HASHING DATA STRUCTURE

Relative Sort Array

arr1 = [2, 1, 2, 5, 7, 1, 9, 3, 6, 8, 8]

Arr2 = [2, 1, 8, 3]

Solution :   2 2   1 1   8 8   3   5 6 7 9

Hello world

Relative Sort Array

arr1 = [2, 1, 2, 5, 7, 1, 9, 3, 6, 8, 8]

Arr2 = [99, 22, 444, 56]

Solution :   **1 1 2 2 3 5 6 7 8 8 9**

Hello world

HASHING DATA STRUCTURE

■ **Relative Sort Array**

■ **arr1 = [2, 1, 2, 5, 7, 1, 9, 3, 6, 8, 8]**

■ **Arr2 = [2, 1, 8, 3]**

map <int, int > mp;

Solution :   Map   [*ordered map*]

    1) Create Map of arr1

      Let's say map <int, int > mp;

    2) Iterate elements from arr2

    for(int i=0 ; i< 4 ; I ++)

    {

      Int x = arr2[i] ;

      We find x in map

      Then Print x as many as it present in arr1

    }

1 -> 2

2 -> 2

3 -> 1

5 -> 1

6 -> 1

7 -> 1

8 -> 2

9 -> 1

Hello world

HASHING DATA STRUCTURE

Relative Sort Array

arr1 = [2, 1, 2, 5, 7, 1, 9, 3, 6, 8, 8]

Arr2 = [2, 1, 8, 3]

Solution :   for(int i=0 ; i< 4 ; I ++)

{
    Int x = 2

}

Ans = 2 2

map <int, int > mp;

1 -> 2

~~2 -> 2~~

3 -> 1

5 -> 1

6 -> 1

7 -> 1

8 -> 2

9 -> 1

Hello world

◻ **Relative Sort Array**

◻ **arr1 = [2, 1, 2, 5, 7, 1, 9, 3, 6, 8, 8]**

◻ **Arr2 = [2, 1, 8, 3]**

Solution :   **for(int i=0 ; i< 4 ; I ++)**

{

Int x = 1

}

Ans = 2 2 1 1

map <int, int > mp;

~~1 -> 2~~

~~2 -> 2~~

3 -> 1

5 -> 1

6 -> 1

7 -> 1

8 -> 2

9 -> 1

Hello world

HASHING DATA STRUCTURE

Relative Sort Array

arr1 = [2, 1, 2, 5, 7, 1, 9, 3, 6, 8, 8]

Arr2 = [2, 1, 8, 3]

Solution :   for(int i=0 ; i< 4 ; I ++)

  {

Int x = 8

  }

Ans = 2 2 1 1 8 8

map <int, int > mp;

~~1 -> 2~~

~~2 -> 2~~

3 -> 1

5 -> 1

6 -> 1

7 -> 1

~~8 -> 2~~

9 -> 1

Hello world

Relative Sort Array

arr1 = [2, 1, 2, 5, 7, 1, 9, 3, 6, 8, 8]

Arr2 = [2, 1, 8, 3]

Solution : **for(int i=0 ; i< 4 ; I ++)**

{

Int x = 3

}

Ans = 2 2 1 1 8 8 3

map <int, int > mp;

~~1 -> 2~~

~~2 -> 2~~

~~3 -> 1~~

5 -> 1

6 -> 1

7 -> 1

~~8 -> 2~~

9 -> 1

Hello world

HASHING DATA STRUCTURE

☐ **Relative Sort Array**

☐ **arr1 = [2, 1, 2, 5, 7, 1, 9, 3, 6, 8, 8]**

☐ **Arr2 = [2, 1, 8, 3]**

Solution :   for(int i=0 ; i< 4 ; I ++)

                {
                            Int x = 3

                }

            Now iterate map in line by line

Ans = 2 2 1 1 8 8 3 5 6 7 9

map <int, int > mp;

~~1 -> 2~~

~~2 -> 2~~

~~3 -> 1~~

5 -> 1

6 -> 1

7 -> 1

~~8 -> 2~~

9 -> 1

Hello world