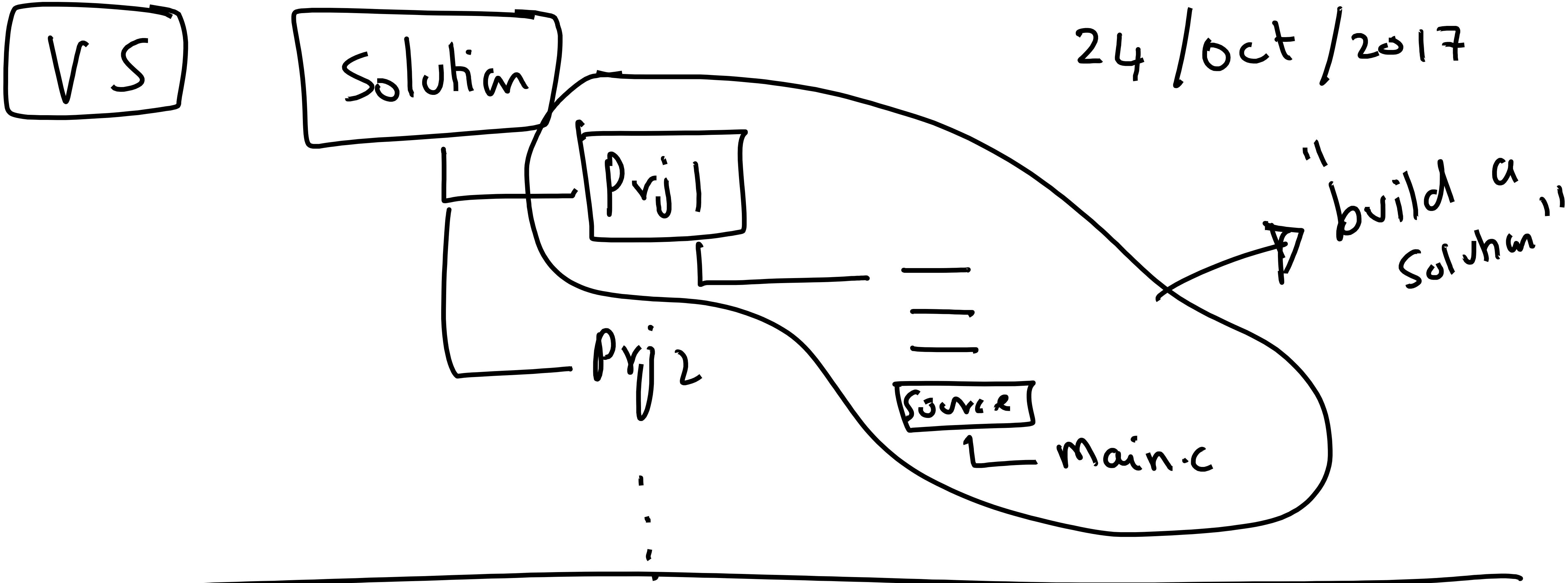


24/oct/2017

Syntax

printf (   ) ✓

scanf (   ) $\xrightarrow{\text{VS}}$ scanf_s (   )

strcpy (a, b) ⟼ scanf strcpy_s (   )

---

if - elseif - else

switch

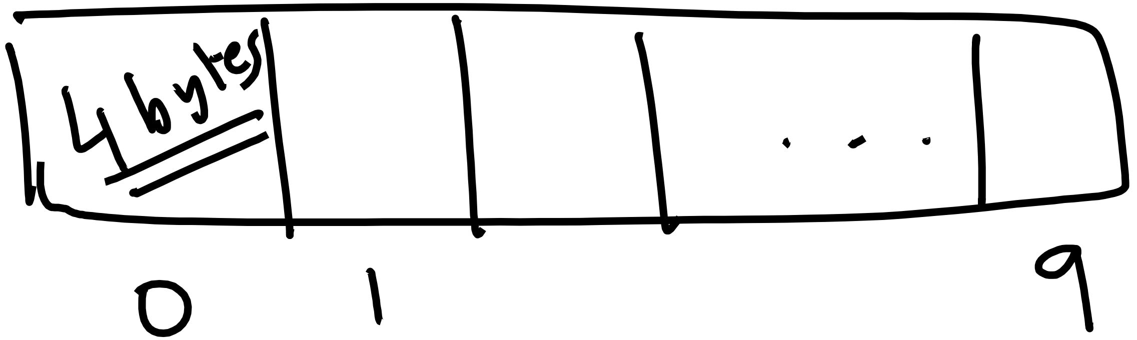
loops — while

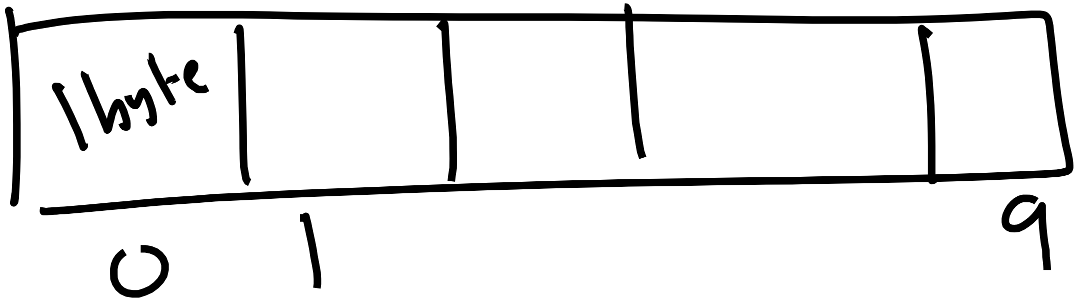
— do - while

— for

Arrays

```
int array[10];
float valves[10];
char name[10];
```

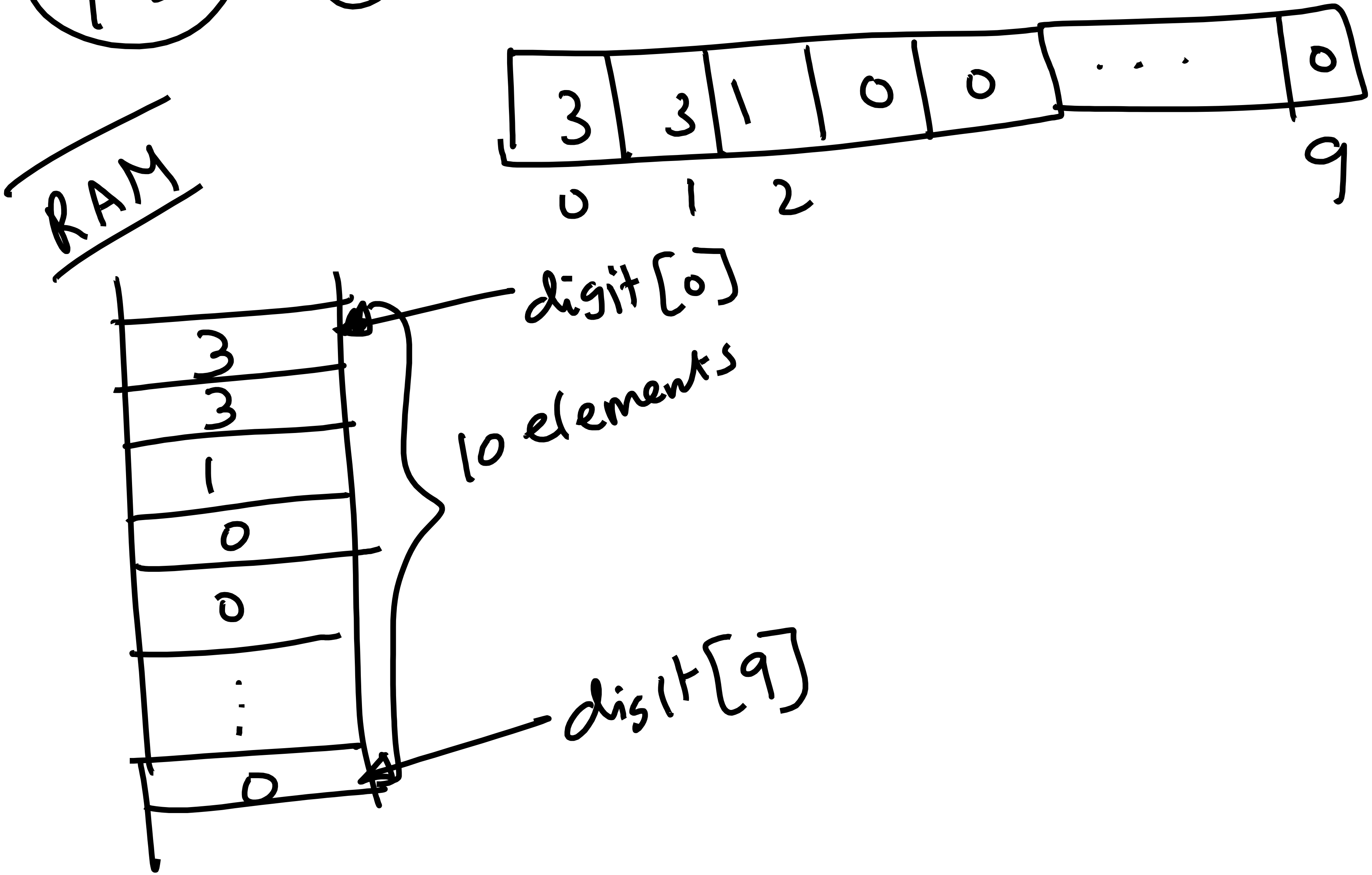




```
array[i] = array[i] + 10;
```

T3-1 a

```
int digits[10] = {3, 3, 1};
```



T3.2

```
float data[100], d[100], avg;
    ⋮
for (i=0; i<100; i++) {
    avg = avg+data[i];
 }
  avg = avg/100; /* avg/strlen?   */
for (i=0; i<100; i++) {
    d[i]= data[i]-avg;
```

try this!

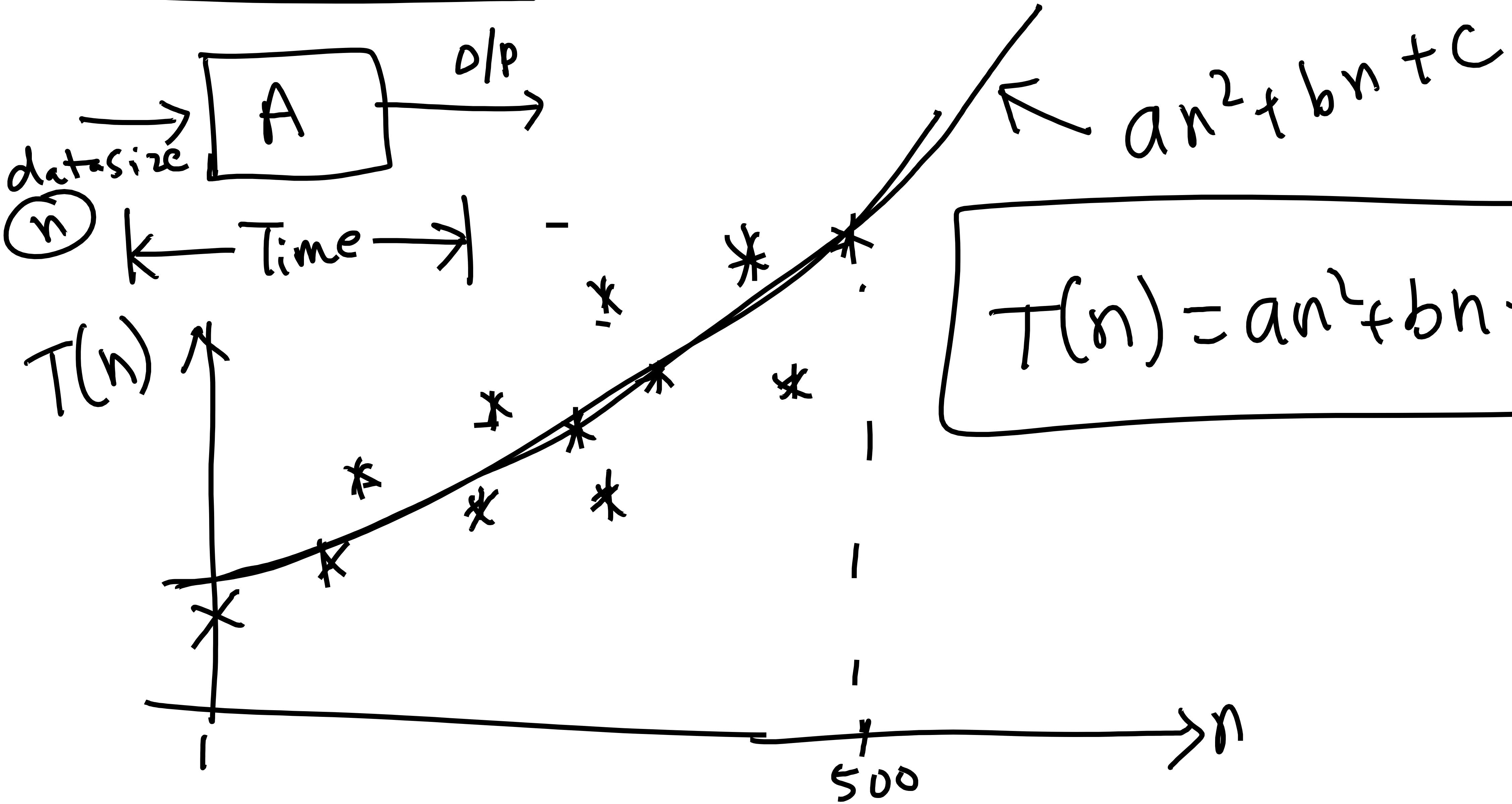
$$\texttt{float array}\underset{\substack{\downarrow\\ \text{row}}}{[10]}\underset{\substack{\searrow\\ \text{col}}}{[10]};$$

$$\begin{bmatrix} (0,0) & (0,1) & \cdots & (0,9) \\ (1,0) & & & (1,9) \\ \vdots & & & \\ (9,0), & (9,1), & \cdots & (9,9) \end{bmatrix}$$

array[3][4] = {1, 2, 3, 4, ... , 12};

$$\text{array} \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \end{bmatrix}$$

Time Complexity



$$\boxed{T(n)=an^2+bn+c}$$

$T(n) = an^2 + bn + c$ $\longrightarrow$ n^2 term dominates

n	$T(n)$
1	(a, b, c)
5	$(a5^2, b5)$
⋮	
500	$(500^2 \quad 500)$
⋮	
1mil	$1\text{mil}^2, 1\text{mil}$

$n^3 + n^2 + n + K$

$\boxed{\text{Big } O(\cdot)}$

Order of complexity is n^2

$O(n^2)$.

```
→ for (i=0; i < 100 ; i++) {          O(n²)
                  n
      for (j=0; j<100; j++) {
                    n
              a[i][j] = a[i][j] + 1;

      }
  }
```

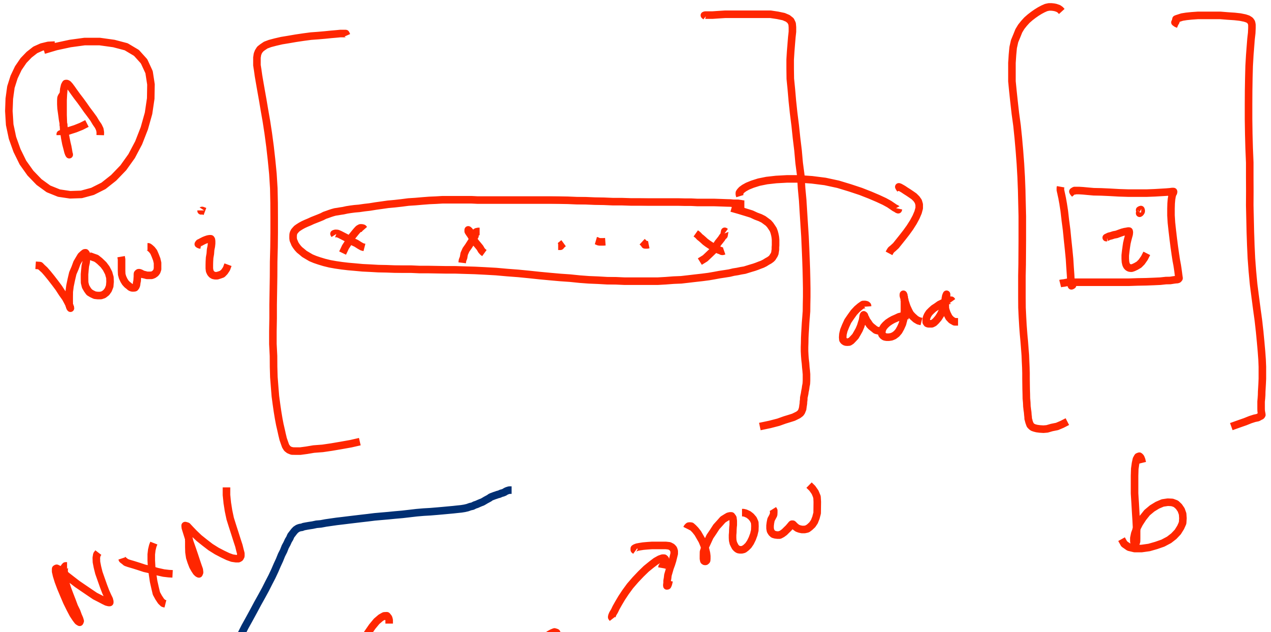
$O(n^2)$

→ i — 100

for a fixed i j runs 100x

& that j runs 100x

100 x 100 = 10 000



$$b[i] = \sum_{j=0}^{N-1} a[j]$$

row

```
for (i=0; i<N; i++){
    sum=0;
    for(j=0; j<N; j++)
        sum= a[i][j]+sum;
    }
    b[i]=sum;
}
```

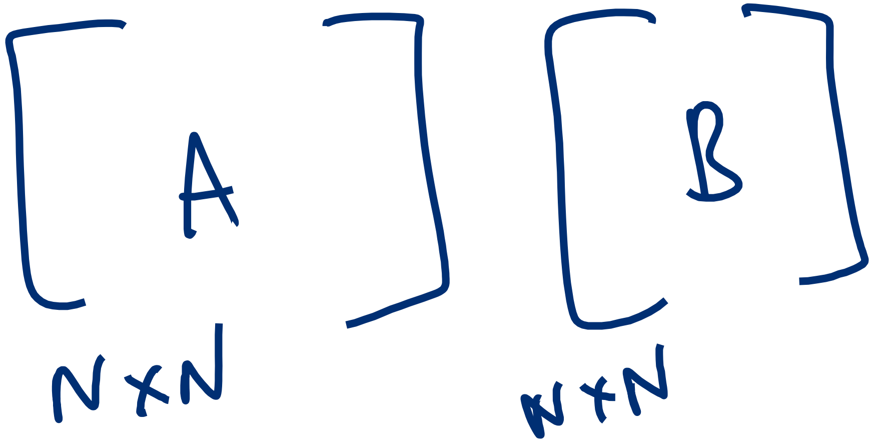
col

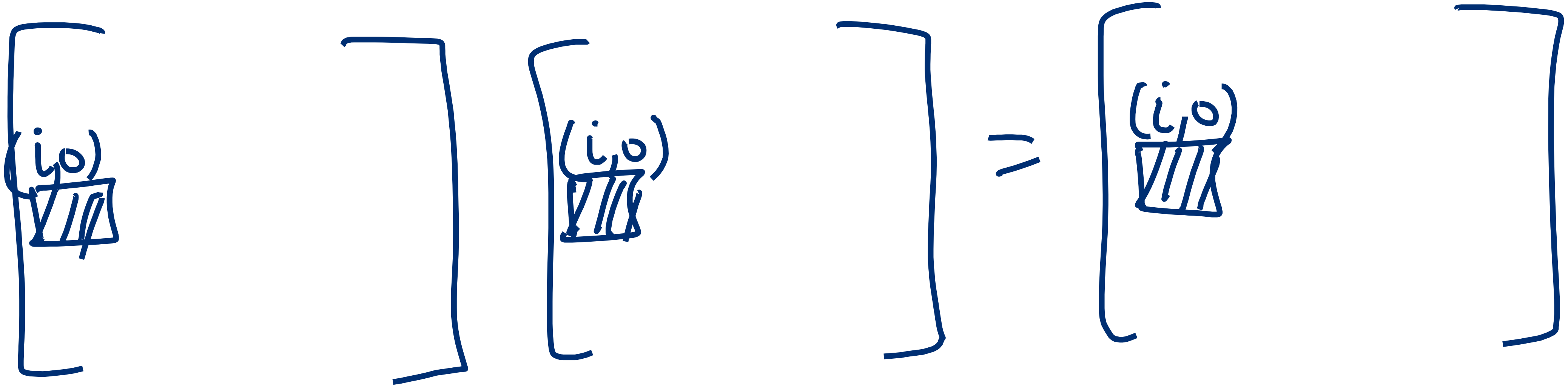
Time Complexity?

$O(N^2)$

```
b[i]=b[i]+
    a[i][j];
```

B





a[i][0] + b[i][0] => c[i][0]

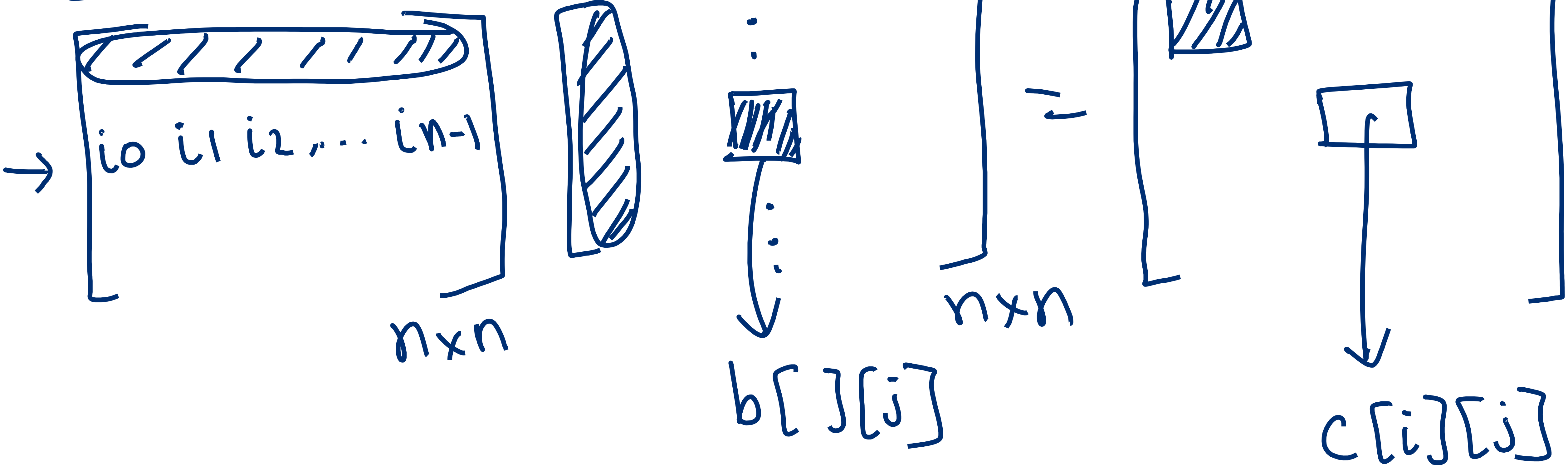
```
for (i=0; i<N; i++){
    for(j=0; j<N; j++){
        c[i][j]= a[i][j]+b[i][j];
    }
}
```

Time Complexity ?

$O(N^2)$

```
int a[20][20], b[20][20],
    c[20][20];
```

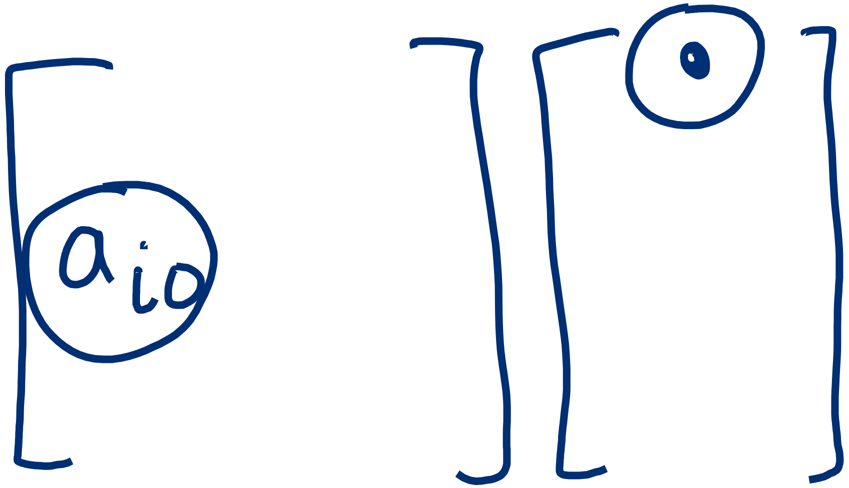
C mat. mult



$$a_{i0} b_{0j} + a_{i1} b_{1j}$$

$$+ a_{i2} \cdot b_{2j} + \cdots$$

$$+ a_{ik} \cdot b_{kj} + \cdots a_{in-1} \cdot b_{n-1j} = c_{ij}$$



$$\boxed{c_{ij} = \sum_{k=0}^{n-1} a_{ik} \cdot b_{kj}}$$

$i = 0, 1, \ldots, n-1$

$j = 0, 1, \ldots, n-1$

```
for(i=0; i<N; i++){   /* row */
    for(j=0; j<N; j++){   /* col */
        for(k=0; k<N; k++){
            c[i][j]= c[i][j] +
                a[i][k]*b[k][j];
        }
    }
}
```

$$\boxed{\text{Time complexity} = O(N^3)}$$

Space Complexity

Ⓐ $\underline{\underline{4}}n^2 + \underline{\underline{4}}n = \overset{\text{integer size}}{\underline{\underline{4}}}(n^2+n) \rightarrow \underline{\underline{O(n^2)}}$

Ⓑ $n^2+n^2+n^2 = 3n^2 \rightarrow 4 \times 3n^2$
$= \underline{\underline{12n^2}}$

Ⓒ $n^2+n^2+n^2 = 3n^2 \rightarrow \underline{\underline{12n^2}}$

ⓓ $A_{N\times N} \times A_{N\times N} = \underline{A^2 = B}$

$\boxed{A^3}$?

$$\underbrace{A \times A}_{\downarrow} \times \underset{\downarrow}{A}$$

$$\underbrace{B \times A}_{C}$$

Time Complexity = ?

$A \times A \rightarrow B$ $\quad$ N^3

$B \times A \rightarrow C$ $\quad$ N^3

$\underline{2N^3}$

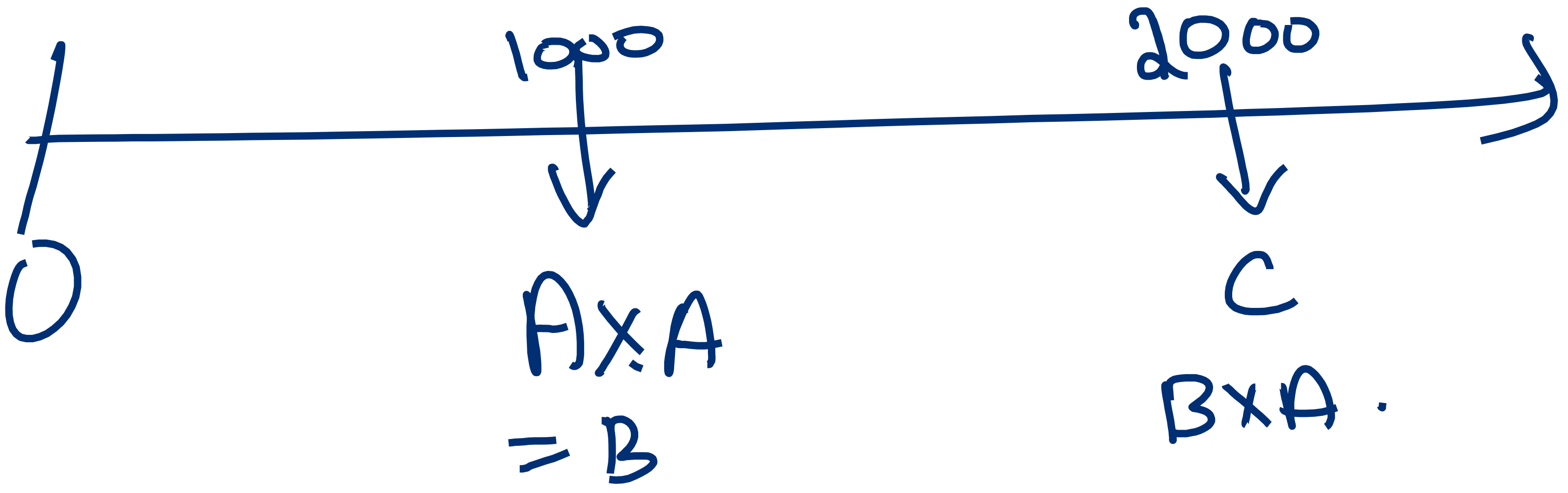
N = 10   t = 1msec

$\boxed{A^2} = O(N3)$ 10x10x10 = 1000 x1 = 1000 msec

↓

B

BxA = O(N3) = 1000msecs



Space complexity

$$\begin{tikzcd} A, \arrow[d] & B, \arrow[d] & C \arrow[d] & \\ n^2 & n^2 & n^2 & = 3n^2 \end{tikzcd}$$

$$\begin{tikzcd} \underbrace{A \times A} \arrow[d] & \times A \arrow[d] \\ B & \times A \end{tikzcd}$$

$$\underbrace{B \times A}_{C}$$

$\boxed{\text{int}} \longrightarrow \quad 4 \times 3n^2 = 12\,n^2$ bytes

When the algo terminates mem. used $= 2\,n^2$

$= 8\,n^2$ bytes