

PROGRAMMER QUIZ

Kategori soal yang tidak memiliki tanda bintang (*) tidak wajib untuk diisi, tetapi jika diisi akan menjadi nilai tambah

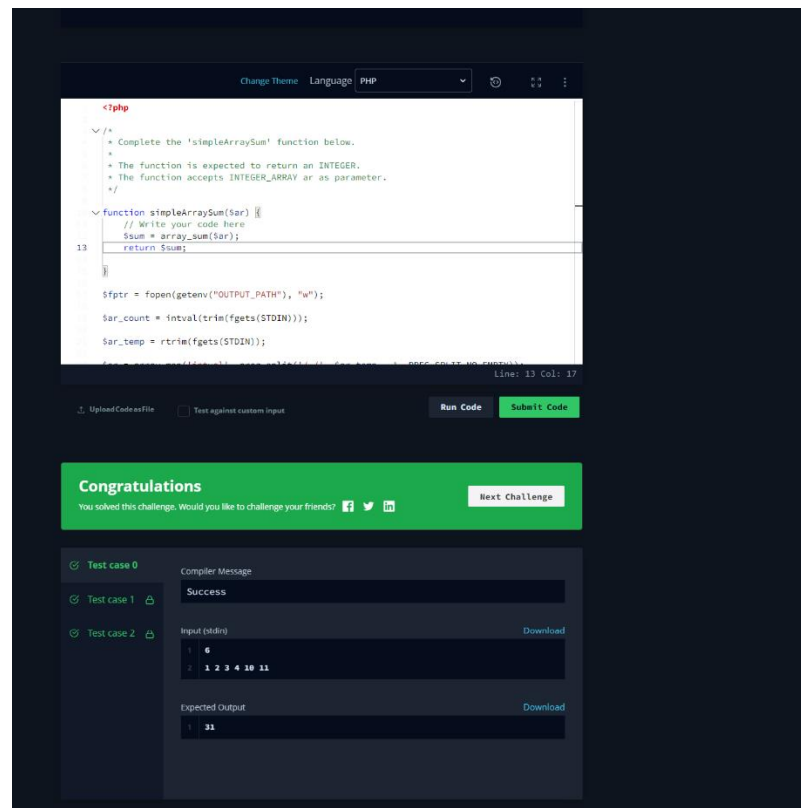
Warmup test*

1. Dibawah ini terdapat beberapa link untuk melakukan tes dasar algoritma, test ini bisa dilakukan dengan bahasa apa saja (diutamakan Node Js, Go, Python, Java, Kotlin, PHP).

a. Simple Array Sum

<https://www.hackerrank.com/challenges/simple-array-sum>

Hasil Test:



b. Compare the Triplets

<https://www.hackerrank.com/challenges/compare-the-triplets>

Hasil Test:

The screenshot shows the Hackerrank interface for the 'Compare the Triplets' challenge. At the top, there's a code editor with a C++ solution. The code defines a function `compareTriplets` that takes two integer arrays, `a` and `b`, and returns an array of two integers representing the scores. The function iterates through both arrays, comparing elements at each index. If `a[i] > b[i]`, it increments `alice_score`; if `a[i] < b[i]`, it increments `bob_score`. The final scores are returned in an array.

```
1 // Complete the 'compareTriplets' function below.
2 //
3 // The function is expected to return an INTEGER_ARRAY.
4 // The function accepts following parameters:
5 // 1. INTEGER_ARRAY a
6 // 2. INTEGER_ARRAY b
7 //
8
9 function compareTriplets(a, b) {
10     let alice_score = 0;
11     let bob_score = 0;
12     for (let i = 0; i < a.length; i++) {
13         if (a[i] > b[i]) {
14             alice_score++;
15         } else if (a[i] < b[i]) {
16             bob_score++;
17         }
18     }
19     return [alice_score, bob_score];
20 }
```

Below the code editor, there are buttons for 'Run Code' and 'Submit Code'. A message indicates that the user has earned 10.00 points and is 10 points away from the 1st star badge. A green 'Congratulations' banner says 'You solved this challenge. Would you like to challenge your friends?' with a 'Next Challenge' button.

On the left, there's a list of test cases (0 to 6). On the right, the 'Compiler Message' section shows 'Success'. Below that, the 'Input (stdin)' is displayed as:

```
1 5 6 7
2 3 6 10
```

And the 'Expected Output' is:

```
1 1 1
```

At the bottom, there's a footer with links: Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy.

c. A Very Big Sum

<https://www.hackerrank.com/challenges/a-very-big-sum>

Hasil Test:

The screenshot displays the HackerRank interface for the 'A Very Big Sum' challenge. At the top, the code editor shows a JavaScript solution for the 'aVeryBigSum' function. The function takes an array 'ar' and returns the sum of its elements using a for loop. Below the code editor, there are buttons for 'Run Code' and 'Submit Code'. A notification banner indicates that the user has earned 10.00 points and is 70 points away from the 2nd star badge. A green 'Congratulations' banner follows, stating 'You solved this challenge. Would you like to challenge your friends?' with social media icons and a 'Next Challenge' button. Below this, the 'Test case 0' results are shown, including the input (5 and a large array of numbers) and the expected output (5000000015). The footer contains links for Blog, Scoring, Environment, FAQ, About Us, Support, Careers, Terms Of Service, and Privacy Policy.

```
function readline() {  
    return inputString[currentLine++];  
}  
  
/*  
 * Complete the 'aVeryBigSum' function below.  
 * The function is expected to return a LONG_INTEGER.  
 * The function accepts LONG_INTEGER_ARRAY ar as parameter.  
 */  
  
function aVeryBigSum(ar) {  
    // Write your code here  
    let sum = 0;  
    for (let i = 0; i < ar.length; i++) {  
        sum += ar[i];  
    }  
    return sum;  
}
```

You have earned 10.00 points!
You are now 70 points away from the 2nd star for your problem solving badge.

Congratulations
You solved this challenge. Would you like to challenge your friends? [f](#) [t](#) [in](#) [Next Challenge](#)

Test case 0
Compiler Message: Success
Input (stdin):
1 5
2 1000000001 1000000002 1000000003 1000000004 1000000005
Expected Output:
1 5000000015

[Blog](#) | [Scoring](#) | [Environment](#) | [FAQ](#) | [About Us](#) | [Support](#) | [Careers](#) | [Terms Of Service](#) | [Privacy Policy](#)

d. StairCase

<https://www.hackerrank.com/challenges/staircase>

Hasil Test:

```
//  
function readLine() {  
    return inputString[currentLine++];  
}  
  
/*  
 * Complete the 'staircase' function below.  
 * The function accepts INTEGER n as parameter.  
 */  
  
function staircase(n) {  
    // Write your code here  
    for (let i = 1; i <= n; i++) {  
        const spaces = ' '.repeat(n - i);  
        const hashes = '#'.repeat(i);  
        console.log(spaces + hashes);  
    }  
}  
  
function main() {  
    const n = parseInt(readLine().trim(), 10);  
}
```

Line: 34 Col: 38

Upload Code as File ☐ Test against custom input Run Code Submit Code

Test case 0 ☒
Test case 1 ☒
Test case 2 ☒
Test case 3 ☒
Test case 4 ☒
Test case 5 ☒
Test case 6 ☒

Compiler Message
Success

Hidden Test Case
Unlock this testcase for 5 hacks.
Unlock

[Blog](#) | [Scoring](#) | [Environment](#) | [FAQ](#) | [About Us](#) | [Support](#) | [Careers](#) | [Terms Of Service](#) | [Privacy Policy](#)

Design pattern & System Optimization

1. Aplikasi modern dituntut untuk banyak melakukan banyak task dalam suatu request, sebagai contoh ketika user melakukan register pada suatu aplikasi maka aplikasi tersebut harus mengirim email konfirmasi kepada user, atau memasukan user ke mailing list.

Kegiatan diatas bisa menyebabkan sistem membutuhkan waktu yang lebih untuk memproses semua task tersebut, sedangkan user tidak bisa menunggu terlalu lama sampai task tersebut selesai dilakukan.

Bagaimana solusi yang anda tawarkan untuk menangani hal tersebut ?

Jawaban :

Untuk permasalahan tersebut menurut saya bisa diatasi dengan pemanfaatan Asynchronous Processing dimana kita bisa menerapkan tugas - tugas tersebut secara asinkron. Kita juga bisa memanfaatkan microservices agar setiap tugas dapat dijalankan menjadi layanan terpisah. Sehingga jikapun terjadi permasalahan pada 1 layanan, layanan lainnya masih dapat berjalan dengan baik. Kita juga bisa memanfaatkan caching untuk menyimpan data secara sementara waktu sehingga tidak perlu berulang kali mengambil data pada suatu halaman.

Code Knowledge & Optimization

1. Apa fungsi dan kegunaan dari code refactoring ?

Jawaban:

Fungsi utama code refactoring adalah untuk meningkatkan kualitas kode dimana refactoring dapat mengurangi kompleksitas dan mengurangi duplikasi kode.

2. Apakah komentar dalam kode berguna ?, sebagian orang menyarankan komentar dalam kode sebaiknya dihindari. Apakah anda setuju dengan hal tersebut ?, jelaskan alasannya.

Jawaban :

Menurut saya penggunaan komen dalam kode adalah hal yang baik. Karena beberapa kali saya terbantu untuk memahami kode dari rekan saya berdasarkan dari komen yang tersedia di kode tersebut.

3. What makes good code good?

Answer :

Good code are code that can be not only readable by the computer but by other programmer too. And good code must have the efficiency and maintainability too.

Web development

1. Jelaskan tentang API versioning dan kegunaannya, dan bagaimana anda akan manage API versioning pada Web Services ?

Database

1. How would you explain the recent rise in interest in NoSQL?

Code collaboration & Team management

1. How would you deal with legacy code?
2. What is the biggest difference between Agile and Waterfall?

Software development knowledge*

1. Why is writing software difficult? What makes maintaining software hard?

Answer:

Writing software can be difficult because it's based on requirement from client that can be complex and sometimes need to changing requirements that have been negotiated before. The use of technology stack also need to be done carefully so it's not affect the software performance. Beside of that, maintaining software also can be so hard. Because when maintaining software, we have to make sure that the software we create is still compatible with current technological developments, whether in the form of the technology stack used or third parties.