

Oct 24

# Merge Sort (a, n)

floor [0.3] = 0

ceil [0.37] = 1

$O(1) \rightarrow$  If  $n=1$  then return  $a_1$

$$O(n) \left\{ \begin{array}{l} a_L = a_1, \dots, a_{\lfloor \frac{n}{2} \rfloor} \\ a_R = a_{\lfloor \frac{n}{2} \rfloor + 1}, \dots, a_n \end{array} \right.$$

$\leftarrow T(\lfloor \frac{n}{2} \rfloor)$

return MERGE (MergeSort( $a_L, \lfloor \frac{n}{2} \rfloor$ ),

$\uparrow$   
 $O(n)$

MergeSort( $a_R, n - \lfloor \frac{n}{2} \rfloor$ )

$\uparrow$   $T(n - \lfloor \frac{n}{2} \rfloor)$

$T(n)$  def max runtime of MergeSort over ALL inputs of size  $n$ .

Thm:  $T(n)$  is  $O(n \log n)$

$$T(n) \leq O(1) + O(n) + T(\lfloor \frac{n}{2} \rfloor) + T(n - \lfloor \frac{n}{2} \rfloor) + O(n)$$

$n=1, T(1) = O(1)$

$$\begin{aligned} n > 1 \quad T(n) &\leq O(n) + T(\lfloor \frac{n}{2} \rfloor) + T(n - \lfloor \frac{n}{2} \rfloor) \\ &= O(n) + T(\lfloor \frac{n}{2} \rfloor) + T(\lceil \frac{n}{2} \rceil) \\ &\leq O(n) + T(\frac{n}{2}) + T(\frac{n}{2}) \\ &= O(n) + 2T(\frac{n}{2}) \end{aligned}$$

$$T(n) = \begin{cases} O(1) & \text{if } n=1 \\ O(n) + 2T(\frac{n}{2}) & \text{if } n > 1 \end{cases}$$