

UNION 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 use a set and just blindly insert all elements of both arrays into the set. Though this has space complexity as well.

Optimal solution is obviously using two pointer approach, one at the start of both arrays. We keep on checking if the element is not there already and appending the smaller one out of the 2.

Dry Run :

array 1 : { 1, 1, 2, 3, 4, 5 }

array 2 : { 2, 3, 4, 4, 5, 6 }

union : { }



array 1 : { 1, 0, 2, 3, 4, 5 }

array 2 : { 2, 3, 4, 4, 5, 6 }

union : { 1, } ↓

array 1 : { 1, 1, 2, 3, 4, 5 }

array 2 : { 2, 3, 4, 4, 5, 6 }

union : { 1, } ↓

array 1 : { 1, 1, 2, 3, 4, 5 }

array 2 : { 2, 3, 4, 4, 5, 6 }

union : { 1, 2, } ↓

array 1 : { 1, 1, 2, 3, 4, 5 }

array 2 : { 2, 3, 4, 4, 5, 6 }

union : { 1, 2, } ↓

array 1 : { 1, 1, 2, 3, 4, 5 }

array 2 : { 2, 3, 4, 4, 5, 6 }

union : { 1, 2, 3, } ↓

array 1 : { 1, 1, 2, 3, 4, 5 }

array 2 : { 2, 3, 4, 4, 5, 6 }
 ↓
 j

union : { 1, 2, 3, }

array 1 : { 1, 1, 2, 3, 4, 5 }

array 2 : { 2, 3, 4, 4, 5, 6 }
 ↓
 j

union : { 1, 2, 3, 4, }

array 1 : { 1, 1, 2, 3, 4, 5 }

array 2 : { 2, 3, 4, 4, 5, 6 }
 ↓
 j

union : { 1, 2, 3, 4, }

↓

array 1 : { 1, 1, 2, 3, 4, 5 }

array 2 : { 2, 3, 4, 4, 5, 6 }
 ↓
 j

union : { 1, 2, 3, 4 }

array 1 : { 1, 1, 2, 3, 4, 5 }

array 2 : { 2, 3, 4, 4, 5, 6 }
 ↓
 j

union : { 1, 2, 3, 4, 5 }

↓

array 1 : { 1, 1, 2, 3, 4, 5 }
i

array 2 : { 2, 3, 4, 4, 5, 6 }
j

union : { 1, 2, 3, 4, 5, }

Finally, union : { 1, 2, 3, 4, 5, 6 }

Pseudocode :

```
union (arr1, N1, arr2, N2) {
    int final []
    i, j, c = 0, 0, 0
    while (i < N1 && j < N2) {
        if (arr1[i] <= arr2[j]) {
            if (final[c] != arr1[i]) {
                final[c] = arr1[i]
                c++
            }
            i++
        } else {
            if (final[c] != arr2[j]) {
                final[c] = arr2[j]
                c++
            }
            j++
        }
    }
    while (i < N1) {
        if (final[c] != arr1[i]) {
            final[c] = arr1[i]
        }
    }
}
```

C ++

}

i ++

}

while (j < N1) {

 if (final[c] != arr2[j]) {

 final[c] = arr2[j]

 c ++

 }

}

j ++

}