Algorithm 1 MergeSort$(A, left, right)$

Input: 数组$A[1...n]$，数组下标$left, right$
Output: 递归数组$A[left..right]$

```
1: if left ≥ right then
2:     return A[left..right]
3: end if
4: mid ← ⌊(left+right)/2⌋
5: MergeSort(A, left, mid).
6: MergeSort(A, mid + 1, right).
7: Merge(A, left, mid, right).
8: return A[left..right]
```

Algorithm 2 Merge$(A, left, mid, right)$

Input: 数组$A[1...n]$，数组下标$left, mid, right$
Output: 递增数组$A[left..right]$

```
1: A'[left..right] ← A[left..right]
2: i ← left, j ← mid + 1, k ← 0
3: while i ≥ mid and j ≥ right do
4:     if A'[i] ≥ A'[j] then
5:         A[left + k] ← A'[i]
6:         k ← k + 1, i ← i + 1
7:     else
8:         A[left + k] ← A'[j]
9:         k ← k + 1, j ← j + 1
10:    end if
11: end while
12: if i ≥ mid then
13:    A[left + k..right] ← A'[i..mid]
14: else
15:    A[left + k..right] ← A'[j..right]
16: end if
17: return A[left..right]
```