CSE 102 1-30-20

Exercise

Write an algorithm that
Given an array

$A = (A_1, \ldots, A_n)$

determine # of inversions in A.

Defn An inversion is a pair (i,j) with $1 \le i < j \le n$ with $A_i > A_j$

Inversions (A, p, r)

		# COMP
1.	if $p < r$	0
2.	$q = \lfloor \frac{p+r}{2} \rfloor$	0
3.	$a =$ Inversions(A, p, q)	$T(\lceil \frac{n}{2} \rceil)$
4.	$b =$ Inversions(A, q+1, r)	$T(\lfloor \frac{n}{2} \rfloor)$
5.	$c =$ Between(A, p, q, r)	$\lceil \frac{n}{2} \rceil \cdot \lfloor \frac{n}{2} \rfloor$

Between (A, p, q, r)

```
1. count = 0
2. for i = p to q
3.     for j = q+1 to r
4.         if A[i] > A[j]
5.             count++
6. return count
```

$$\# \text{ comparisons by } \overset{A}{\text{Between}}\left(1, \underset{\overset{\shortparallel}{q}}{\left\lfloor \frac{1+n}{2} \right\rfloor}, n\right)$$

$$= \text{length}(A[1 \ldots q]) \cdot \text{length}(A[q+1 \ldots n])$$

$$= (q-1+1) \cdot (n-(q+1)+1)$$

$$= q \cdot (n-q)$$

note: $q = \left\lfloor \frac{1+n}{2} \right\rfloor = \left\lceil \frac{n}{2} \right\rceil$

$$= \left\lceil \frac{n}{2} \right\rceil \cdot \left(n - \left\lceil \frac{n}{2} \right\rceil\right)$$

$$= \left\lceil \frac{n}{2} \right\rceil \cdot \left\lfloor \frac{n}{2} \right\rfloor$$

let $T(n)$ = # of comp. By Inv.

$$T(n) = \begin{cases} 0 & (n=1) \\ T\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + T\left(\left\lceil \frac{n}{2} \right\rceil\right) + \left\lfloor \frac{n}{2} \right\rfloor \cdot \left\lceil \frac{n}{2} \right\rceil & (n \geq 2) \end{cases}$$

For master thm, simplify

$$T(n) = 2T\left(\frac{n}{2}\right) + n^2$$

comp. n^2 to $n^{\log_2 2} = n$

winner

by case 3: $T(n) = \Theta(n^2)$

Analysis of MergeSort(A, 1, n)

$$T(n) = \begin{cases} 0 & (n = 1) \\ T\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + T\left(\left\lceil \frac{n}{2} \right\rceil\right) + (n-1) & (n \geq 2) \end{cases}$$

Simplify :

$T(n) = 2T\left(\frac{n}{2}\right) + n$

comp. n to $n^{\log_2 2} = n$

<u>case 2</u> : $T(n) = \Theta(n \log n)$.