Problem 2723: Add Two Promises

Problem Information

Difficulty: Easy
Acceptance Rate: 0.00%
Paid Only: No

Problem Description

Given two promises

promise1

and

promise2

, return a new promise.

promise1

and

promise2

will both resolve with a number. The returned promise should resolve with the sum of the two numbers.

Example 1:

Input:

promise1 = new Promise(resolve => setTimeout(() => resolve(2), 20)), promise2 = new Promise(resolve => setTimeout(() => resolve(5), 60))

Output:

7

Explanation:

The two input promises resolve with the values of 2 and 5 respectively. The returned promise should resolve with a value of 2 + 5 = 7. The time the returned promise resolves is not judged for this problem.

Example 2:

Input:

promise1 = new Promise(resolve => setTimeout(() => resolve(10), 50)), promise2 = new Promise(resolve => setTimeout(() => resolve(-12), 30))

Output:

-2

Explanation:

The two input promises resolve with the values of 10 and -12 respectively. The returned promise should resolve with a value of 10 + -12 = -2.

Constraints:

promise1

and

promise2

are promises that resolve with a number

Code Snippets

JavaScript:

```
/**
* @param {Promise} promise1
* @param {Promise} promise2
* @return {Promise}
*/
var addTwoPromises = async function(promise1, promise2) {

};

/**
* addTwoPromises(Promise.resolve(2), Promise.resolve(2))
* .then(console.log); // 4
*/
```

TypeScript:

```
type P = Promise<number>

async function addTwoPromises(promise1: P, promise2: P): P {

};

/**
* addTwoPromises(Promise.resolve(2), Promise.resolve(2))
* .then(console.log); // 4
*/
```

Solutions

JavaScript Solution:

```
/**
* Problem: Add Two Promises
* Difficulty: Easy
* Tags: general
*
* Approach: Optimized algorithm based on problem constraints
* Time Complexity: O(n) to O(n^2) depending on approach
* Space Complexity: O(1) to O(n) depending on approach
```

```
*/

/**
* @param {Promise} promise1
* @param {Promise} promise2
* @return {Promise}
*/
var addTwoPromises = async function(promise1, promise2) {

};

/**
* addTwoPromises(Promise.resolve(2), Promise.resolve(2))
* .then(console.log); // 4
*/
```

TypeScript Solution:

```
/**
* Problem: Add Two Promises
* Difficulty: Easy
* Tags: general
*
* Approach: Optimized algorithm based on problem constraints
* Time Complexity: O(n) to O(n^2) depending on approach
* Space Complexity: O(1) to O(n) depending on approach
*/

type P = Promise<number>

async function addTwoPromises(promise1: P, promise2: P): P {

};

/**
* addTwoPromises(Promise.resolve(2), Promise.resolve(2))
* .then(console.log); // 4
*/
```