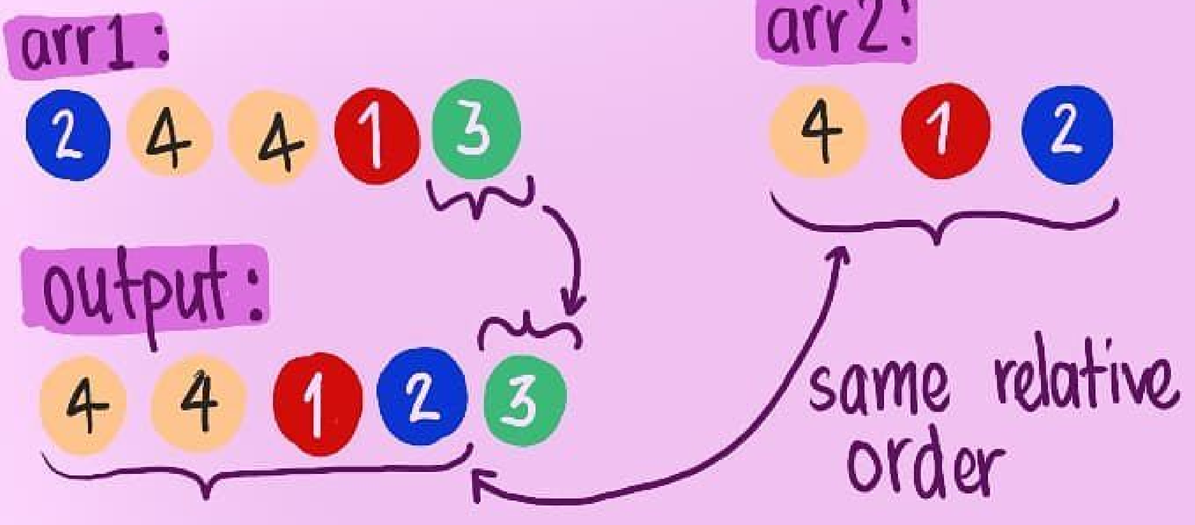
1122. Relative Sort Array

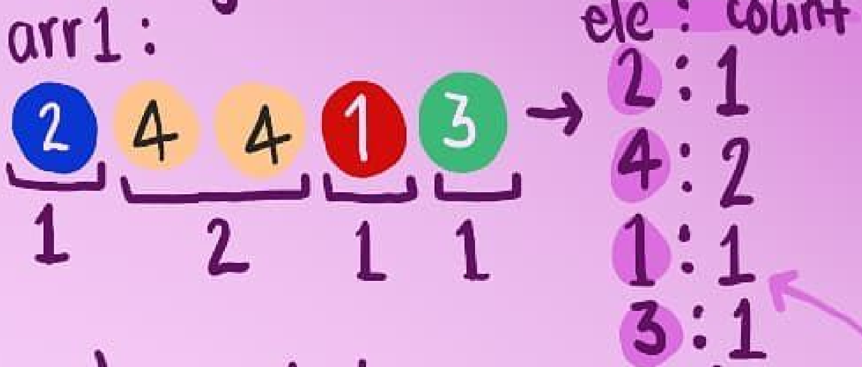
Given two arrays arr1 and arr2, elements of arr2 are distinct, and all elements in arr2 are also in arr1.

Sort the elements of arr1 such that the relative ordering of items in arr1 are the same as in arr2. Elements that don't appear in arr2 should be placed at the end of arr1 in ascending order.

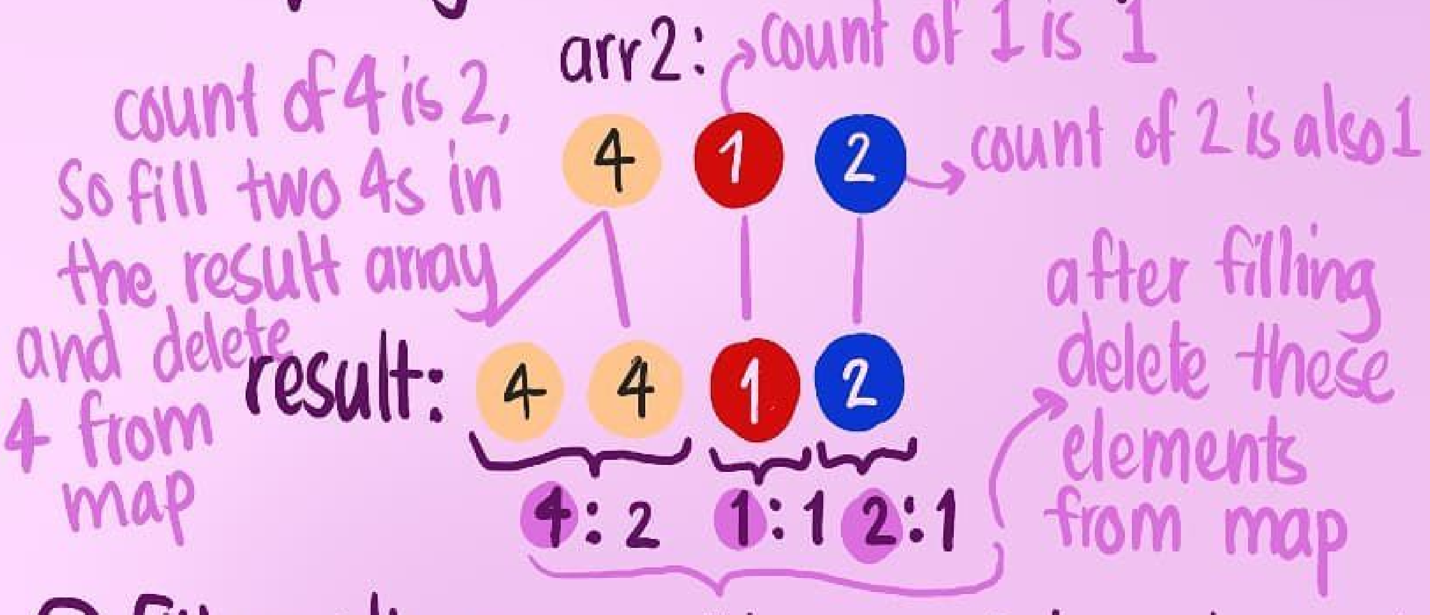


APPROACH

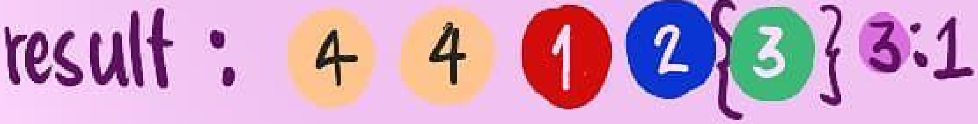
① Count frequency of each element of arr1



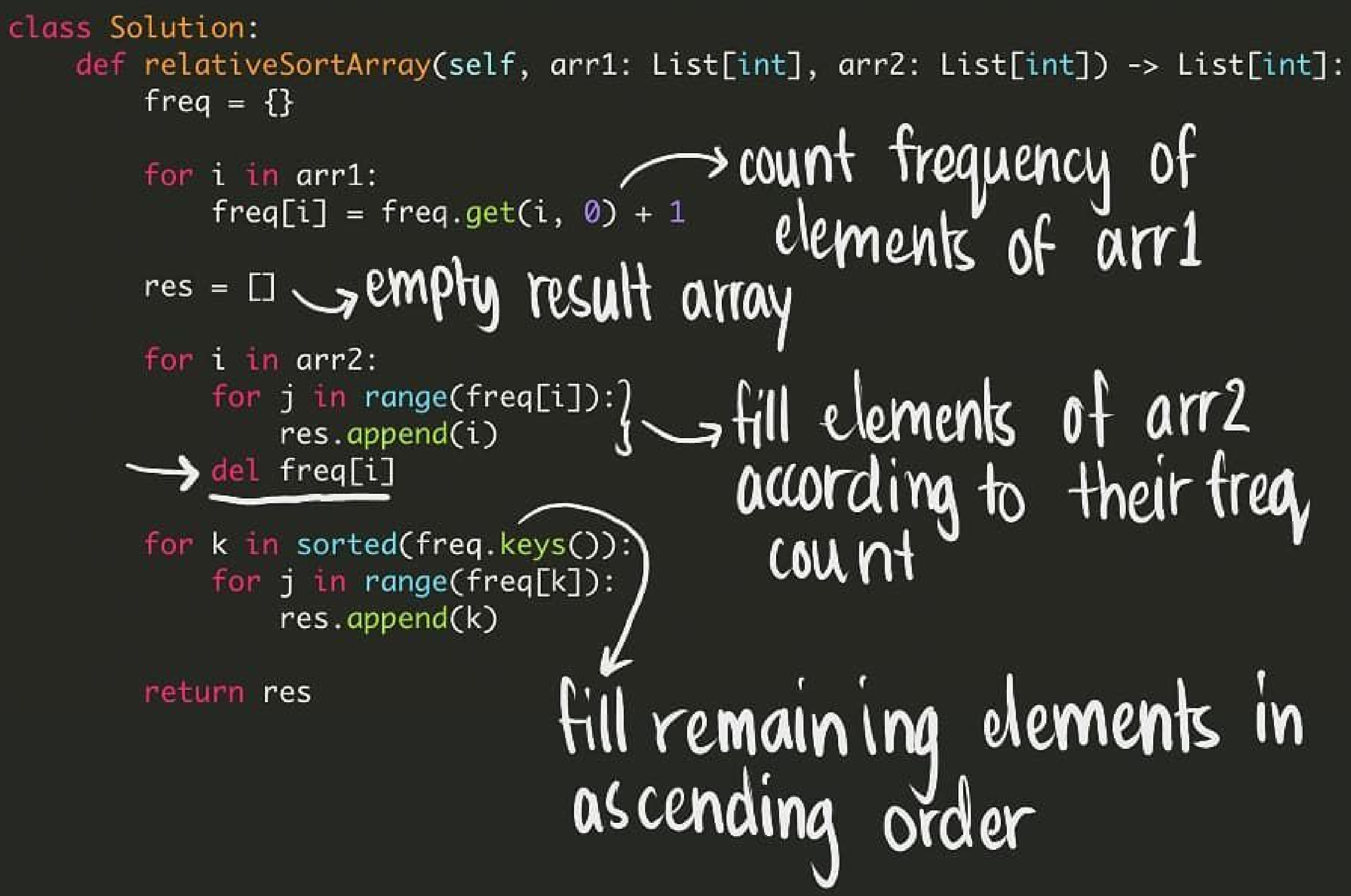
② for every element in arr2, start filling result array according to frequency of arr2[i] in map.


arr2:
count of 1 is 1
count of 4 is 2, so fill two 4s in the result array and delete 4 from map
4 1 2
count of 2 is also 1
result: 4 4 1 2
4 : 2 1 : 1 2 : 1
after filling delete these elements from map

③ Fill result array with remaining elements in map (ascending order)



1122. RELATIVE SORT ARRAY



```
class Solution:
    def relativeSortArray(self, arr1: List[int], arr2: List[int]) -> List[int]:
        freq = {}

        for i in arr1:
            freq[i] = freq.get(i, 0) + 1

        res = []

        for i in arr2:
            for j in range(freq[i]):
                res.append(i)
            del freq[i]

        for k in sorted(freq.keys()):
            for j in range(freq[k]):
                res.append(k)

        return res
```

count frequency of elements of arr1

empty result array

fill elements of arr2 according to their freq count

fill remaining elements in ascending order