

Make Us Whole

Problem	Submissions	Leaderboard	Discussions
---------	-------------	-------------	-------------

Author : niko.ac.id

Anda diberikan sebuah array berisi bilangan bulat acak. Anda diminta untuk mengimplementasikan algoritma Merge Sort untuk mengurutkan array tersebut.

Namun, alih-alih menggabungkan dua sub-array secara langsung, Anda diminta untuk menggunakan pendekatan divide and conquer dengan cara berikut:

1. Pertama, bagi array menjadi dua bagian yang sama besar (jika ukuran array ganjil, maka bagian kiri harus lebih besar dari bagian kanan).
2. Kemudian, bagi kembali masing-masing bagian tersebut menjadi dua bagian yang sama besar.
3. Lakukan langkah 2 hingga setiap bagian hanya memiliki satu elemen.
4. Kemudian, gabungkan dua sub-array pertama menjadi sebuah array yang terurut menggunakan pendekatan Merge Sort.
5. Lakukan langkah 4 hingga semua sub-array tergabung dan array awal menjadi terurut.

Implementasikan algoritma Merge Sort dengan menggunakan pendekatan divide and conquer seperti di atas!

Input Format

- Baris pertama berisi sebuah integer n , yang merupakan panjang dari array.
- Baris kedua berisi n buah integer yang merupakan isi dari array.

Constraints

- $1 \leq n$
- $1 \leq \text{arr}[i] \leq 1000$

Output Format

- Setiap baris menampilkan array kiri dan kanan saat melakukan pembagian dalam algoritma Merge Sort.
- Baris terakhir merupakan Sorted Array

Sample Input 0

```
10
4 0 6 5 3 9 8 2 1 7
```

Sample Output 0

```
[4, 0, 6, 5, 3] [9, 8, 2, 1, 7]
[4, 0] [6, 5, 3]
```

```
[4] [0]
[6] [5, 3]
[5] [3]
[9, 8] [2, 1, 7]
[9] [8]
[2] [1, 7]
[1] [7]
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

Sample Input 1

```
5
1 8 7 62 2
```

Sample Output 1

```
[1, 8] [7, 62, 2]
[1] [8]
[7] [62, 2]
[62] [2]
[1, 2, 7, 8, 62]
```

[f](#) [t](#) [in](#)

Contest ends in 2 days

Submissions: [79](#)

Max Score: 100

Difficulty: Medium

Rate This Challenge:

☆☆☆☆☆

[More](#)

Python 3



```
1 def merge(left, right):
2     result = []
3     i = j = 0
4     while i < len(left) and j < len(right):
5         if left[i] <= right[j]:
6             result.append(left[i])
7             i += 1
8         else:
9             result.append(right[j])
10            j += 1
11    result += left[i:]
12    result += right[j:]
13    return result
14
15 def merge_sort(arr):
16     if len(arr) <= 1:
17         return arr
18     middle = len(arr) // 2
19     left = arr[:middle]
20     right = arr[middle:]
21     print(left, right)
22     left = merge_sort(left)
23     right = merge_sort(right)
24     return merge(left, right)
25
26 n = int(input())
27 arr = list(map(int, input().split()))
```

```
28  
29 result = merge_sort(arr)  
30 print(result)  
31
```

Line: 1 Col: 1



[Upload Code as File](#)



Test against custom input

Run Code

Submit Code