

Algorithms

* review runtimes for access of different data structures *

```

01 closestpair( $p_1, \dots, p_n$ ) : array of 2D points
02   best1 =  $p_1$ 
03   best2 =  $p_2$ 
04   bestdist =  $\text{dist}(p_1, p_2)$ 
05   for  $i = 1$  to  $n$ 
06     for  $j = 1$  to  $n$ 
07        $\text{newdist} = \text{dist}(p_i, p_j)$ 
08       if ( $i \neq j$  and  $\text{newdist} < \text{bestdist}$ )
09          $\text{best1} = p_i$ 
10          $\text{best2} = p_j$ 
11          $\text{bestdist} = \text{newdist}$ 
12   return ( $\text{best1}, \text{best2}$ )

```

Handwritten annotations:
 - $O(1)$ in pink next to lines 02-04.
 - $\sim n$ times in blue next to the outer for loop (05-11).
 - $\sim n$ times in blue next to the inner for loop (06-11).
 - $O(1)$ in blue next to the if statement (08-11).
 - $O(1)$ in blue next to the assignment statements (09-11).
 - $O(1)$ in blue next to the return statement (12).
 - \rightarrow Any two points in pink pointing to the $\text{dist}(p_i, p_j)$ call.
 - \star in yellow next to the for loop headers.

02-04: $O(1)$

05-11: $O(n \cdot n \cdot 1) = O(n^2)$

12: $O(1)$

$O(1 + n^2 + 1) = O(n^2 + 2) = O(n^2)$

```

01 mergesort( $L = a_1, a_2, \dots, a_n$ : list of real numbers)
02   if ( $n = 1$ ) then return  $L$  ①
03   else
04      $m = \lfloor n/2 \rfloor$ 
05      $L_1 = (a_1, a_2, \dots, a_m)$ 
06      $L_2 = (a_{m+1}, a_{m+2}, \dots, a_n)$ 
07     return merge(mergesort( $L_1$ ), mergesort( $L_2$ ))

```

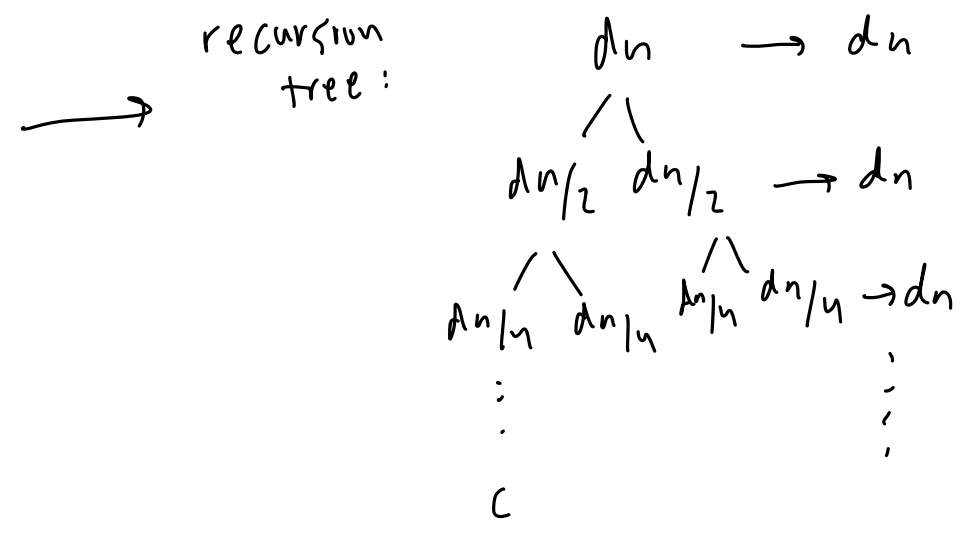
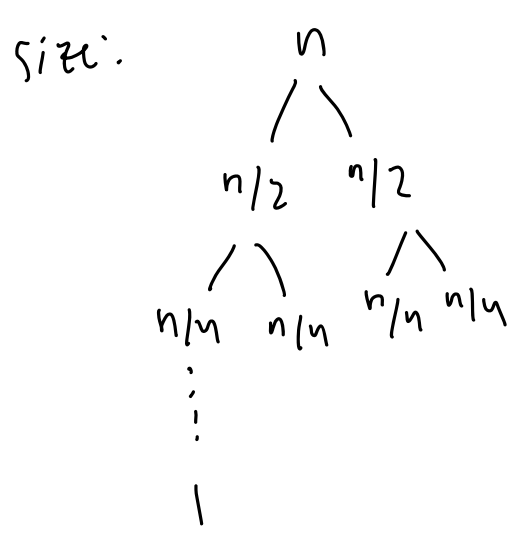
$O(1)$
 $O(n)$ ②

① $T(1) = c$
 $T(n) = 2T(n/2) + \underbrace{O(n)}_{\text{extra work}}$ ③

$T(n) = 2T(n/2) + dn$

③ $O(n)$

$T(1) = c$
 $T(n) = 2T(n/2) + \underline{dn}$



total work: $dn(\log_2 n) + c(2^{\log_2 n})$
 $dn \cdot \log_2 n + cn$

$$O(n \log_2 n) = \underline{O(n \log n)} \rightarrow O(n^2)$$

$$\Theta(n \log n)$$