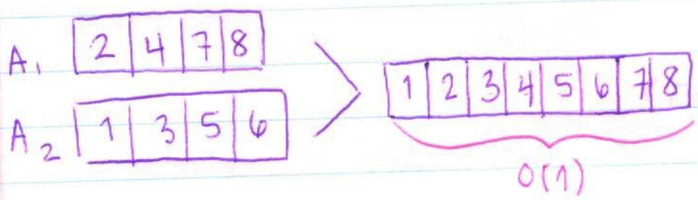
Nov 24
COMP 2402

Sorting Algorithms

→ MergeSort

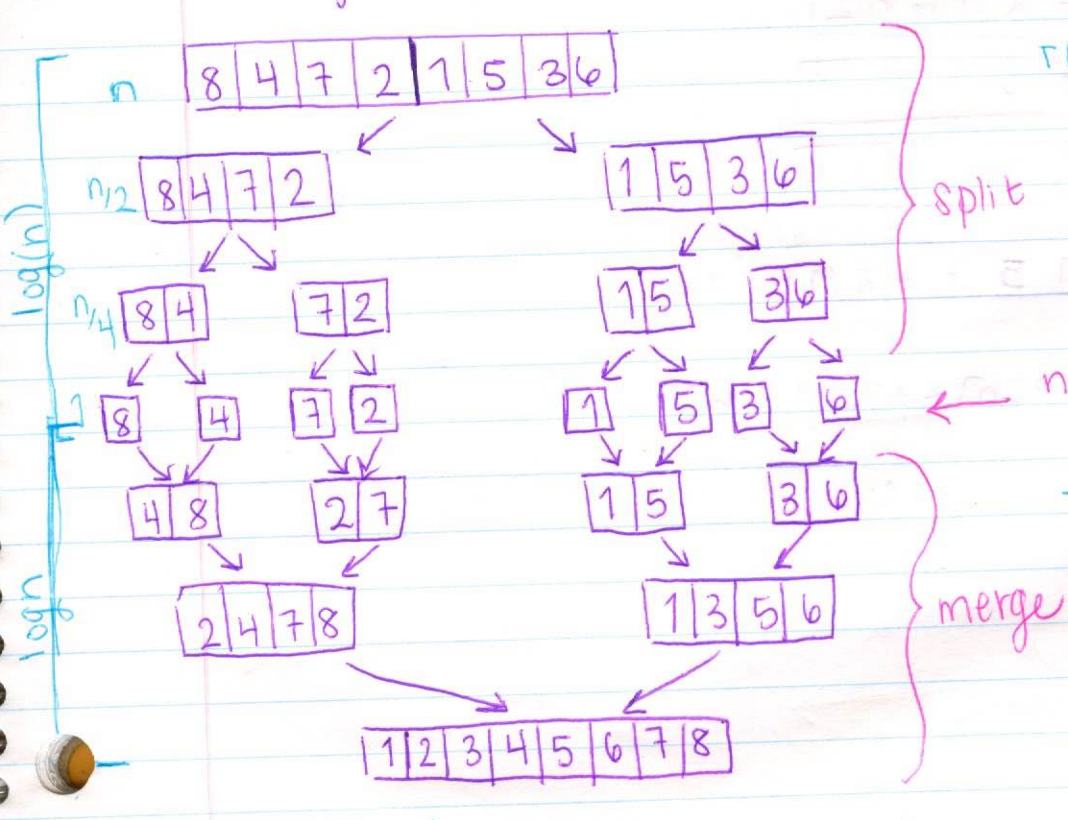
merge $(A[0 \ldots n/2], A[n/2+1 \ldots n-1])$



MergeSort $(A[0 \ldots n-1])$ — $T(n)$

if $(n \le 1)$ return A // $O(1)$
mergeSort $(A[0 \ldots n/2])$ — $T(n/2)$
mergeSort $(A[n/2 \ldots n])$ — $T(n/2)$
merge $(A[0 \ldots n/2], A[n/2 \ldots n])$ // $O(n)$

$T(n/2) = O(n/2) + 2T(n/4)$

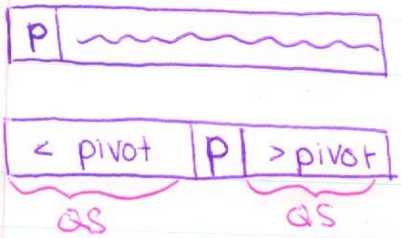


n sorted array
↳ 0 camparisons

$$T(n) = O(n) + 2T(n/2)$$
$$= O(n) + 2[O(n/2) + 2T(n/4)]$$
$$= O(n) + O(n) + 4T(n/4)$$
$$= O(n) + O(n) + 4[O(n/4) + 2T(n/8)]$$
$$= O(n) + O(n) + O(n) + 8T(n/8)$$
$$\vdots$$
$$= K \cdot O(n) + 2^K T(n/2^K)$$
$$= O(n \log n) + O(n)$$

recurse until...
$n/2^K = 1 \longrightarrow \log(n) = K$

→ QuickSort



QuickSort(A[0...n])

    pivot = A[0]
    p = partition(A[0...n], pivot)
    QuickSort(A[0...p])
    QuickSort(A[p+1...n])

partition(A[0...n], pivot)

p

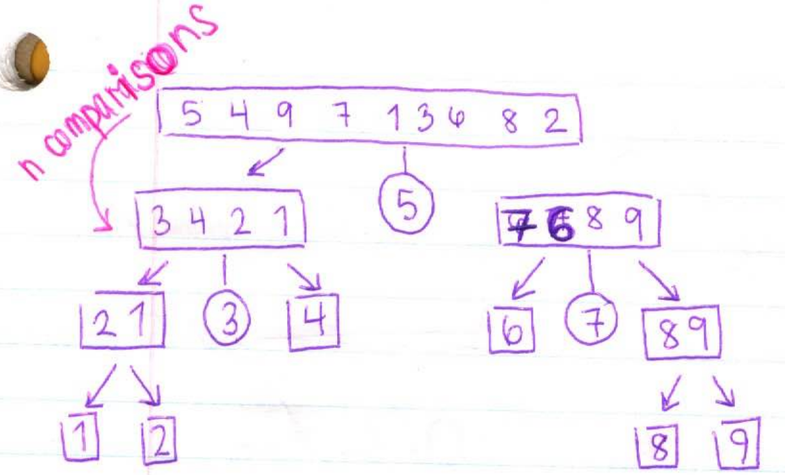
| 5 4 9 7 3 1 6 8 2 |

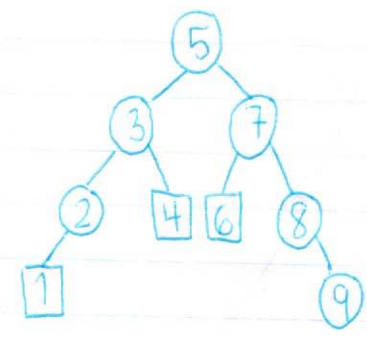
| 5 4 2 7 3 1 6 8 9 |

| 5 4 2 1 3 7 6 8 9 |

| 3 4 2 1 5 7 6 8 9 |

A[0...p] < A[p] < A[p+1...n]





$$\Pr\{i \text{ path } x\} = 1/(i-x+1)$$

$$E[\text{len}(\text{path}(x))] = H_{x+1} + H_{n-x} - 2$$

$$E[\#\ i \text{ compare to pivot}] = H_{i+1} + H_{n-i}$$

$$E[\#\text{ of comparisons}] = \sum_{i=0}^{n-1} H_{i+1} + H_{n-i} - 2$$

$$= \underbrace{H_1 + H_2 + \dots H_n}_{} + \underbrace{H_1 + H_2 + \dots H_n}_{}$$

$$= 2\sum_{i=1}^{n} H_i$$

$$\le 2\sum_{i=1}^{n} H_n$$

$$= 2nH_n$$

$$\le 2n(\ln(n)+1)$$

$$= 2n\ln(n) + O(n)$$