

## Сортиране чрез сливане.

### Counting sort.

1. Сортиране чрез сливане — сортиране на 2 сортирани масива.

Пример:

arr1: 2 3 7 10 12 } i1 → 0  
arr2: 1 4 6 11 13 } i2 → 0

size arr3 → size arr1 + size arr2; arr3 → сортиран.

arr3[0] = min(arr1[0], arr2[0]) = 1

arr1: 2 3 7 10 12 } i1 → 0  
arr2: ✕ 4 6 11 13 } i2 → 1  
arr3: 1

arr[1] = min(arr[0], arr[1]) = min(arr1[i1], arr2[i2]) = 2

arr1: ✕ 3 7 10 12 } i1 → 1  
arr2: ✕ 4 6 11 13 } i2 → 1  
arr3: 1 2

arr1: ✕ ✕ 7 10 12 } i1 → 2  
arr2: ✕ 4 6 11 13 } i2 → 1  
arr3: 1 2 3

arr1: ✕ ✕ 7 10 12 } i1 → 2  
arr2: ✕ ✕ 6 11 13 } i2 → 2  
arr3: 1 2 3 4

arr3: 1 2 3 4 6 7 10 11 12 13

## 2. Counting sort - само с положителни числа

Пример:

arr: 3 4 1 1 2 3 6 5 6 6

осир: колко пъти се е появило числото  $i$  в arr

осир:  $\frac{2}{1} \frac{1}{2} \frac{2}{3} \frac{1}{4} \frac{1}{5} \frac{3}{6}$   $\rightarrow$  осир[arr[i]]++

След това се отпечатват 2 единици, 1 двойка, 2 тројки...

Output: 1 1 2 3 3 4 5 6 6 6

## Стабилен Counting Sort

Проблем: студент - оценка за семестър.

Student 1 - 3

Student 6 - 3

Student 2 - 4

Student 7 - 6

Student 3 - 1

Student 8 - 5

Student 4 - 1

Student 9 - 6

Student 5 - 2

Student 10 - 6

} обекти.

• За да направим стабилен Counting sort ще използваме масив или префикси.

Пример за:

arr:  $\frac{5}{0} \frac{7}{1} \frac{3}{2} \frac{2}{3} \frac{6}{4}$

pref[i] - ще съдържа сумата на числата до  $i$ -то.

pref[0] = 5 = arr[0]

pref[1] = 5 + 7 = 12

pref[2] = 5 + 7 + 3 = 15

pref[3] = 5 + 7 + 3 + 6 = 21

}  $\Rightarrow$  pref[i] = pref[i-1] + arr[i],  
 $i \geq 1$

За: arr: 3 4 1 1 2 3 6 5 6 6 (оценките на студентите)

осир:  $\frac{2}{1} \frac{1}{2} \frac{2}{3} \frac{1}{4} \frac{1}{5} \frac{3}{6}$

Направим масив или префикси на осир, който се казва

sum	$\frac{2}{1}$	$\frac{3}{2}$	$\frac{5}{3}$	$\frac{6}{4}$	$\frac{7}{5}$	$\frac{10}{6}$
-----	---------------	---------------	---------------	---------------	---------------	----------------

$sum[i]$  - показва кой е последният индекс, на който  $i$  ще се появи.

arr: 3 4 1 1 2 3 6 5 6 6  
 arrSorted: 1 2 3 4 5 6 7 8 9 10

$sum[6] = 10$   
 $\downarrow$   
 $sum[6] = 9$

arr: 3 4 1 1 2 3 6 5 6 6  
 arrSorted: 1 2 3 4 5 6 7 8 9 10

$sum[6] = 9$   
 $\downarrow$   
 $sum[6] = 8$  ) последн. поз., на който сл. 6-ца ще се появи.

arr: 3 4 1 1 2 3 6 5 6 6  
 arrSorted: 1 2 3 4 5 6 7 8 9 10

$sum[5] = 7$   
 $\downarrow$   
 $sum[5] = 6$  (invalid)

arr: 3 4 1 1 2 3 6 6 6 6  
 arrSorted: 1 2 3 4 5 6 7 8 9 10

$sum[6] = 8$   
 $\downarrow$   
 $sum[6] = 7$  (invalid)

arr: 3 4 1 1 2 3 6 6 6 6  
 arrSorted: 1 2 3 4 5 6 7 8 9 10

$sum[3] = 5$   
 $\downarrow$   
 $sum[3] = 4$

arr: 3 4 1 1 2 6 6 6 6 6  
 arrSorted: 1 2 3 4 5 6 7 8 9 10

$sum[2] = 3$   
 $\downarrow$   
 $sum[2] = 2$

arr: 3 4 1 1 6 6 6 6 6 6  
 arrSorted: 1 2 3 4 5 6 7 8 9 10

arr: 3 4 1 6 6 6 6 6 6 6  
 arrSorted: 1 2 3 4 5 6 7 8 9 10

и т.н.

Накрая сортираните студенти ще бъдат подредени по следния начин:

Student 3  
1

Student 4  
1

Student 5  
2

Student 1  
3

Student 6  
3

Student 2  
4

Student 8  
5

Student 7  
6

Student 9  
6

Student 10  
6