

Контролно №1 / Вариант С

Здравейте, контролното ще се проведе по следният начин:

- 1) В .zip файлът ще намерите първоначалният код, който трябва да довършите.
- 2) След като сте готови със задачите, моля, прикачете архивирания файл в системата.
- 3) **ВАЖНО Е ДА ИЗТРИЕТЕ папка node_modules (ако има).**

1. Площ и периметър

Напишете JS функция, която изчислява площта и периметъра на правоъгълник по дадени две страни.

Входът идва като масив от низове, които трябва да бъдат анализирани като числа.

Изходът трябва да бъде отпечатан на конзолата на два реда.

Примери

Input	Output
[2, 2]	4 8
[1, 3]	3 8
[2.5, 3.14]	7.85 11.28

2. Hashtag Generator

Напишете функция, в нов JS файл, наречена `generateHashtag`, която приема низ като вход и генерира хаштаг съгласно следните правила:

1. Генеририаният хаштаг трябва да започва с `#`.
2. Всички думи във входния низ трябва да започват с главна буква.
3. Пространствата между думите трябва да бъдат премахнати.
4. Ако генеририаният хаштаг е по-дълъг от 140 символа или ако входният низ е празен (или резултатът след обработка е празен), функцията трябва да върне `false`.

Function Signature
<pre>/** * Generates a hashtag from the input string. * @param {string} str - The input string. * @returns {string boolean} - The generated hashtag string or false. */ function generateHashtag(str: string): string boolean;</pre>

Примери

- ⇒ generateHashtag("JavaScript is awesome"); // "#JavaScriptIsAwesome"
- ⇒ generateHashtag("hello world"); // "#HelloWorld"
- ⇒ generateHashtag("This is a very very very very very very very very
very very very long input that should result in a false hashtag because
it exceeds the character limit of 140"); // false
- ⇒ generateHashtag(""); // false

3. Quick Sort Algorithm

Вие Имплементирайте функция, наречена **quickSort**, която приема масив от числа като аргумент и връща сортиран масив, използвайки алгоритъма за бързо сортиране.

Function Signature

```
/**  
 * Sorts an array using the Quick Sort algorithm.  
 * @param {number[]} array - The array to be sorted.  
 * @returns {number[]} - The sorted array.  
 */  
function quickSort(array: number[]): number[]
```

Примери

- ⇒ console.log(quickSort([5, 4, 3, 2, 1]));
// Output: [1, 2, 3, 4, 5]
- ⇒ console.log(quickSort([64, 34, 25, 12, 22, 11, 90]));
// Output: [11, 12, 22, 25, 34, 64, 90]

В нов файл **quickSort.txt**, опиши какво представлява алгоритъма за бързо сортиране и каква е средна времева сложност.