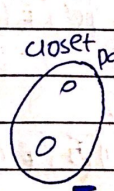
Brute Force Algorithm..

The Brute force algorithm is all about trial & error letting a Computer Work systematically through a range of possibilities untill it arrives at the correct solution.

→ e.g; Padlock with 4 digit. 0 to 9

Closet pair problem- In metric space

→ find a pair of points with the smallest distance b|w them-



Example To find Closet pair Using brute force algorithm:

Points = (1,2), (4,6), (7,8), (2,9), (5,3)

Calculate Distance: $\because \sqrt{(x_2-x_1)^2+(y_2-y_1)^2}$

→ Distance b|w $(\overset{x_1}{1},\overset{x_2}{2})$ & $(\overset{y_1}{4},\overset{y_2}{6})$

$= \sqrt{(4-1)^2+(6-2)^2} = \sqrt{(3)^2+(4)^2} = 5$

→ Distance : $(\overset{x_1}{1},\overset{x_2}{2})$ & $(\overset{y_1}{7},\overset{y_2}{8})$

$= \sqrt{(7-1)^2+(8-2)^2} = \sqrt{(6)^2+(6)^2} = 6\sqrt{2}$

→ Distance = $(\overset{x_1}{1},\overset{x_2}{2})$, $(\overset{y_1}{2},\overset{y_2}{9})$

$= \sqrt{(2-1)^2+(9-2)^2} = \sqrt{(1)^2+(7)^2} = 5\sqrt{2}$

M T W T F S

Distance = (x_1, y_1) & (x_2, y_2) (1,2) & (2,9)

$= \sqrt{(2-1)^2 + (9-2)^2} = \sqrt{(7)^2 + (1)^2} = 5\sqrt{}$

Distance = (1,2), (5,3)

$= \sqrt{(5-1)^2 + (3-2)^2} = \sqrt{(4)^2 + (1)^2} = \sqrt{}$

Now: (4,6) & (7,8)

$= \sqrt{(7-4)^2 + (8-6)^2} = \sqrt{(3)^2 + (2)^2} = \sqrt{}$

→ (4,6) & (2,9)

$= \sqrt{(2-4)^2 + (9-6)^2} = \sqrt{(7)^2 + (2)^2} = \sqrt{}$

→ (4,6) & (5,3)

$= \sqrt{(5-4)^2 + (3-6)^2} = \sqrt{(1)^2 + (3)^2} = \sqrt{}$

Now; (7,8) & (2,9)

$= \sqrt{(2-7)^2 + (9-8)^2} = \sqrt{(5)^2 + (1)^2} = \sqrt{}$

(7,8) & (5,3)

$= \sqrt{(5-7)^2 + (3-8)^2} = \sqrt{(2)^2 + (5)^2} = \sqrt{29}$

Now; (2,9) & (5,3)

$= \sqrt{(5-2)^2 + (3-9)^2} = \sqrt{(3)^2 + (6)^2} = 18$

So; the smallest distance is between (4, 6) & (5,3).

Time Complexity = $O(N^2)$.

Date: __/__/20

Complexity analysis of Close Pair:

The time Complexity of ucledian distance algorithm depends on the dimensionality of the space in which the distance are being Calculated.

Euclidean distance b/w two point=
$(P_1, P_2, P_3 \cdots P_n)$ & $(q_1, q_2 q_3 \cdots q_n)$

$$= \sqrt{(q_1 - P_1)^2 + (q_2 - P_2)^2 + \cdots (q_n - P_n)^2}$$

Operations of ucledian:

→ Difference Compute b/w Corresponding point.
→ Square of difference.
→ Sum of squared difference
→ Take the square root of sum.

Each operation is constant time, so the overall Complexity of b/w two Points in n-dimentional Space is $O(N)$.

→ If you have m pair of points so Complexity is $O(N*m)$.

M T W T F S

Date: __/__/2

Convex hull - Problem:

A set of points is convex if for Point, P & Q, the entire line seg PQ is in the set-

They are used for detecting outlier by some statistical techniques.
Also used in Collision detection-

What is polygon in Convex H

A polygon is a region of the plo bounded by a cycle of line seg Joined end-to-end.

Two types of polygon:

→ Regular polygon: Equal length of all si
→ Irregular polygon: Unequal length sid

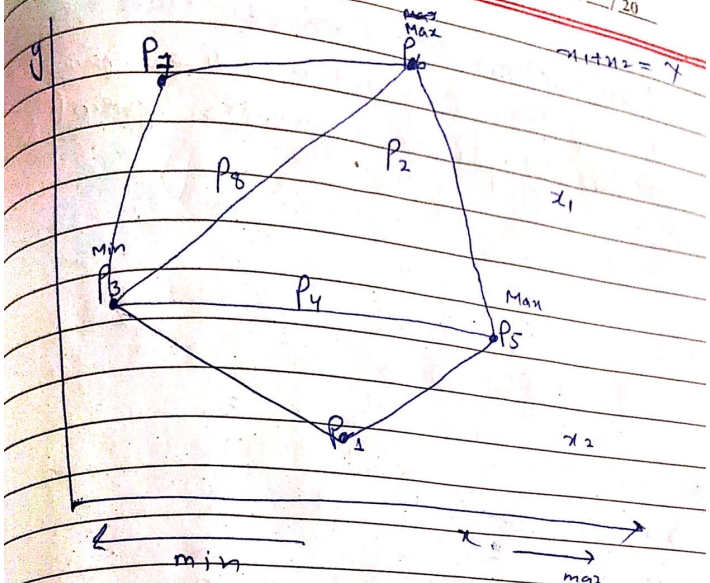
- Identical angle at each vertex.
- Angle differ from one another.

Diff b/w Convex & Concave pol

Each angle $< 180^\circ$ called Convex.
$> 180^\circ$ called concave.

clockwise

Date: __/__/20



Inter point of polygon are = P_8, P_2, P_4

line segment / outies are

P_1, P_5, P_6, P_7, P_3

Extreme point in Polygon-

Extreme point of a Convex set is a point of this set that is not a middle point of any line segment with end points in the set.

→ that does not lie on any open line segment.

M T W T F S

Date: __/__/ 2(

Application of Conver Hull -

→ Computer Visualization / Video games.

→ Geographical information system (G

A

B

Date: __/__/20

Dynamic Programming:

dynamic programming is a powerful design technique that can be used to solve Problems of a Computational nature.

→ It refer to the tabular method.

→ It is a technique where an algorithmic Problem is first broken down into Sub-Problems, the result are saved & than sub Problems are optimized to find overall solution.

"Programming" here means "planning".

Main Idea:

→ set up recurrence Relation.

→ Solve Smaller instances.

→ Record Solution in a table.

→ extract solution.

Element of DP ① (overlapping sub Problem)

→ Split Problem into sub Problem as similar as divide & conquer.

→ It is found in that where bigger problem share the same smaller Problem.

② Optimal sub Structure:-

It is an optimal selection among all the possible substructures that can help to select the best