;
51 }
52 /*
* Complete the function below.
54 */
55 function braces(values) {
56
     return values.map(expression=>isValueBalanced(expression)?'YES':'NO',values);
57 }
58
 TESTCASE TYPE STATUS SCORE TIME TAKEN MEMORY USED
                                     0.06 sec

1 0.06 sec

1 0.06 sec

9 0.06 sec

9 0.05 sec

9 0.05 sec
 TestCase 0Easy② Success10.06 secTestCase 1Easy② Success10.06 sec
                                                                 30.1 MB
                                                                 28 MB
 TestCase 2 Easy ⊘ Success
                                                                 30 MB
                Easy Success
 TestCase 3
                                                                 28.1 MB
                Easy Success
Easy Success
 Testcase 4
                                                                  28.1 MB
 Testcase 5
                                                                  30.3 MB
                                     10 0.06 sec
                Easy Success
 Testcase 6
                                                                 28.1 MB
                Easy Success
                                      10
 Testcase 7
                                                 0.06 sec
                                                                  30.1 MB
No Comments
```





Score 50

Problem Solving

QUESTION DESCRIPTION

Given an array of integers, your task is to count the number of duplicate array elements. Duplicate is defined as two or more identical elements. For example, in the array [1, 2, 2, 3, 3, 3], the two twos are one duplicate and so are the three threes.

Function Description

Complete the function countDuplicates in the editor below. The function must return an integer denoting the number of non-unique (duplicate) values in the numbers array.

countDuplicates has the following parameter(s):

numbers[numbers0,...numbersn-1]: an array of integers to process

CONSTRAINTS

- $1 \le n \le 1000$
- $1 \le numbers_i \le 1000, 0 \le i < n$

▶ Input Format Format for Custom Testing

▼ Sample Case 0

Sample Input

```
8
1
3
1
4
5
6
3
2
```

Sample Output

2

Explanation

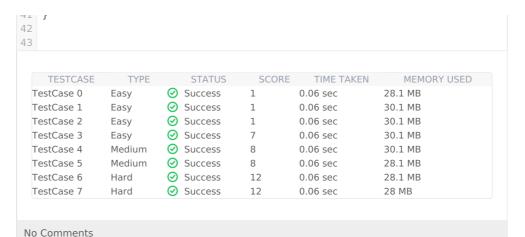
n=8 and numbers=[1, 3, 1, 4, 5, 6, 3, 2]. The integers 1 and 3 both occur more than once, so we return 2 as our answer.

▶ Sample Case 1

CANDIDATE ANSWER

Language used: JavaScript (Node.js)

```
1 /*
 2 * Complete the countDuplicates function below.
 3 */
 4 function countDuplicates(numbers) {
 5
 6
     false for existence
 7
     true for duplication
 8
        numberMap {
 9
         1:true,
          3:true,
11 4:false,
12 5:false,
13 6:false,
14 2:false
15
16
17
      function getNumberMap(numbers){
        let numberMap = {};
18
19
         for(let number of numbers) {
           if(number in numberMap){
             numberMap[number] = true;
           }else{
23
             numberMap[number] = false;
24
25
         }
26
         //console.log('numberMap',numberMap);
27
         return numberMap;
28
29
      function getDuplicate(numberMap){
30
         let duplicateNo = 0;
31
         for(let duplicate of Object.values(numberMap)){
32
           if(duplicate){
33
             duplicateNo++;
34
           }
35
         }
36
         return duplicateNo;
37
      }
38
      let numberMap = getNumberMap(numbers);
39
40
      return getDuplicate(numberMap);
/11 l
```







Score 69



QUESTION DESCRIPTION

In this challenge, you will be given an array of integers and a target value. Determine the number of *distinct* pairs of elements in the array that sum to the target value. Two pairs (a, b) and (c, d) are considered to be distinct if and only if the values in sorted order do not match, i.e., (1, 9) and (9, 1) are indistinct but (1, 9) and (9, 2) are distinct.

For instance, given the array [1, 2, 3, 6, 7, 8, 9, 1], and a target value of 10, the seven pairs (1,9), (2,8), (3,7), (8,2), (9,1), (9,1), and (1,9) all sum to 10 and only three distinct pairs: (1,9), (2,8), and (3,7).

Function Description

Complete the function *numberOfPairs* in the editor below. The function must return an integer, the total number of *distinct* pairs of elements in the array that sum to the target value.

numberOfPairs has the following parameter(s):

 $a[a_0,...a_{n-1}]$: an array of integers to select pairs from k: target integer value to sum to

Constraints

- $1 \le n \le 5 \times 10^5$
- $0 \le a_i \le 10^9$
- $0 \le k \le 5 \times 10^9$

▶ Input Format for Custom Testing

▼ Sample Case 0

Sample Input 0

Sample Output 0

1

Explanation 0

a = [1, 3, 46, 1, 3, 9], k = 47

There are 4 pairs of unique elements where $a_i + a_i = k$:

1 $(a_0 = 1 \ a_2 = 46)$

```
2. (a_2 = 46, a_0 = 1)

3. (a_2 = 46, a_3 = 1)

4. (a_3 = 1, a_2 = 46)

In the list above, all four pairs contain the same values. We only have 1 distinct pair, (1, 46).
```

▶ Sample Case 1

CANDIDATE ANSWER

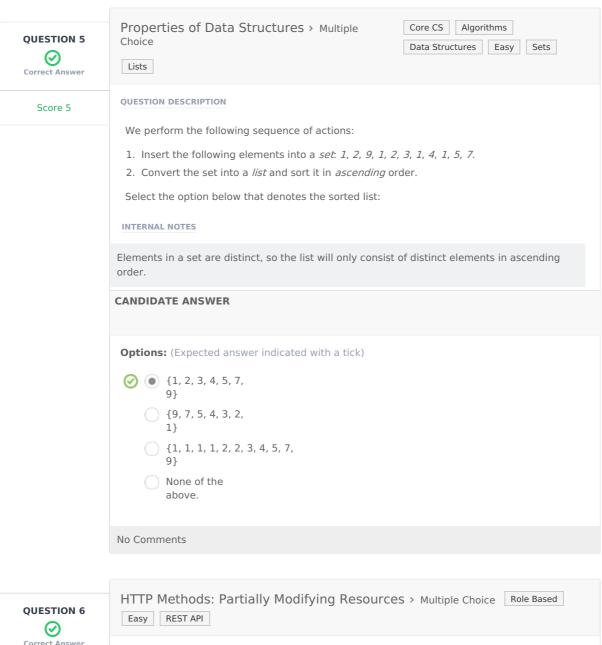
Language used: JavaScript (Node.js)

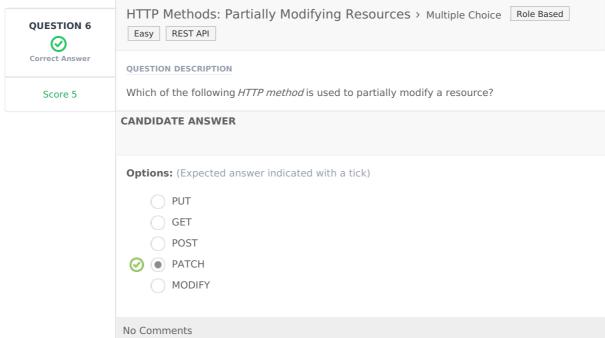
```
1 /*
 2 * Complete the numberOfPairs function below.
 3 */
 4 function numberOfPairs(arr, k) {
 5
 6
      * Write your code here.
 7
      */
 8
      //
 9
11
      1 =  enter =  {1:1}
      46 = > complement 1,
13
14
15
      function getPairs(arr){
16
        var unMatchedNum = {};
17
         var pairs = [];
18
        for(let num of arr) {
19
           const complement = k - num;
20
           if(complement in unMatchedNum){
             pairs.push([num,complement]);
22
23
          }else{
24
             unMatchedNum[num] = num;
25
           }
26
        }
27
        return pairs;
28
29
      function isPairEqual(pair1,pair2){
30
        // if(pair1.every)
31
32
      function filterDistinct(pairs){
33
        //isPairEqual to remove
34
        return pairs;
35
36
      let pairs = getPairs(arr);
      pairs = filterDistinct(pairs);
37
38
      return pairs.length;
39 }
40
41
```

TESTCASE	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
TestCase 0	Easy (Success	1	0.06 sec	30.1 MB
TestCase 1	Easy (Success	1	0.06 sec	30 MB
TestCase 2	Easy (Success	1	0.06 sec	30.2 MB
TestCase 3	Easy (Success	2	0.06 sec	28.1 MB
TestCase 4	Easy (Wrong Answer	0	0.06 sec	28.1 MB
TestCase 5	Easy (Wrong Answer	0	0.06 sec	28.1 MB
TestCase 6	Easy (🛭 Wrong Answer	0	0.05 sec	28.1 MB
TestCase 7	Easy (Success	2	0.06 sec	28 MB
TestCase 8	Easy (Success	4	0.06 sec	30 MB
TestCase 9	Easy (Success	4	0.06 sec	28 MB
TestCase 10	Easy (Success	5	0.07 sec	30 MB
TestCase 11	Easy (Success	5	0.07 sec	30.3 MB
TestCase 12	Easy (Success	6	0.06 sec	28 MB
TestCase 14	Easy (Success	19	0.18 sec	53.5 MB
TestCase 16	Easy (Success	19	0.36 sec	89.3 MB

No Comments

QUESTION 4	Complexity of the Code Snippet > Multiple Choice
Wrong Answer	QUESTION DESCRIPTION
Score 0	Consider the following code snippet:
	<pre>int a = 1; while (a < n) { a = a * 2; }</pre>
	What is the complexity of the above code snippet?
	INTERNAL NOTES
	Task is reducing exponentially by an order of 2.
	CANDIDATE ANSWER
	Options: (Expected answer indicated with a tick)
	O(n)O(1)
	 O(log₂(n)) O(2ⁿ)
	No Comments





QUESTION 7	HTTP Verbs > Multiple Choice Rest Role Based Easy		
Correct Answer	QUESTION DESCRIPTION Identify any item(s) in the list below that are not HTTP verbs:		
Score 5	CANDIDATE ANSWER		
	Options: (Expected answer indicated with a tick) GET POST HEAD DELETE REMOVE PUT PATCH		
	No Comments		
QUESTION 8	REST: HTTP Status Codes > Multiple		
Correct Answer	QUESTION DESCRIPTION		
Score 5	Which HTTP response code describing the attempt of accessing restricted resources?		
	CANDIDATE ANSWER		
	Options: (Expected answer indicated with a tick)		

None of these

No Comments

QUESTION 9	APIs: Components Functionality Testing > Multiple Choice API Role Based Easy
Correct Answer	QUESTION DESCRIPTION
Score 5	Which testing is used for a program's individual components functionality testing?
	CANDIDATE ANSWER
	Options: (Expected answer indicated with a tick)
	Functionality Testing
	✓ ● Unit Testing
	Security Testing
	Smoke Testing
	Regression Testing
	No Comments

PDF generated at: 19 Apr 2018 07:57:54 UTC