Description | Solutions

Get Maximum Sum Arr (Fungible :)

FULLTIME | RELATED PROBLEMS

At Amazon, the team at the fulfillment center is responsible for the packaging process. There is an array, `item_weights`, of `n` items to pack. The team needs to create a new array, `new_arr`, by removing exactly `n/3` items from `item_weights` without changing the order of those remaining.

- The `sum_arr` of array `new_arr` is defined as the sum of the weights or elements in the first half of the array minus the sum of the weights in the second half of the array. - Given `n` items and an array `item_weights`, find the maximum `sum_arr` possible.

Function Description

Complete the function `getMaxSumArr` in the editor below.

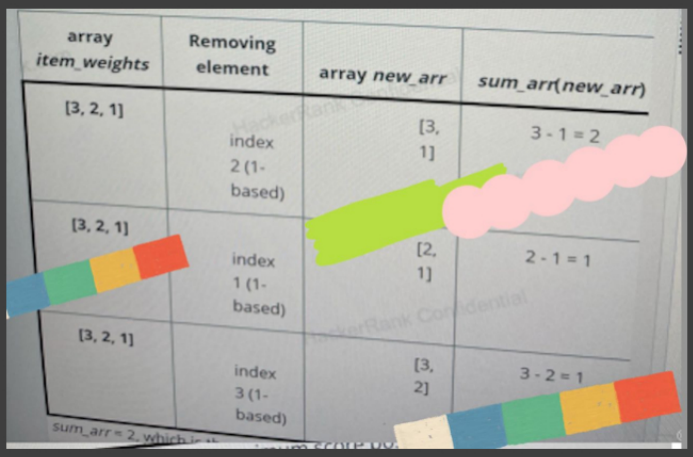
`getMaxSumArr` has the following parameters:

`int item_weights[n]`: item weights

Returns
`int`: the maximum possible `sum_arr`

Example 1:

```
Input:  item_weights = [3, 2, 1]
Output: 2
Explanation:
```



Example 2:

```
Input:  item_weights = [1, 3, 4, 7, 5, 2]
Output: 4
Explanation:

Given n = 6, item_weights=[1, 3, 4, 7, 5, 2], remove the elements
1, 3 to leave new_arr=[4, 7, 5, 2], hence the sum will be (4+7)-
(5+2) = 4.
```

Constraints:

- 3 ≤ `N` ≤ 10^5
- -10^4 ≤ `item_weights[i]` ≤ 10^4
- `n` is divisible by 3

Problem Source

Related Problems

Python | Log In

```
1
2  class Solution:
3      def getMaxSumArr(self, item_weights: List[int]) -> int:
4
```

Testcase | Result

Case 1 | Case 2 | +

item_weights:

[3, 2, 1]

output:

Submit