COP 3503 - 3/31/22

Divide + Conquer

(1) Closest Pair of Pts

(2) Strassen's Alg (Matrix Mult)

Int Mult

if you break in halves,
4 mult of size n/2

↳ Clever algebra ⇒ 3 mult more add/sub.

Strassen's Matrices

$$\left[\begin{array}{c|c} A_0 & A_1 \\ \hline A_2 & A_3 \end{array}\right]\left[\begin{array}{c|c} B_0 & B_1 \\ \hline B_2 & B_3 \end{array}\right] = \begin{bmatrix} A_0B_0 + A_1B_2 & A_0B_1 + A_1B_3 \\ A_2B_0 + A_3B_2 & A_2B_1 + A_3B_3 \end{bmatrix}$$

Review Reg Algs

standard time to mult 2 nxn matrices

$O(n^3)$

```
for(i=0; i<n; i++)
  for(j=0; j<n; j++)
    for(k=0; k<n; k++)
      ans[i][j] += a[i][k] * b[k][j];
```

Strassen's Analysis

$$T(n) = 8T\left(\frac{n}{2}\right) + O(n^2)$$

$A = 8$, $B = 2$, $k = 2$

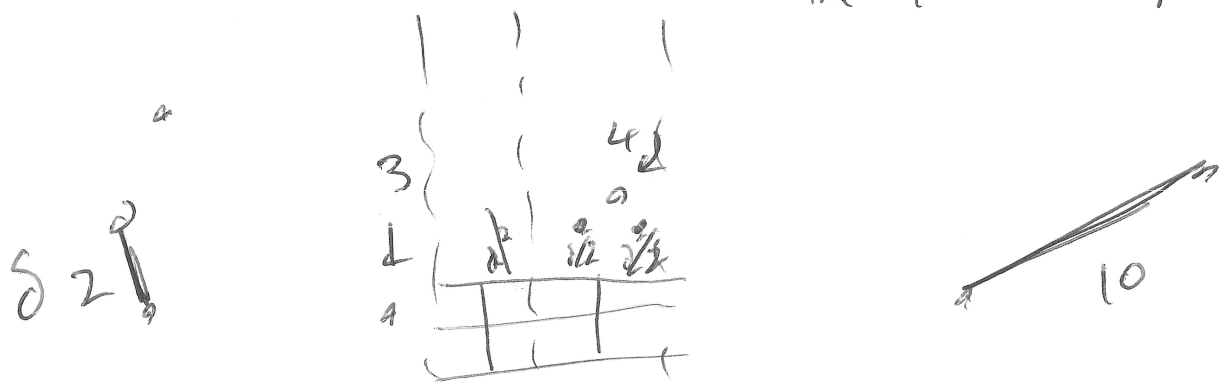
$A > B^k \rightarrow O(n^{\log_2 8}) = O(n^3)$

→ Strassen's goal - use crazy algebra to bring this down to 7 recursive mult.

TYPED NOTES HAVE THESE PRODUCTS + HOW TO RECOMBINE TO GET THE 4 SUBANSWERS TO FORM THE PRODUCT

Closest Pair of Pts

n points
which 2 are closest. (either find distance or the actual 2 pts.)



(3) Take all pts within δ of the splitting line and resort by y.

(4) for pt[i] in sorted list check its distance w/ pts i+1, i+2, ... i+7. If any of these are less than δ update

Note: δ = smaller answer of 2 recursive calls

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

(Merge Sort Recurrence $\Rightarrow$ $O(n \lg n)$

Picture