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:	API 5/5 Algorithms 174/185 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

94.6
194/205

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

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	✓
Q2	**Count Duplicates** › **Coding**	23 min 8 sec	50/ 50	✓
Q3	**Distinct Pairs** › **Coding**	20 min	69/ 75	✓
Q4	**Complexity of the Code Snippet** › **Multiple Choice**	55 sec	0/ 5	✗
Q5	**Properties of Data Structures** › **Multiple Choice**	55 sec	5/ 5	✓
Q6	**HTTP Methods: Partially Modifying Resources** › **Multiple Choice**	20 sec	5/ 5	✓
Q7	**HTTP Verbs** › **Multiple Choice**	29 sec	5/ 5	✓
Q8	**REST: HTTP Status Codes** › **Multiple Choice**	1 min 11 sec	5/ 5	✓
Q9	**APIs: Components Functionality Testing** › **Multiple Choice**	1 sec	5/ 5	✓

QUESTION 1

✓ Correct Answer

Braces › Coding Easy Data Structures Classic Algorithms Stacks Core Skills Problem Solving

Score 50

QUESTION DESCRIPTION

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 $values_i$. 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

- $1 \leq n \leq 15$
- 1 ≤ length of $values_i \leq 100$
- It is guaranteed that each $values_i$ 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

Language used: **JavaScript (Node.js)**

```
1  class Stack {
2    constructor() {
3      this.elements = [];
4    }
5    push(element) {
6      this.elements.push(element);
7    }
8    pop(element) {
9      return this.elements.pop(element);
10   }
11   peek() {
12     return this.elements[this.elements.length - 1];
13   }
14   isEmpty() {
15     return this.elements.length == 0 ? true : false;
16   }
17 }
18 function isBalanced(previousChar, currentChar) {
19   if (previousChar == '{' && currentChar == '}') {
```

```
    return true;
  }
  if (previousChar == '[' && currentChar == ']') {
    return true;
  }
  if (previousChar == '(' && currentChar == ')') {
    return true;
  }

  return false;
}
function isValueBalanced(value) {
  const stack = new Stack();
  const chars = value.split('');

  for (let i = 0; i < chars.length; i++) {
    if (chars[i] == '{' || chars[i] == '[' || chars[i] == '(') {
      stack.push(chars[i]);
    } else {
      let previousChar = stack.pop();

      if (!isBalanced(previousChar, chars[i])) {
        return false;
      }
    }
  }
  //still have unmatched element in stack element
  if (!stack.isEmpty()) {
    return false;
  }
  return true;
}
/*
 * Complete the function below.
 */
function braces(values) {
    return values.map(expression=>isValueBalanced(expression)?'YES':'NO',values);
}

```

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	28 MB
TestCase 2	Easy	Success	1	0.06 sec	30 MB
TestCase 3	Easy	Success	9	0.06 sec	28.1 MB
Testcase 4	Easy	Success	9	0.05 sec	28.1 MB
Testcase 5	Easy	Success	9	0.06 sec	30.3 MB
Testcase 6	Easy	Success	10	0.06 sec	28.1 MB
Testcase 7	Easy	Success	10	0.06 sec	30.1 MB

No Comments

QUESTION 2

Correct Answer

Score 50

Count Duplicates › Coding

Easy | Brute Force | Algorithms | Arrays | Core Skills | 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[numbers$_0$,...numbers$_{n-1}$]: an array of integers to process

Constraints

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)**

```
/*
 * Complete the countDuplicates function below.
 */
function countDuplicates(numbers) {
    /*
    false for existence
    true for duplication
        numberMap {
          1:true,
          3:true,
4:false,
5:false,
6:false,
2:false
        }
    */
    function getNumberMap(numbers){
        let numberMap = {};
        for(let number of numbers){
            if(number in numberMap){
                numberMap[number] = true;
            }else{
                numberMap[number] = false;
            }
        }
        //console.log('numberMap',numberMap);
        return numberMap;
    }
    function getDuplicate(numberMap){
        let duplicateNo = 0;
        for(let duplicate of Object.values(numberMap)){
            if(duplicate){
                duplicateNo++;
            }
        }
        return duplicateNo;
    }
    let numberMap = getNumberMap(numbers);

    return getDuplicate(numberMap);
}
```

```
42
43
```

TESTCASE	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
TestCase 0	Easy	Success	1	0.06 sec	28.1 MB
TestCase 1	Easy	Success	1	0.06 sec	30.1 MB
TestCase 2	Easy	Success	1	0.06 sec	30.1 MB
TestCase 3	Easy	Success	7	0.06 sec	30.1 MB
TestCase 4	Medium	Success	8	0.06 sec	30.1 MB
TestCase 5	Medium	Success	8	0.06 sec	28.1 MB
TestCase 6	Hard	Success	12	0.06 sec	28.1 MB
TestCase 7	Hard	Success	12	0.06 sec	28 MB

No Comments

QUESTION 3

Correct Answer

Score 69

Distinct Pairs › Coding

Binary Search | Data Structures | Medium | Algorithms | Arrays | Classic | Core Skills | Problem Solving

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

```
6
1
3
46
1
3
9
47
```

Sample Output 0

```
1
```

Explanation 0

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

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

1. *(a_0 = 1, a_3 = 46)*

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)**

```
/*
 * Complete the numberOfPairs function below.
 */
function numberOfPairs(arr, k) {
    /*
     * Write your code here.
     */
    //

    /*
    1 => enter => {1:1}
    46 => complement  1,

    */
    function getPairs(arr){
        var unMatchedNum = {};
         var pairs = [];
        for(let num of arr){
            const complement = k - num;
            if(complement in unMatchedNum){

                pairs.push([num,complement]);
            }else{
                unMatchedNum[num] = num;
            }
        }
        return pairs;
    }
    function isPairEqual(pair1,pair2){
        // if(pair1.every)
    }
    function filterDistinct(pairs){
        //isPairEqual to remove
        return pairs;
    }
    let pairs = getPairs(arr);
    pairs = filterDistinct(pairs);
    return pairs.length;
}


```

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

Wrong Answer

Score 0

Complexity of the Code Snippet › Multiple Choice

Core CS | Algorithms | Complexity | Easy

QUESTION DESCRIPTION

Consider the following code snippet:

```
int a = 1;

while (a < n) {
   a = a * 2;
}
```

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)
- ✓ $O(\log_2(n))$
- $O(2^n)$

No Comments

QUESTION 5

Correct Answer

Score 5

Properties of Data Structures › Multiple Choice

Core CS | Algorithms | Data Structures | Easy | Sets | Lists

QUESTION DESCRIPTION

We perform the following sequence of actions:

1. Insert the following elements into a *set*: *1*, *2*, *9*, *1*, *2*, *3*, *1*, *4*, *1*, *5*, *7*.
2. Convert the set into a *list* and sort it in *ascending* order.

Select the option below that denotes the sorted list:

INTERNAL NOTES

Elements in a set are distinct, so the list will only consist of distinct elements in ascending order.

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

- [x] {1, 2, 3, 4, 5, 7, 9}
- [ ] {9, 7, 5, 4, 3, 2, 1}
- [ ] {1, 1, 1, 1, 2, 2, 3, 4, 5, 7, 9}
- [ ] None of the above.

No Comments

QUESTION 6

Correct Answer

Score 5

HTTP Methods: Partially Modifying Resources › Multiple Choice

Role Based | Easy | REST API

QUESTION DESCRIPTION

Which of the following *HTTP method* is used to partially modify a resource?

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

- [ ] PUT
- [ ] GET
- [ ] POST
- [x] PATCH
- [ ] MODIFY

No Comments

QUESTION 7

Correct Answer

Score 5

HTTP Verbs › Multiple Choice Rest Role Based Easy

QUESTION DESCRIPTION

Identify any item(s) in the list below that are *not* HTTP verbs:

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

- [ ] GET
- [ ] POST
- [ ] HEAD
- [ ] DELETE
- [x] REMOVE
- [ ] PUT
- [ ] PATCH

No Comments

QUESTION 8

Correct Answer

Score 5

REST: HTTP Status Codes › Multiple Choice Easy Role Based REST API

QUESTION DESCRIPTION

Which *HTTP response code* describing the attempt of accessing *restricted resources*?

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

- [ ] 200
- [ ] 401
- [ ] 402
- [x] 403
- [ ] None of these

No Comments

QUESTION 9

Correct Answer

Score 5

APIs: Components Functionality Testing › Multiple Choice

API | Role Based | Easy

QUESTION DESCRIPTION

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