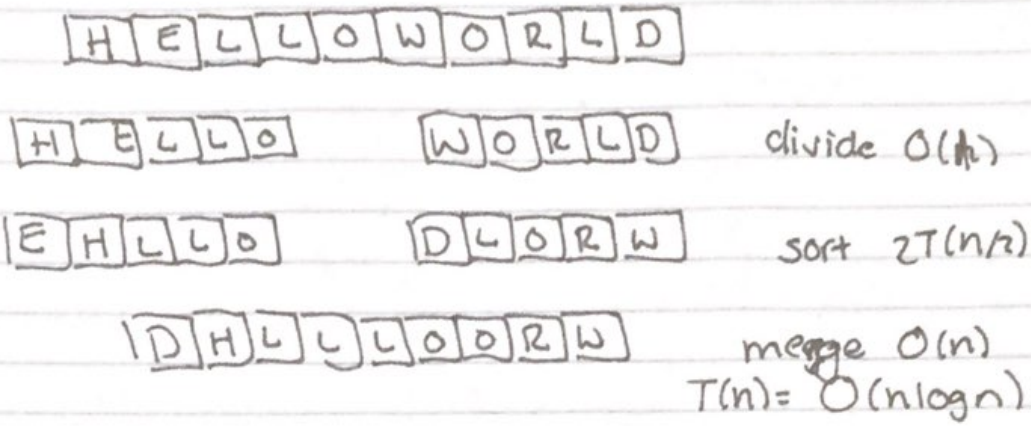
Merge-Sort

- divide arrays into two halves
- recorsively sort each half
- merge two halves to make a whole



Ways to prove:
- recursion tree
- Telescoping

If $T(n)$ satisfies, $T(n) = n \log_2 n$

$$T(n) = \underbrace{2T(\tfrac{n}{2})}_{\text{sorting}} + \underbrace{n}_{\text{merging}}$$

For $n > 1$

$$\frac{T(n)}{n} = \frac{2T(n/2)}{n} + 1$$

$$= \frac{T(n/2)}{n/2} + 1$$

$$= \frac{T(n/4)}{n/4} + 1 + 1$$

$$= \frac{T(n/n)}{n/n} + 1 + \dots\dots\dots^{(\log_2 n)} \dots + 1$$

$$= \log_2 n$$