Given Array

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 8 | 5 | 2 | 1 | 3 | 5 | 4 | 2 | 3 |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|  |  |  |  |  |  |  |  | 1 |  |
|  |  |  |  |  | 1 |  |  |  |  |
|  |  | 1 |  |  |  |  |  |  |  |
|  | 1 |  |  |  |  |  |  |  |  |
|  |  |  | 1 |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  | 2 |  |  |  |  |
|  |  |  |  | 1 |  |  |  |  |  |
|  |  | 2 |  |  |  |  |  |  |  |
| 0 | 1 | 2 | 2 | 1 | 2 | 0 | 0 | 1 | 0 |

INDEX

COUNT

COUNT

FINAL COUNT

Sum the index from 1-9 not 0-9, 0 index will be as it is ( From Final Count)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 3 | 5 | 6 | 8 | 8 | 8 | 9 | 9 |

From Given Array And Final Count after sum from index 1 , Match and Place Your Numbers Sortedly

Given Array For Maching

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 8 | 5 | 2 | 1 | 3 | 5 | 4 | 2 | 3 |

Index

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 2 | 1 | 2 | 0 | 0 | 1 | 0 |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 3 | 5 | 6 | 8 | 8 | 8 | 9 | 9 |

Final Index Approach 1:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  | 8 |
|  |  |  |  |  |  |  | 5 |  |
|  |  | 2 |  |  |  |  |  |  |
| 1 |  |  |  |  |  |  |  |  |
|  |  |  |  | 3 |  |  |  |  |
|  |  |  |  |  |  | 5 |  |  |
|  | 2 |  |  |  | 4 |  |  |  |
|  |  |  | 3 |  |  |  |  |  |
| 1 | 2 | 2 | 3 | 3 | 4 | 5 | 5 | 8 |

Approach 2:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |

Final Index

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  | 8 |
|  |  |  |  |  |  |  | 5 |  |
|  |  | 2 |  |  |  |  |  |  |
| 1 |  |  |  |  |  |  |  |  |
|  |  |  |  | 3 |  |  |  |  |
|  |  |  |  |  |  | 5 |  |  |
|  | 2 |  |  |  | 4 |  |  |  |
|  |  |  | 3 |  |  |  |  |  |
| 1 | 2 | 2 | 3 | 3 | 4 | 5 | 5 | 8 |

Final Sorted Array

Extra Part>>>>>

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 2 | 1 | 2 | 0 | 0 | 1 | 0 |

Two approach we can do , 1.Like On After summation

Each element from Count , We found the Exact Max Numbers . BuT Here’s The problem for 0 zeros there is such no Index for 0 zero, such there was zero in the elements then what happened? Right I Know we can implement code for that, but the scenario would be complex ,right? 2.So for this reason we sould choose the Index 0-9 and From Count Max we can Decrease the right or Max elements By one . Than we do the tmplement our logic on Code.

Psudo Code : (4 for loops Only!!)

N=total elements of the array /array size

K=(0-9) Unique Numbers in the array/max digit in the array

For (i=0; i<n; i++)

{

++count[A[i]];

}

For (i=1; i<=k;i++)

{

Count[i]= count[i]+count[i-1];

}

For (i=n-1; i>=0; i--)

{

b[--count[A[i]]]=A[i];

}

For(i=0;i<n;i++) Or For (i=0;i<n-1;i++)

{

A[i]=b[i];

}

My sudo code:

For(i=0;i<size;i++)

{

++count[Mus[i]];

}

For(i=1;i<=k;i++)

{

Count[i]=count[i]+count[i-1];

}

For (i=size-1;i>=0;i--)

{

Final[--count[Mus[i]]]=Mus[i];

}

For(i=0;i<size;i++) Or For(i=0;i<size-1;i++)

{

Final[i]=Mus[i];

}

**PseudoCode Algorithm:**

Int Final\_Sorted[40];

Int Count[10];

Int K=Given\_array[0];

For(i=1;i<size;i++)

{

If (Given\_array[i]>K)

K=Given\_array[i];

}

For (i=0;i<=K;i++)

{

Count[i]=0;

}

For(i=0;i<size;i++)

{

++Count[Given\_array[i]];

}

For(i=1;i<=K;i++)

{

Count[i]=Count[i]+Count[i-1];

}

For (i=size-1;i>=0;i--)

{

Final\_Sorted[--Count[Given\_array[i]]]=Given\_array[i];

}

For(i=0;i<size;i++)

{

Final\_Sorted[i]=Given\_array[i];

}