Today: 1:15 - 2:15pm ET

Homework 3 & 4: Write solutions soon after you solve the problems.

Counting Inversions:

Input: Array $A$ of $n$ distinct integers.

Output: $\# \text{ inversions in } A$.


Example

$$
\begin{array}{cccccc}
  & 1 & 2 & 3 & 4 & 5 \\
A & 18 & 9 & 4 & 11 & 12 \\
\end{array}
$$

$\# \text{ inv } = 5$.

Aryan's algorithm: Go through each pair of nouns in $A$. 
and increment the count if the pair is inverted.

Running time: $\Theta(n^2)$

Target runtime: $\Theta(n \log n)$.

The runtime recurrence that gives us $\Theta(n \log n)$ runtime is the MergeSort recurrence

$$T(n) = 2T\left(\frac{n}{2}\right) + \text{Constant}.$$
\# inv = \# inv + 4

Sort and Count \((A[1..n])\)

\[
\text{if } n = 1 \text{ then } \\
\quad \text{return } (A, 0)
\]

\[
\text{mid } \leftarrow \left\lfloor \frac{1 + n}{2} \right\rfloor \\
\quad \text{sorted left half} \\
\quad \text{unsorted left half}
\]

\([\quad]) \quad O(1)

\([\quad]) \quad O(1)
\[(x, i_1) \leftarrow \text{Sort and Count} \ (A [1 \ldots \text{mid}]) \quad T(n/2)\]

\[(y, i_2) \leftarrow \text{Sort and Count} \ (A [\text{mid}+1 \ldots n]) \quad T(n/2)\]

\[(z, i_3) \leftarrow \text{Merge and Count} \ (x, y) \quad O(n)\]

\[\text{return} \ (z, i_1 + i_2 + i_3) \quad O(1)\]

\[T(n) = 2T(n/2) + O(n)\]

\[= \Theta(n \log n)\]

\[\text{Merge and Count} \ (x, y)\]

\[l \leftarrow 1, \ r \leftarrow 1, \ q \leftarrow 1, \ \# \text{inv} \leftarrow 0\]

\[\text{while } l \leq |x| \text{ and } r \leq |y| \text{ do}\]

\[\text{if } x[l] < y[r] \text{ then}\]

\[z[q] \leftarrow x[l]\]

\[l \leftarrow l+1, \ q \leftarrow q+1\]

\[\text{else}\]

\[\# \text{inv} \leftarrow \# \text{inv} + |x| - l + 1\]

\[z[q] \leftarrow y[r]\]
\[ r \leq r+1, \quad q \leq q+1 \]

Append the non-empty array to \( Z \).

output \((Z, \text{#inv})\).

**Closest Pair**

**Input:** \( n \) points \((\text{set P})\) in a plane.
- each pt has \( x \) \& \( y \) coordinates.

**Output:** Closest pair \( q \) pts.

\[
\sqrt{(x_2-x_1)^2 + (y_2-y_1)^2}.
\]

\((x_1, y_1)\) \quad \rightarrow \quad (x_2, y_2)\)

**Agatha's alg.**

- Check every pair of pts within
dist formula.

Runaway time: $\Theta\left(n^2\right)$

Target runtime: $\Theta\left(n \log n\right)$

Target Recurrence

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

Let \( \delta \) be the distance between two points.

Given \( \delta \), we have \( \frac{\delta}{\sqrt{2}} < \delta \).

Proof:

0. Sort points \( m \) by their \( x \)-coordinates.
    Let the points be \( X \) and \( Y \).
    All the points \( m \) are in the \( x \)-cood.

1. Divide the points into two halves \( L \) and \( R \).
   Let \( P_L \) and \( P_R \) be the points in the two half-plane.
   Let \( \mathcal{L} \) be the dividing line (median) in the \( x \)-coordinate.
2. \( \delta_L \leftarrow CP(PL) \) \( T(n^{1/2}) \rightarrow \) By IH

3. \( \delta_R \leftarrow CP(PR) \) \( T(n^{1/2}) \rightarrow \) By IH

4. \( \delta \leftarrow \min \{ \delta_L, \delta_R \} \) \( O(1) \)

5. Consider the pts in the strip of width 2\( \delta \) around the line \( l \). Let \( S \) be the pts. \( O(n) \)

6. Process the pts in \( S \) in \( \uparrow \) order \( \downarrow \)

   \[
   \text{their } y \text{-coordinates. For each pt. } p \\
   \text{Compare } p \text{ with } 2 \text{ pts. after it. Update } \delta \text{ as needed. } O(n) \leq O(n + n \log n) \leq O(n^2) \]

7. Output \( \delta \) and the associated pair of pts. \( O(n) \).
$$T(n) = 2T(n/2) + n.$$ 

$$T(n) = 2T(n/2) + n\log n$$

$$= n\log^2 n.$$

**Recurrence of CP(n) (Steps 1-7):**

$$T(n) = 2T(n/2) + n = O(n\log n)$$

if $$n \leq 3$$ then
do bottom.