



comp

UNIVERSIDADE FEDERAL DE SERGIPE

DEPARTAMENTO DE COMPUTAÇÃO

PROGRAMAÇÃO FUNCIONAL (COMP0393), PROF. HENDRIK MACEDO

UNIDADE 3 - 08/10/2024

NOME: \_\_\_\_\_

81

Obs 1: Em muitos problemas a seguir, há o uso da função pré-definida `Math.floor()`. Seu objetivo é retornar o maior número inteiro que é menor ou igual a um determinado número decimal. Em outras palavras, a função "arredonda para baixo" um número decimal, descartando a parte fracionária. Ex: `Math.floor(4.7) => 4` e `Math.floor(-3.1) => -4`.

Obs 2: Cada questão da prova vale 0.625pt

Para cada função a seguir, assinale a alternativa que melhor representa seu objetivo

<pre>const fun = (n) =&gt; {   if (n==1) return 0   else return 1+fun(Math.floor(n/2)) }</pre>	<ul style="list-style-type: none"><li><input type="radio"/> retorna a qtde de zeros de um inteiro</li><li><input checked="" type="radio"/> calcula o <math>\log_2 n</math></li><li><input type="radio"/> retorna a qtde de algarismos de um inteiro</li><li><input type="radio"/> calcula <math>\sqrt{n}</math></li></ul>
<pre>const fun = (n) =&gt; {   if (n==0) return "0"   else if (n==1) return "1"   else return fun(Math.floor(n/2)) +   (n%2).toString() }</pre>	<ul style="list-style-type: none"><li><input checked="" type="radio"/> aproxima para o maior inteiro e transforma em string</li><li><input type="radio"/> representação decimal do binário</li><li><input checked="" type="radio"/> representação binária do inteiro</li><li><input type="radio"/> aproxima para o menor inteiro e transforma em string</li></ul>
<pre>const fun = (a, b) =&gt; {   if (b == 0) return 0   else return fun(a,b-1)+a }</pre>	<ul style="list-style-type: none"><li><input checked="" type="radio"/> multiplica dois naturais</li><li><input type="radio"/> soma dois naturais</li><li><input type="radio"/> potencia <math>a^b</math></li><li><input type="radio"/> potencia <math>b^a</math></li></ul>
<pre>const fun = (n) =&gt; {   const helper = (a) =&gt; (b) =&gt; {     if (b == 1) return true     else if ((a%b) == 0) return false     else return helper(a)(b-1)   }   if (n==1) return false   else return helper(n)(Math.floor(n/2)) }</pre>	<ul style="list-style-type: none"><li><input type="radio"/> retorna o maior divisor de um número inteiro</li><li><input type="radio"/> retorna o maior fator primo</li><li><input type="radio"/> testa se um inteiro é divisível por 2</li><li><input checked="" type="radio"/> testa se um inteiro é primo</li></ul>
<pre>const fun = ([x, ...xs], [y, ...ys]) =&gt; {   if (x === undefined    y === undefined)     return [];   else return [[x, y], ...fun(xs, ys)]; }</pre>	<ul style="list-style-type: none"><li><input type="radio"/> retorna a soma dos pares das duas listas</li><li><input type="radio"/> retorna a concatenação dos elementos das duas listas</li><li><input checked="" type="radio"/> retorna uma lista de pares, combinando elementos das listas de entrada</li><li><input type="radio"/> retorna os elementos comuns entre as duas listas</li></ul>
<pre>const fun = ([x, ...xs], z) =&gt; {   if (z === 0) return true;   else if (x === undefined    z &lt; 0) return false;   else return fun(xs, z - x)    fun(xs, z); };</pre>	<ul style="list-style-type: none"><li><input type="radio"/> retorna todos os subconjuntos que somam o valor z</li><li><input checked="" type="radio"/> verifica se existe um subconjunto cujos elementos somam o valor z</li><li><input type="radio"/> retorna a soma de todos os subconjuntos que podem ser formados</li><li><input type="radio"/> verifica se todos os elementos da lista são menores que z</li></ul>

Para cada problema a seguir, assinale a alternativa que melhor representa a implementação da função

Calcular o mdc entre 2 inteiros. Qual o trecho A e o trecho B, respectivamente? <pre>const fun = (a, b) =&gt; {   if (a == b) return a   else if (a&gt;b) return A   else return B }</pre>	<ul style="list-style-type: none"><li><input type="radio"/> <code>fun(a,a-b)</code> e <code>fun(b,b-a)</code></li><li><input type="radio"/> <code>fun(a,b-a)</code> e <code>fun(b-a,a)</code></li><li><input checked="" type="radio"/> <code>fun(a,b)</code> e <code>fun(b,a)</code></li><li><input type="radio"/> <code>fun(a-b,b)</code> e <code>fun(a,b-a)</code></li></ul>
---	--

Como calcular o fatorial de um número natural?	<input type="radio"/> const fun = (n) => n==1 ? 1 : fun(n-1) <input type="radio"/> const fun = (n) => n==1 ? 1 : fun(n-1)*n <input checked="" type="radio"/> const fun = (n) => n==0 ? 1 : fun(n-1)*n <input type="radio"/> const fun = (n) => n==0 ? 1 : fun(n-1)
Como remover todos os elementos pares de uma lista de inteiros? <pre>const fun = ([x, ...xs]) =&gt; {   if (x === undefined) return [];   else if A return fun(xs);   else return [x, ...fun(xs)]; };</pre>	<input checked="" type="radio"/> (x % 2 === 0) <input type="radio"/> (x / 2 === 0) <input type="radio"/> (x % 2 === 1) <input type="radio"/> (x / 2 === 1)
Função recursiva que verifica se uma lista de inteiros está ordenada em ordem crescente. Qual o trecho? <pre>const fun = ([x, y, ...xs]) =&gt; {   if (y === undefined) return true;   else if (x &gt; y) return false;   else return fun(B); };</pre>	<input type="radio"/> xs <input checked="" type="radio"/> [y, ...xs] <input type="radio"/> xs.slice(1) <input type="radio"/> [x, ...xs]

Para cada aplicação de função a seguir, assinale a alternativa que representa o resultado obtido

<pre>const fun = (n) =&gt; {   if (n &gt;= 101) return n-101   else return fun(fun(n+11)) } console.log(fun(98))</pre>	<input type="radio"/> 88 <input checked="" type="radio"/> 91 <input type="radio"/> 99 <input type="radio"/> 101
<pre>const fun = (n) =&gt; {   if (n == 0) return 0   else return 1+fun(Math.floor(n/10)) } console.log(fun(4887655303))</pre>	<input type="radio"/> 4 <input checked="" type="radio"/> 10 <input type="radio"/> 49 <input type="radio"/> 4000000000
<pre>const fun = (fn, [x, ...xs]) =&gt; {   if (x === undefined) return [];   else return [fn(x), ...fun(fn, xs)]; }; console.log(fun((n)=&gt;n*2, [1,2,3,4,5]));</pre>	<input type="radio"/> [1, 4, 9, 16, 25] <input type="radio"/> [2, 3, 4, 5, 6] <input checked="" type="radio"/> [2, 4, 6, 8, 10] <input type="radio"/> [1, 2, 3, 4, 5]
<pre>const fun = ([x, y, ...xs]) =&gt; {   if (y === undefined) return [];   else return [x + y, ...fun(xs)]; }; console.log(fun([1, 2, 3, 4, 5]));</pre>	<input checked="" type="radio"/> [3, 7] <input type="radio"/> [3, 5, 7, 9] <input checked="" type="radio"/> [3, 7, 5] <input type="radio"/> [15]

As funções a seguir resolvem problemas bem conhecidos. Para cada questão, escreva o nome do problema (apenas o nome famoso).

<pre>const fun = (n, a = 0, b = 1) =&gt; {   if (n === 0) return a;   if (n === 1) return b;   return fun(n - 1, b, a + b); };</pre>	<u>Fibonacci</u> ✓
<pre>const fun = ([x, ...xs]) =&gt; {   if (x === undefined) return [];   const a = xs.filter(n =&gt; n &lt;= x);   const b = xs.filter(n =&gt; n &gt; x);   return [...fun(a), x, ...fun(b)]; };</pre>	<u>Quick sort</u> ✓