

TE : Time Efficiency

Brute force

- Compares distances for all  $n(n-1)/2$

$$\frac{n^2 - n}{2}$$

Keep the Highest factor

- Brute force runs in time  $\boxed{\Theta(n^2)} = \bar{O}(n)$

---

Divide and Conquer

- sort halves

$$\frac{n}{2} \quad \frac{n}{2}$$

- solve for each half

- merge together

$n$

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

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

$$\boxed{T(n) = \Theta(n \log n)}$$

## Space Efficiency

Brute force      def BF A

$i = 0$        $\subset$  values B

Val = [ ]       $\cap$

$j = i + 1$       n E

$d = \dots$        $n \cdot n \cdot F$

$$\underbrace{A+B+C+D}_{k_1} + \underbrace{nE}_{k_2} + \underbrace{n^2F}_{k_3}$$

$$\underbrace{\quad}_{k_1} + n \underbrace{\quad}_{k_2} + n^2 \underbrace{\quad}_{k_3}$$

Increases

$O(n^2)$

---

## Divide conquer

def Rec A  
  C(in<sub>1</sub>, in<sub>2</sub>) B

  P<sub>1</sub> C

  P<sub>r</sub> D

  Q<sub>1</sub> E

  Q<sub>r</sub> F

  while

    P<sub>1</sub> nG

    Q<sub>1</sub> nH

    P<sub>r</sub> nI

    Q<sub>r</sub> nJ

  while

    while (k <= nam)  $n^2 \cdot k$

$O(n^2)$