

Problem 2797: Partial Function with Placeholders

Problem Information

Difficulty: Easy

Acceptance Rate: 89.50%

Paid Only: Yes

Problem Description

Given a function `fn` and an array `args`, return a function `partialFn`.

Placeholders `"_` in the `args` should be replaced with values from `restArgs` starting from index `0`. Any remaining values in the `restArgs` should be added at the end of the `args`.

`partialFn` should return a result of `fn`. `fn` should be called with the elements of the modified `args` passed as separate arguments.

Example 1:

Input: fn = (...args) => args, args = [2,4,6], restArgs = [8,10] **Output:** [2,4,6,8,10]

Explanation: const partialFn = partial(fn, args) const result = partialFn(...restArgs) console.log(result) // [2,4,6,8,10] There are no placeholders "_" in args therefore restArgs is just added at the end of args. Then the elements of the args are passed as separate arguments to fn, which returns passed arguments as an array.

Example 2:

Input: fn = (...args) => args, args = [1,2,"_",4,"_",6], restArgs = [3,5] **Output:**

[1,2,3,4,5,6] **Explanation:** const partialFn = partial(fn, args) const result = partialFn(...restArgs) console.log(result) // [1,2,3,4,5,6] Placeholders "_" are replaced with values from the restArgs. Then the elements of the args are passed as separate arguments to fn, which returns passed arguments as an array.

Example 3:

****Input:**** fn = (a, b, c) => b + a - c, args = ["_", 5], restArgs = [5, 20] ****Output:**** -10
****Explanation:**** const partialFn = partial(fn, args) const result = partialFn(...restArgs)
console.log(result) // -10 Placeholder "_" is replaced with 5 and 20 is added at the end of args.
Then the elements of the args are passed as separate arguments to fn, which returns -10 (5 +
5 - 20).

****Constraints:****

* `fn` is a function * `args` and `restArgs` are valid JSON arrays * `1 <= args.length <= 5 * 104` * `1 <= restArgs.length <= 5 * 104` * `0 <= number of placeholders <= restArgs.length`

Code Snippets

JavaScript:

```
/**  
 * @param {Function} fn  
 * @param {Array} args  
 * @return {Function}  
 */  
var partial = function(fn, args) {  
  
    return function(...restArgs) {  
  
    }  
};
```

TypeScript:

```
type JSONValue = null | boolean | number | string | JSONValue[] | { [key: string]: JSONValue };  
type Fn = (...args: JSONValue[]) => JSONValue  
  
function partial(fn: Fn, args: JSONValue[]): Fn {  
  
    return function(...restArgs) {  
  
    }  
};
```