BACK

Array Conversion

136,304

DESCRIPTION | MY SOLUTIONS | LEADERBOARD | COMMENTS | README

CODEWRITING SCORE: 300/300

Given an array of 2^k integers (for some integer k), perform the following operations until the array contains only one element:

- On the 1^{st}, 3^{rd}, 5^{th}, etc. iterations (1-based) replace each pair of consecutive elements with their sum;
- On the 2^{nd}, 4^{th}, 6^{th}, etc. iterations replace each pair of consecutive elements with their product.

After the algorithm has finished, there will be a single element left in the array. Return that element.

Example

For `inputArray = [1, 2, 3, 4, 5, 6, 7, 8]`, the output should be
`arrayConversion(inputArray) = 186`.

We have `[1, 2, 3, 4, 5, 6, 7, 8] -> [3, 7, 11, 15] -> [21, 165] -> [186]`, so the answer is `186`.

Input/Output

- **[execution time limit] 4 seconds (js)**
- **[input] array.integer inputArray**

  Guaranteed constraints:

  `1 ≤ inputArray.length ≤ 20`,

  `-9 ≤ inputArray[i] ≤ 99`.

- **[output] integer**

[JavaScript (ES6)] Syntax Tips

```
// Prints help message to the console
// Returns a string
function helloWorld(name) {
    console.log("This prints to the console when you Run Tests");
    return "Hello, " + name;
}
```