INTERSECTION OF TWO SORTED ARRAYS

☆ In this problem we have to return a new array which contains the union of the two sorted arrays.

Brute force solution is to iterate through each element of first array and check if it exists in the second array, if yes then it is added to the final. To know if the element is already in the final array, we use a hashlist. Time complexity is O(n1 × n2) Space Complexity is O(n2)

Optimal solution uses the two pointer approach, where each pointer starts at the start of both arrays. Whenever one element is lesser, its pointer is moved and if both are equal, the element is added to final and both pointers are moved.

Pseudocode :

```
intersection(arr1, n1, arr2, n2) {
    i, j = 0, 0
    int final
```

```
        while(i< n1 && j<n2) {
           if (arr1[i] == arr2[j]) {
              final.push(arr1[i])
              i++
              j++
           } else if (arr1[i] > arr2[j]) {
              j++
           } else i++
        }
   }
```