Heap Sort - II

Complete Course on Algorithms - GATE

Subbarao Lingamgunta • Lesson 39 • May 9, 2023

Selection procedure

i/p: An array of n-distinct ele, integer - K

o/p: Find K^{th} smallest ele.

$SP(a, 1, 12, K)$

ex A

80	55	92	64	88	23	131	49	180	29	31	69
1	2	3	4	5	6	7	8	9	10	11	12

$K = \cancel{8}$

$\Downarrow$ partition(n) $\Longrightarrow O(n)$

$\overset{8}{m} = \left(\underset{1}{55}\ \underset{2}{64}\ \underset{3}{23}\ \underset{4}{49}\ \underset{5}{29}\ \underset{6}{31}\ \underset{7}{69}\right) \underset{8}{80} \left(\underset{9}{92}\ \underset{10}{88}\ \underset{11}{131}\ \underset{12}{180}\right)$

$\Downarrow SP(a, 1, 7, K)$

$\Downarrow$ p(n) $\Longleftrightarrow O(n)$

$\overset{5}{m} = \left(\underset{1}{23}\ \underset{2}{49}\ \underset{3}{29}\ \underset{4}{31}\right) \underset{5}{55} \left(\underset{6}{64}\ \underset{7}{69}\right)$

```
SP(a, i, j, k)                      => T(n)
{
  if(i==j) return(a[k])
  else
  {
    m = Partition(a, i, j)          => n
      if(m==k) return(a[k]);
      else
        if(k<m)
          SP(a, i, m-1, k)          => T(m-i)
        else
          SP(a, m+1, j, k);         => T(j-m)
  }
}
```

let $T(n) = TC$ of above algo

RR

$$T(n) = \begin{cases} O(1) & \text{if } n = 1 \\ n + T(n-i) \text{ (or) } T(j-n) & \text{if } n > 1 \end{cases}$$

Bestcase, Avgcase

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

$\Downarrow$ Sn

$= n + n/2 + n/2^2 + \dots + \frac{n}{2^k}$

$\Longrightarrow O(n)$

Worstcase

$T(n) = n + T(n-1)$

$\Downarrow$ n

$= 1 + 2 + 3 + \dots + n = O(n^2)$

Strassen's matrix multiplication

$c_{11} = a_{11} * b_{11} + a_{12} * b_{21} + a_{13} * b_{31} + a_{14} * b_{41}$

without - DAC

Matrix Addition

for (i = 1 to n)

for (j = 1 to n)

c[i][j] = A[i][j] ± B[i][j]

⇓

n^2 - ele

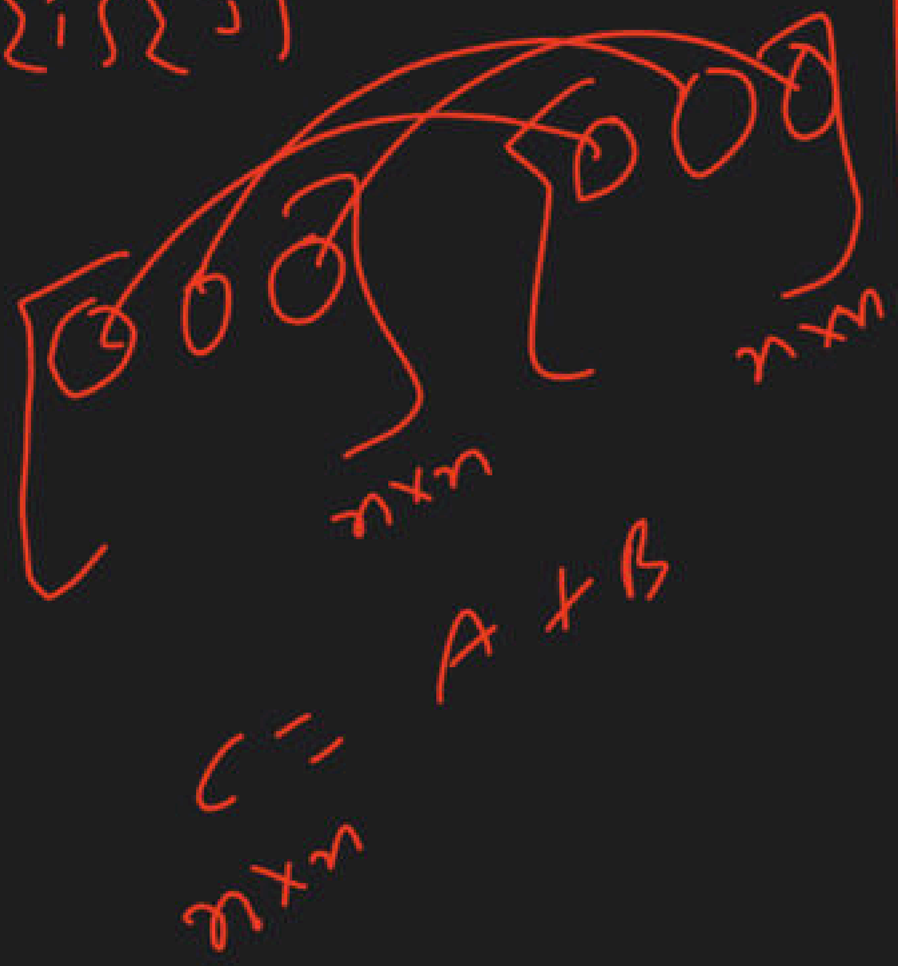
⇓

n^2 * 1 - add

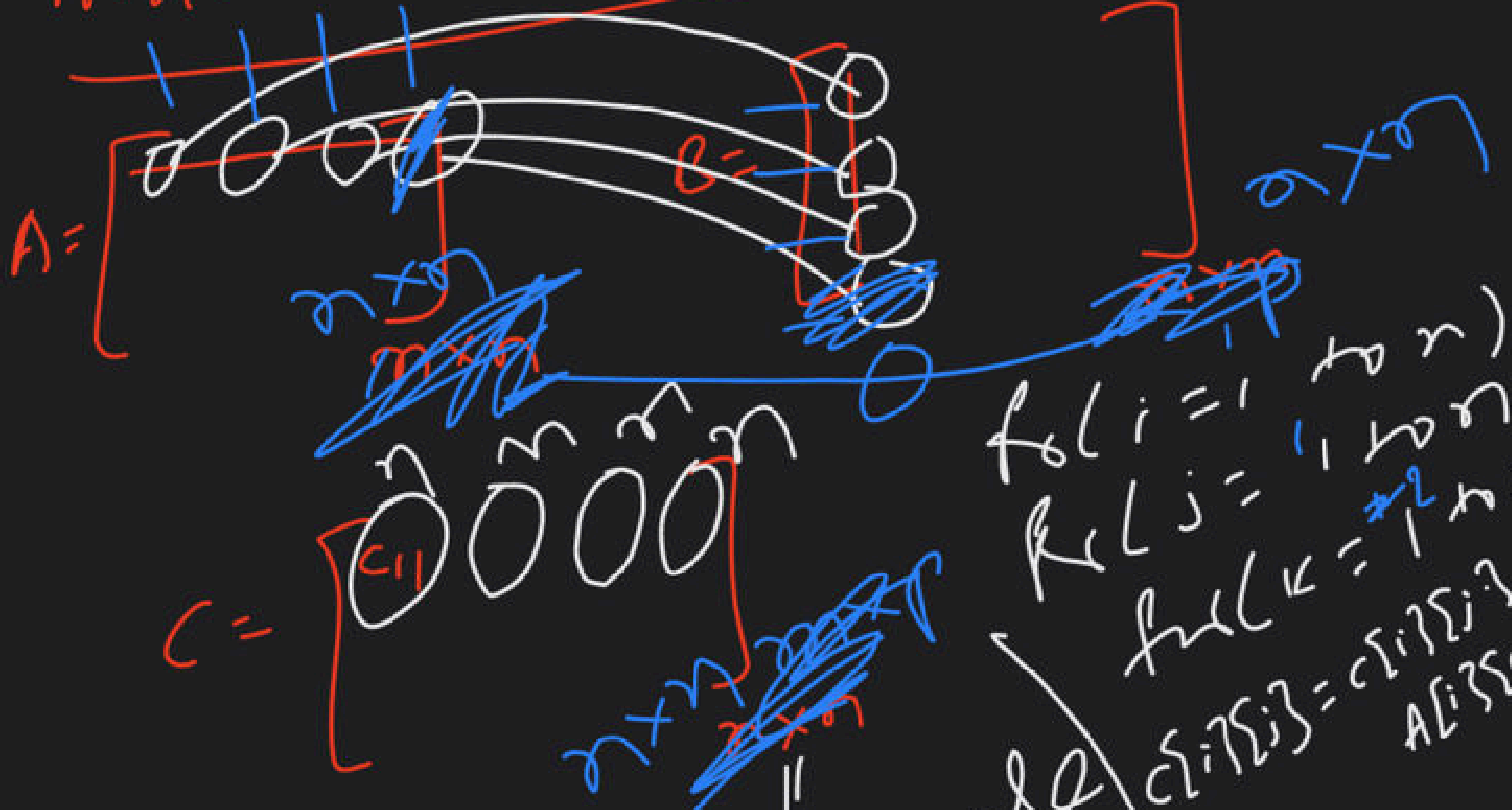
$n^2 * O(1)$

⇓

$O(n^2)$



Matrix multiplication



for (i = 1 to n)

for (j = 1 to n)

for (k = 1 to n)

c[i][j] = c[i][j] + A[i][k] * B[k][j]

⇓

n^2 * ele

⇓

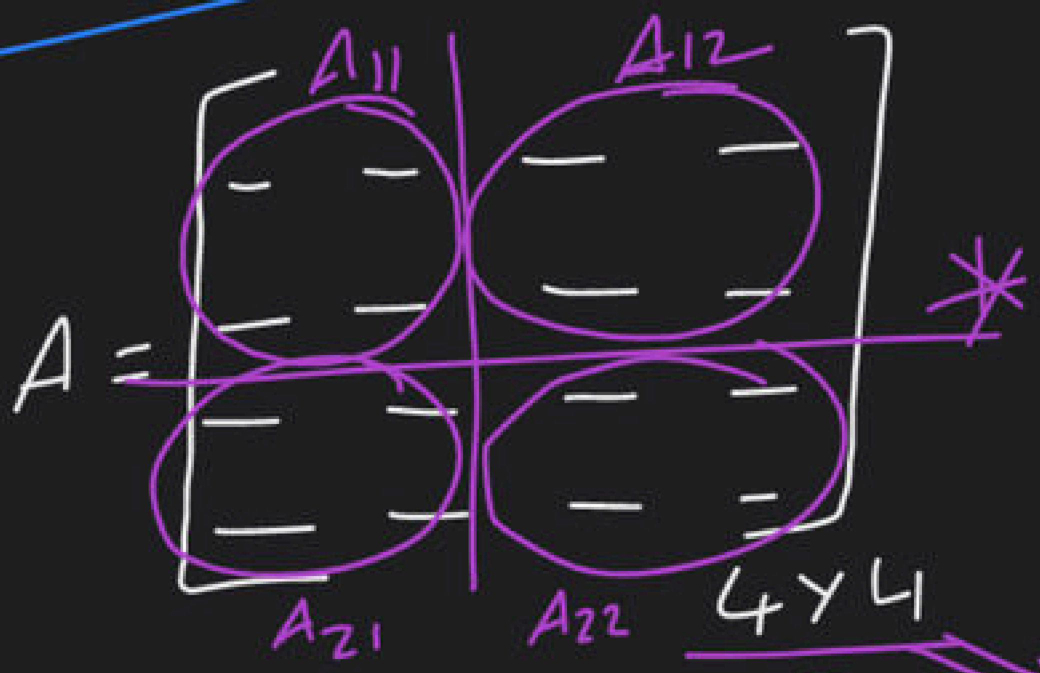
$n^2 * n$

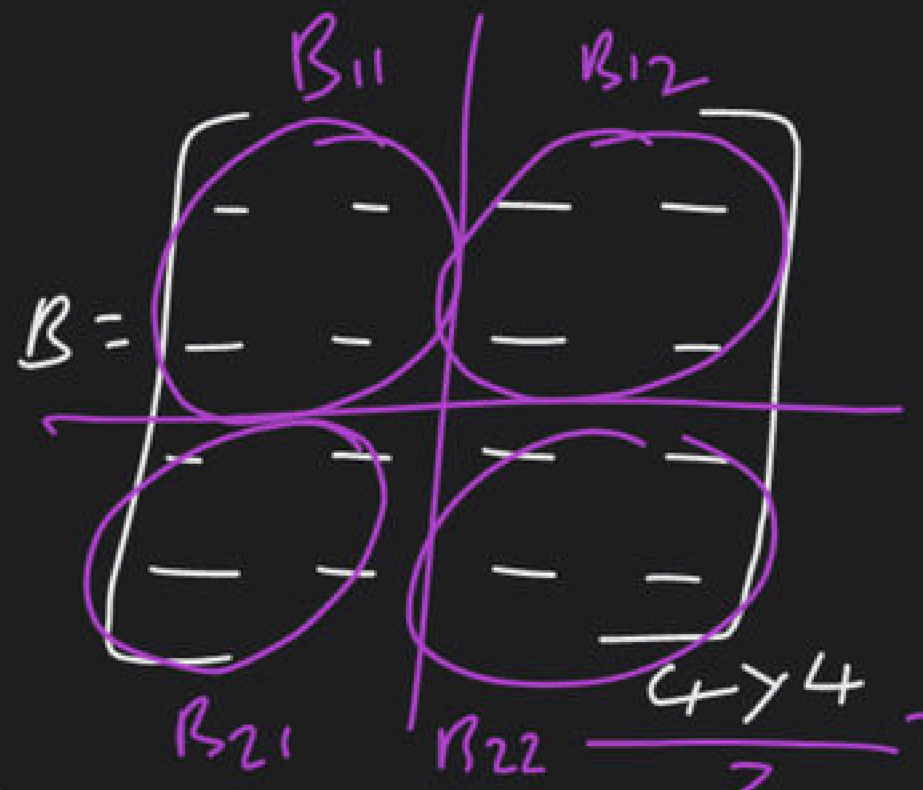
$O(n^3)$

With-DAC

Matrix-multiplication

ex

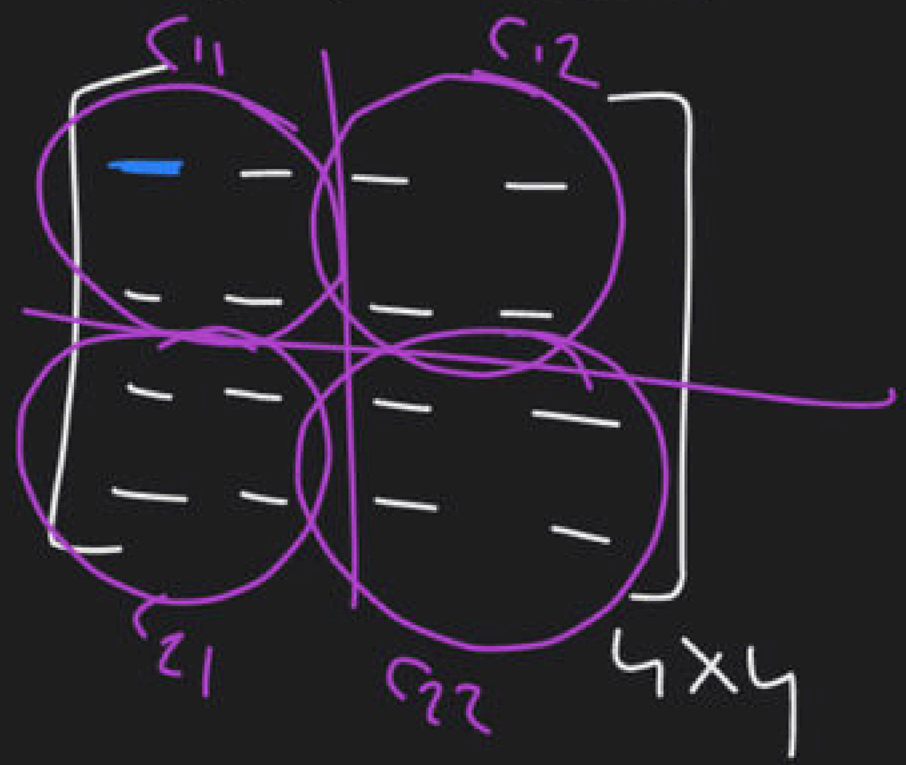




$\frac{4 \times 4}{2} \Rightarrow 2\times 2$ $\frac{4 \times 4}{2} = 2 \times 2$

$C = A \times B$

$4\times 4 \quad 4\times 4 \quad 4\times 4$



$C_{11} = A_{11} * B_{11} + A_{12} * B_{21}$

$C_{12} = A_{11} * B_{12} + A_{12} * B_{22}$

$C_{21} = A_{21} * B_{11} + A_{22} * B_{21}$

$C_{22} = A_{21} * B_{12} + A_{22} * B_{22}$

$T(n)$ = TC to multiply $n \times n$ matrices

$T(4) = 8T(4/2) + 4 * (4/2)^2$

$T(8) = 8T(8/2) + 4 * (8/2)^2$

$T(64) = 8T(64/2) + 4 * (64/2)^2$

RR

$$T(n) = \begin{cases} O(1) & \text{if } n \le 2 \\ 8T(n/2) + 4 * (n/2)^2 \end{cases}$$

$\Downarrow$

$T(n) = 8T(n/2) + n^2$

$\Rightarrow O(n^3)$

$$T(n) = 8T(n/2) + n^2 \Longrightarrow O(n^{\log_2 8})$$

$$T(n) = 7T(n/2) + 18(n/2)^2$$

$$= 7T(n/2) + 4.5n^2$$

$$= 7T(n/2) + n^2$$

$$= O(n^{\log_2 7}) \Longrightarrow O(n^{2.81})$$