Divide & Conquer - II

Integer Multiplication

- Input : Two n - bit integers x and y
- Output : Product of x and y

Integer Multiplication

- Input : Two n - bit integers x and y
- Output : Product of x and y
- Bruteforce method - $O(n^2)$

Integer Multiplication

- Input : Two n - bit integers x and y
- Output : Product of x and y

$$xy = x_1y_12^n + x_0y_0 + [(x_1 + x_0)(y_1 + y_0) - x_1y_1 - x_0y_0]2^{\frac{n}{2}}$$

Integer Multiplication

- Input : Two n - bit integers x and y
- Output : Product of x and y

$$xy = x_1y_12^n + x_0y_0 + [(x_1 + x_0)(y_1 + y_0) - x_1y_1 - x_0y_0]2^{\frac{n}{2}}$$

$$T(n) = 3T(\frac{n}{2}) + n$$

$$= n^{\log_2 3} = n^{1.59}$$

Integer Multiplication

- Recursive_Multiply(x,y)

$$x = x_1 2^{\frac{n}{2}} + x_0$$

$$y = y_1 2^{\frac{n}{2}} + y_0$$

$$p = \texttt{Recursive\_Multiply}(x_1 + x_0, y_1 + y_0)$$

$$q = \texttt{Recursive\_Multiply}(x_1, y_1)$$

$$r = \texttt{Recursive\_Multiply}(x_0, y_0)$$

$$\texttt{return}\ \ q2^n + (p - q - r)2^{\frac{n}{2}} + r$$

Closest Pair of Points

- Input : P, set of n points in the plane
- Output : p,q in P that minimises dist(p,q)

Closest Pair of Points

- Input : P, set of n points in the plane
- Output : p,q in P that minimises dist(p,q)

Points on the real line?

Closest Pair of Points

- Input : P, set of n points in the plane
- Output : p,q in P that minimises dist(p,q)

Brute force ?

Closest Pair of Points

- Input : P, set of n points in the plane
- Output : p,q in P that minimises dist(p,q)

Divide and Conquer

Solve for n/2 points on the left and n/2 points on the right

Closest Pair of Points

- Input : P, set of n points in the plane
- Output : p,q in P that minimises dist(p,q)

Divide and Conquer

Solve for n/2 points on the left and n/2 points on the right

- How is the solution related to the solutions of the sub-problems?

Closest Pair of Points

- Input : P, set of n points in the plane
- Output : p,q in P that minimises dist(p,q)

Divide and Conquer

Solve for n/2 points on the left and n/2 points on the right

- How is the solution related to the solutions of the sub-problems?

Closest Pair of Points

(q_0, r_0) - closest pair of points among n/2 points in the left

(q_1, r_1)-closest pair of points among n/2 points in the right

d= min (dist(q_0, r_0), dist(q_1, r_1))

Closest Pair of Points

(q_0, r_0) - closest pair of points among n/2 points in the left

(q_1, r_1)-closest pair of points among n/2 points in the right

d= min (dist(q_0, r_0), dist(q_1, r_1))

The solution is either (q_0, r_0) OR (q_1, r_1) OR some (q', r') such that q' is in left and r' is in right and dist(q', r') $\leq d$

Closest Pair of Points

The solution is either (q_0, r_0) OR (q_1, r_1) OR some (q', r') such that q' is in left and r' is in right and dist(q', r') $\leq d$

Let L be the line that divides P into two

Let S be the set of points in P that lie within distance d from L

$q', r' \in S$

Closest Pair of Points

Let S_y be S sorted by y co-ordinates

Claim : If there exists s and s′ in S such that dist(s, s′) < d, then s and s′ are within 15 positions of each other in S_y

- Closest_Pair(P)

P_x - P sorted by x-co-ordinate

P_y - P sorted by y-co-ordinate

return Closest_Pair_Rec(P_x,P_y)

- Closest_Pair_Rec(P_x,P_y)

```
If |P|<3 then solve by brute force
```

Construct Q_x, Q_y, R_x, R_y

(q_0, q_1) = Closest_Pair_Rec(Q_x, Q_y)

(r_0, r_1) = Closest_Pair_Rec(R_x, R_y)

d= min (dist(q_0, q_1), dist(r_0, r_1))

Let S be the set of points in P that lie within distance d from L

Construct S_y

For each point in S_y, compute distance from next 15 points in S_y

Find the minimum of all distances - (s, s')

If min < d, return (s, s')

else

    if dist(q_0, q_1)=d, then return (q_0, q_1)

    else return (r_0, r_1)