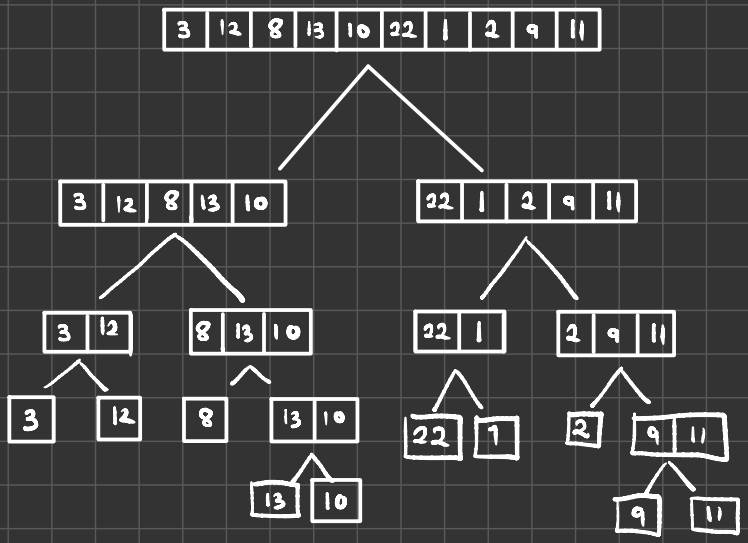
6. (2019)

Assuming a non-null integer array is inputed.

conside a sample array (drawn as blocks)

Merge Sort starts by recursively dividing the array by half into smaller arrays until only a single element is in each array.

Note that if only a single element is in the original array the algorithm exits immediately



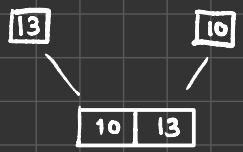
Next, the algorithm will recursively merge back the individual arrays in a sorted manner (I shall assume min to max)

For the merge consider the left half of the diagram the merge process is as follows

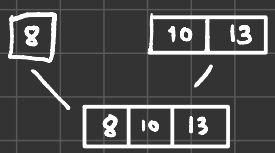
1. An array is created to store the sorted data
2. Iterate and Compare the left array and right array elements. If the left array's element is less than the right element's array insert it into the resulting array else insert the right array's element.

   If any array is empty then append the remaining elements of the other one into the resulting array

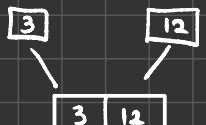
So in our example



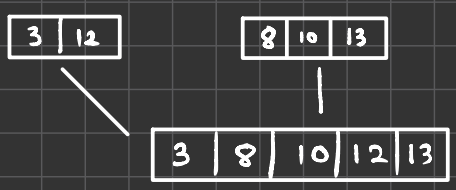
10 is added first then 13 is appended.
∵ 10 < 13



The next recursive merge 8 is added then 10, 13 is apended



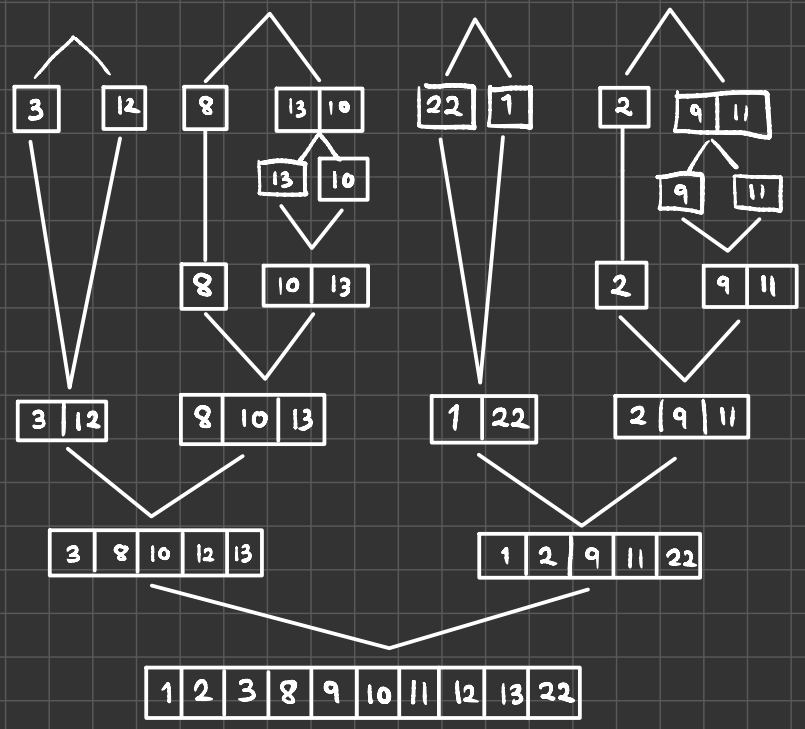
3 and 12 are merged



3 inserted ∵ 3 < 8
8 inserted ∵ 8 < 12
10 inserted ∵ 10 < 12
12 inserted ∵ 12 < 13
} Uses 2 pointers
13 inserted ∵ left array's empty

This is done until each array is merged into one sorted array

(The above arrays are omitted for space)



And thus the array is sorted.