

Advent JS

Retos y regalos de programación navideños

Ejercicio 2

The screenshot shows the Advent JS challenge interface. At the top, it says 'Reto #2: Ponemos en marcha la fábrica' with a factory icon. The score is '270 puntos' (270 points) and it's 'Completado por 7417 usuarios' (Completed by 7417 users). The challenge is marked as 'Fácil' (Easy). The code editor on the left shows the following JavaScript code:

```
1 function manufacture(gifts, materials) {
2   const availableMaterials = new Set(materials);
3   const manufacturableGifts = [];
4
5   for (const gift of gifts) {
6     let canManufacture = true;
7     for (const char of gift) {
8       if (!availableMaterials.has(char)) {
9         canManufacture = false;
10        break;
11      }
12    }
13    if (canManufacture) {
14      manufacturableGifts.push(gift);
15    }
16  }
17
18  return manufacturableGifts;
19 }
20
```

On the right, the results panel shows 'Puntos conseguidos: 270' (Points achieved: 270), '2496 ops/s (Más alto es mejor)' (2496 ops/s (Higher is better)), and 'Complejidad cognitiva: 3 (Más bajo es mejor)' (Cognitive complexity: 3 (Lower is better)). It also states '¡Todos los tests pasaron!' (All tests passed!).

Código JS:

```
function manufacture(gifts, materials) {

  const availableMaterials = new Set(materials);
  const manufacturableGifts = [];

  for (const gift of gifts) {
    let canManufacture = true;
    for (const char of gift) {
      if (!availableMaterials.has(char)) {
        canManufacture = false;
        break;
      }
    }
    if (canManufacture) {
      manufacturableGifts.push(gift);
    }
  }
}
```

```

    }
  }

  return manufacturableGifts;
}

```

Ejercicio 3

Advent JS

Reto #3: 🧝 El elfo travieso

220 puntos
Completado por 5407 usuarios

Ir al siguiente reto

Instrucciones

Resultados

Fácil

```

1 function findNaughtyStep(original, modified) {
2   if (original === modified) {
3     return '';
4   }
5
6   const lengthDiff = modified.length - original.length;
7
8   if (lengthDiff === 1) {
9     for (let i = 0; i < modified.length; i++) {
10      if (original[i] !== modified[i]) {
11        return modified[i];
12      }
13    }
14   } else if (lengthDiff === -1) {
15     for (let i = 0; i < original.length; i++) {
16      if (original[i] !== modified[i]) {
17        return original[i];
18      }
19    }
20   }
21
22   return '';
23 }
24
25

```

Enviar solución

Ctrl + Enter

Puntos conseguidos: 220

2780 ops/s (Más alto es mejor)

Complejidad cognitiva: 8 (Más bajo es mejor)

Los retos son puntuados utilizando diferentes factores como la complejidad cognitiva y la velocidad de ejecución. El número de líneas no es un factor determinante.

¡Todos los tests pasaron!

- Test #01
- Test #02
- Test #03
- Test #04
- Test #05
- Test #06
- Test #07: secret

Código JS:

```

function findNaughtyStep(original, modified) {

  if (original === modified) {
    return '';
  }

  const lengthDiff = modified.length - original.length;

  if (lengthDiff === 1) {
    for (let i = 0; i < modified.length; i++) {
      if (original[i] !== modified[i]) {
        return modified[i];
      }
    }
  }
}

```

```

    } else if (lengthDiff === -1) {
      for (let i = 0; i < original.length; i++) {
        if (original[i] !== modified[i]) {
          return original[i];
        }
      }
    }

    return '';
  }
}

```

Ejercicio 6

Advent JS

Reto #6: Los renos a prueba

260 puntos
Completado por 2568 usuarios

Ir al siguiente reto

1

function maxDistance(movements) {

2

let distancia = 0;

3

let comodines = 0;

4

for (const signo of movements) {

5

if (signo === '>') distancia++;

6

if (signo === '<') distancia--;

7

if (signo === '*') comodines++;

8

}

9

return Math.abs(distancia) + comodines;

10

}

11

12

13

14

JS TS

Instrucciones

Resultados

Fácil

Puntos conseguidos: 260

2186 ops/s (Más alto es mejor)

Complejidad cognitiva: 4 (Más bajo es mejor)

Los retos son puntuados utilizando diferentes factores como la complejidad cognitiva y la velocidad de ejecución. El número de líneas no es un factor determinante.

¡Todos los tests pasaron!

● Test #01

● Test #02

Enviar solución

Ctrl + Enter

Código JS:

```

function maxDistance(movements) {

  let distancia = 0;
  let comodines = 0;

  for (const signo of movements) {
    if (signo === '>') distancia++;
    if (signo === '<') distancia--;
    if (signo === '*') comodines++;
  }
}

```

```

    }

    return Math.abs(distancia) + comodines;
}

```

Ejercicio 10

Advent JS

Reto #10: 🎄 Crea tu propio árbol de navidad

180 puntos
Completado por 1237 usuarios

Ir al siguiente reto

Instrucciones Resultados Fácil

Puntos conseguidos: 180

1419 ops/s (Más alto es mejor)

Complejidad cognitiva: 2 (Más bajo es mejor)

Los retos son puntuados utilizando diferentes factores como la complejidad cognitiva y la velocidad de ejecución. El número de líneas no es un factor determinante.

¡Todos los tests pasaron!

Test #01

```

1 function createChristmasTree(ornaments, height) {
2   ornaments = ornaments.repeat(height**2).split('')
3
4   let tree = ''
5   for (let i = 1; i < height + 1; i++) {
6     tree += ' '.repeat(height - i) + ornaments.splice(0,i).join('') + '\n'
7   }
8
9   return `${tree}${'|'.padStart(height)}\n`
10 }

```

Enviar solución

Ctrl +

Código:

```

function createChristmasTree(ornaments, height) {
    ornaments = ornaments.repeat(height**2).split('')

    let tree = ''
    for (let i = 1; i < height + 1; i++) {
        tree += ' '.repeat(height - i) +
ornaments.splice(0,i).join(' ') + '\n'
    }

    return `${tree}${'|'.padStart(height)}\n`
}

```