

1569 . Número de maneiras de reordenar o array para obter o mesmo BST

Hint

⋮

Duro

✓

👍 1,7K

👎 200

☆

↻

🔒 Empresas

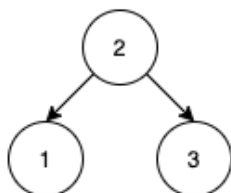
Dado um array `nums` que representa uma permutação de inteiros de `1` até `n`. Vamos construir uma árvore de pesquisa binária (BST) inserindo os elementos de `nums` ordem em um BST inicialmente vazio. Encontre o número de maneiras diferentes de reordenar `nums` de modo que o BST construído seja idêntico àquele formado a partir do array original `nums`.

- Por exemplo, dado `nums = [2, 1, 3]`, teremos 2 como raiz, 1 como filho esquerdo e 3 como filho direito. A matriz `[2, 3, 1]` também produz o mesmo BST, mas `[3, 2, 1]` produz um BST diferente.

Retorne o número de maneiras de reordenar `nums` de forma que o BST formado seja idêntico ao BST original formado a partir de `nums`.

Como a resposta pode ser muito grande, **retorne-a módulo** `$10^9 + 7$` .

Exemplo 1:

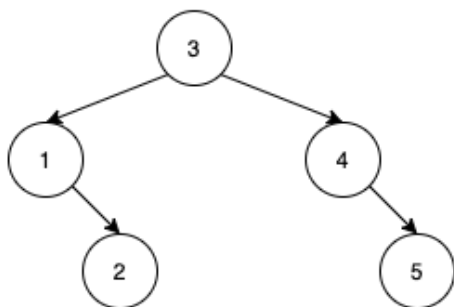


Entrada: `nums = [2,1,3]`

Saída: 1

Explicação: Podemos reordenar `nums` para ser `[2,3,1]`, o que produzirá o mesmo BST. Não há outras maneiras de reordenar `nums` que produzam o mesmo BST.

Exemplo 2:



Entrada: `nums = [3,4,5,1,2]`

Saída: 5

Explicação: As 5 matrizes a seguir produzirão o mesmo BST:

```

[3,1,2,4,5]
[3,1,4,2,5]
[3,1,4,5,2]
[3,4,1,2,5]
[3,4,1,5,2]
  
```