

1. Desarrollar un método en NODE JS que acepte 3 valores entero. La función debería devolver verdadero si se puede construir un triángulo con los lados de la longitud dada y falso en cualquier otro caso.

(En este caso, todos los triángulos deben tener una superficie mayor que 0 para ser aceptados).

2. Utilizando un array variable de strings y un número entero R. Realizar una función que devuelva la primera cadena más larga que consta de R cadenas consecutivas tomadas en el array.

Ejemplos:

variable = ["abcd", "efghij", "klmnop", "qrst", "cuenta", "vwxyz"], k = 2

Concatenando las cadenas consecutivas de variable por 2, obtenemos:

abcdefghij (longitud 10) concatenación de variable[0] y variable[1]

efghijklmnop (" 12) concatenación de variable[1] y variable[2]

klmnopqrst (" 10) concatenación de variable[2] y variable[3]

qrstcuenta (" 10) concatenación de variable[3] y variable[4]

cuenta vwxyz (" 12) concatenación de variable[4] y variable[5]

Dos cadenas son las más largas: " efghijklmnop" y " cuenta vwxyz".

El primero que se tiene es " efghijklmnop", así que el método debería devolver " efghijklmnop".

Nota

cadenas consecutivas: siguen una tras otra sin interrupción

3. Se debe crear una función en Typescript que devuelva una permutación de Josephus, tomando como parámetros un array inicial de elementos a permutar como si estuvieran en un círculo y contar cada k lugares hasta que no quede ninguno.

Consejos y notas: ayuda empezar a contar desde 1 hasta n, en lugar del rango habitual de 0 a n-1; k siempre será ≥ 1 .

Por ejemplo, con una matriz=[1,2,3,4,5,6,7] y k=3, la función debería actuar de esta manera.

$n=7, e=3 \Rightarrow$ se tienen 7 elementos y se elimina el tercero en la posición cada vez hasta que queda 1.

[1,2,3,4,5,6,7] – secuencia inicial

[1,2,4,5,6,7] \Rightarrow el 3 se elimina

[1,2,4,5,7] \Rightarrow el 6 se elimina

[1,4,5,7] \Rightarrow el 2 se elimina, etc.

En el siguiente link puede documentarse sobre la permutación Josephus:

http://en.wikipedia.org/wiki/Josephus_problem