



Arrays - Part I

Course on Data Structure

CS & IT Engineering

Data structure
Arrays -I

Lecture Number- 02

By- Pankaj Sir





Topics

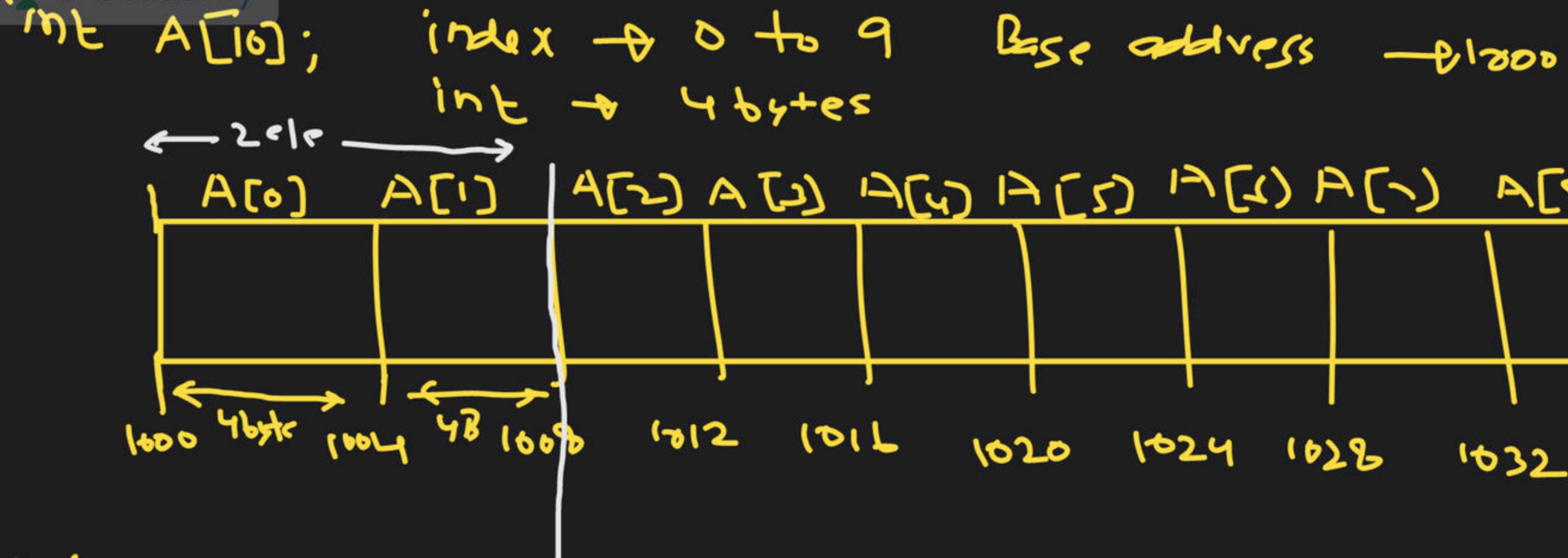
to be covered



- 1 Arrays

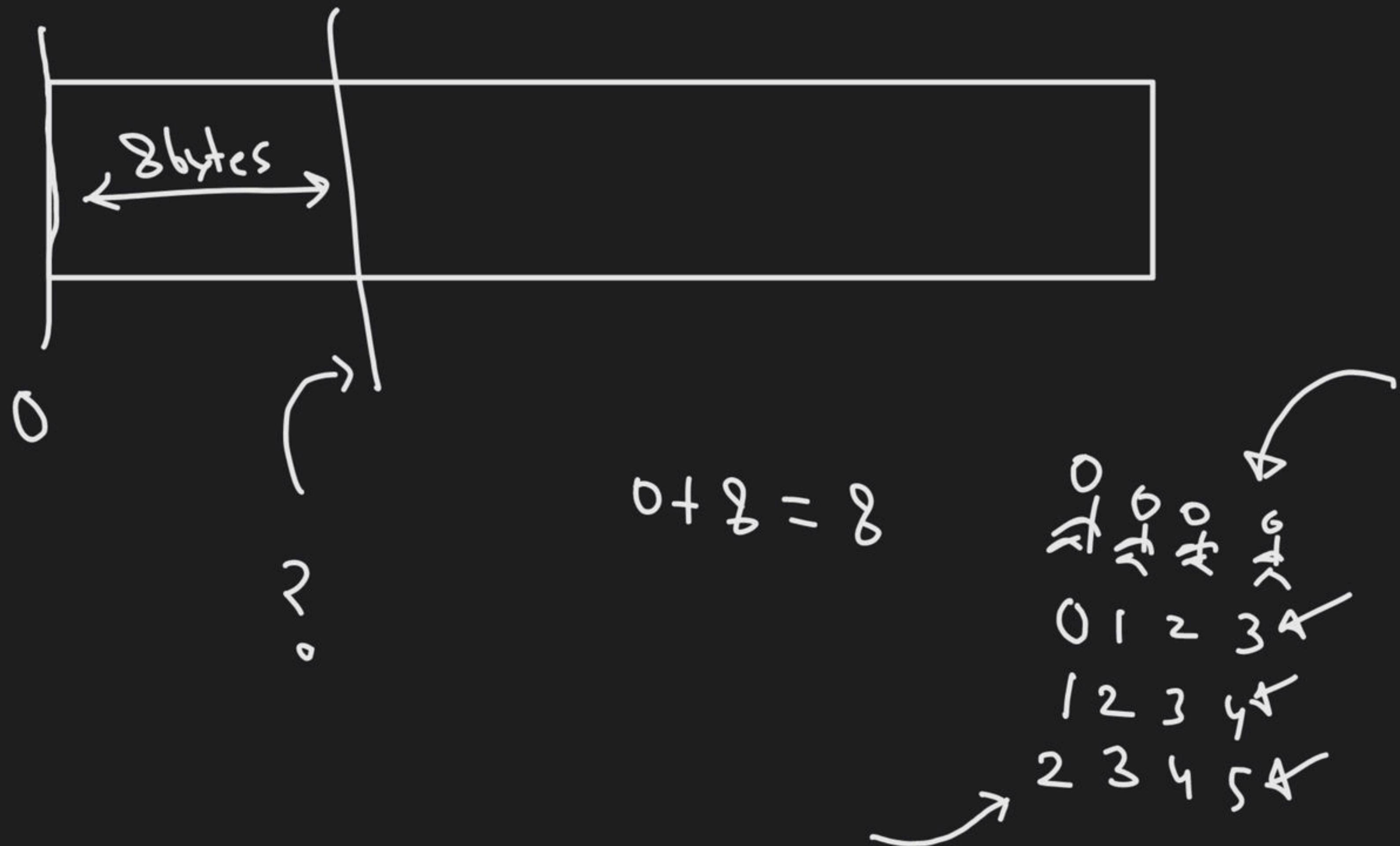
Arrays

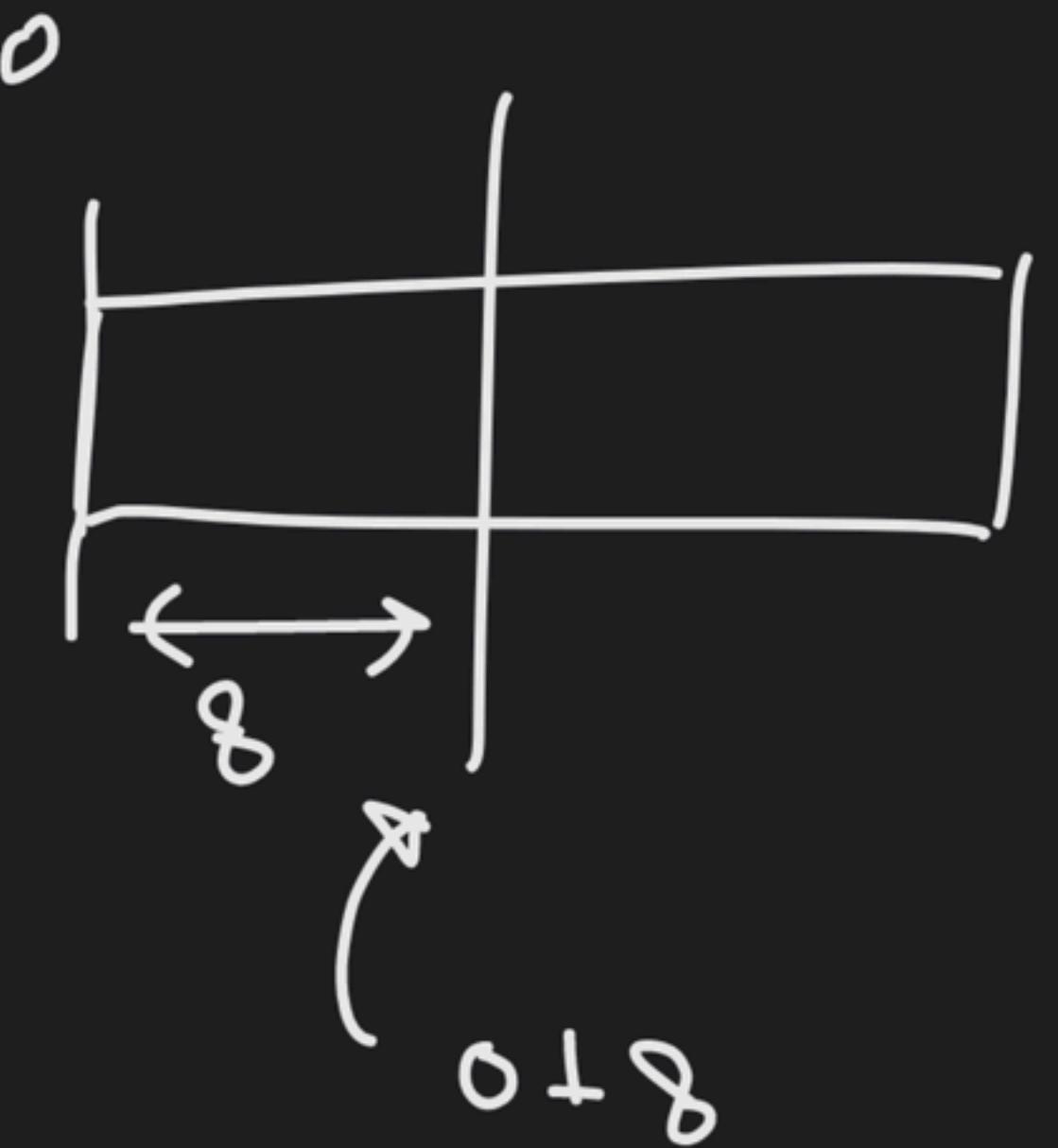
- * Similar types of elements (size of each element is same)
- * One after another in memory.
- * Relative addressing.
- * Base address
- * Index starts from 0 (practically)

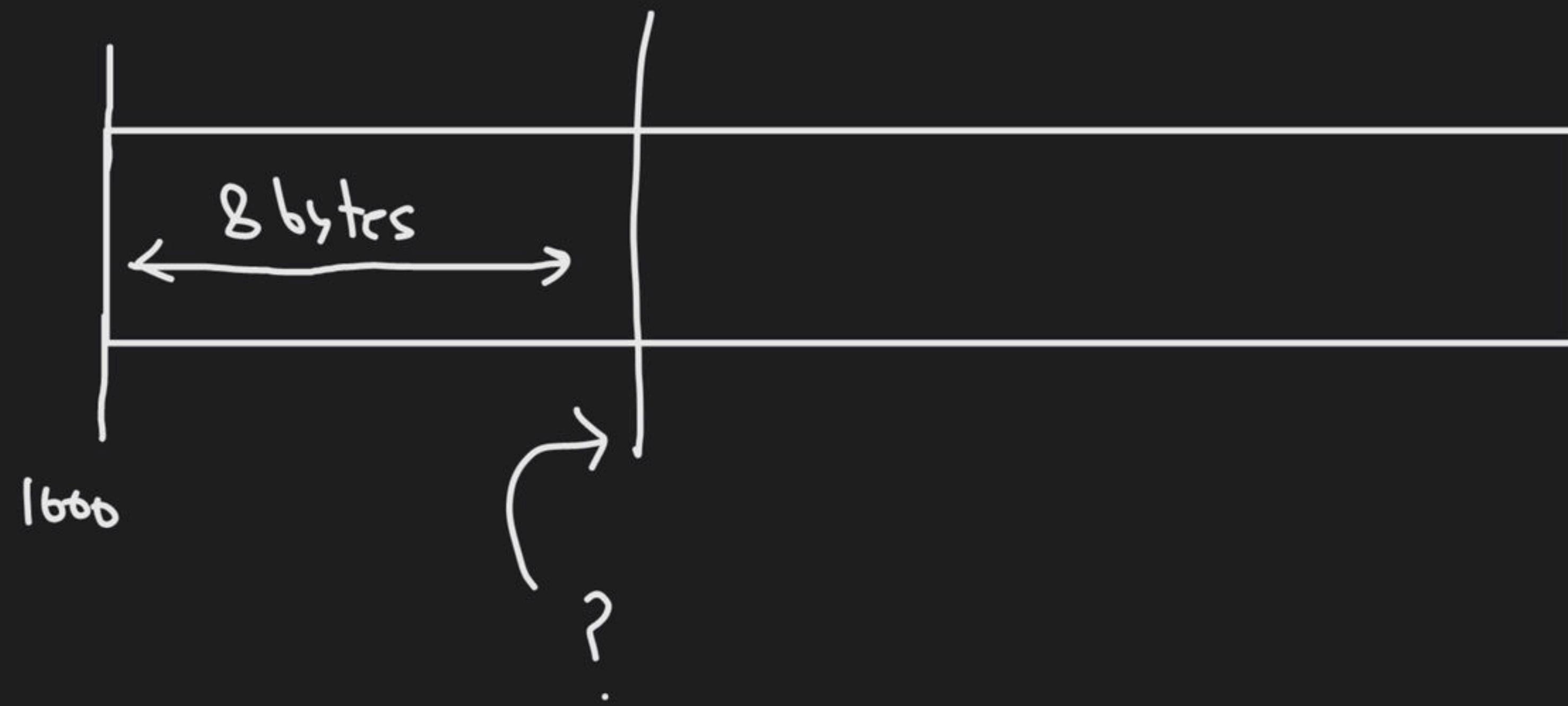


addr(A[2]) ?

- ① How many elements already filled before A[2]?
- ② How much memory already filled by these 2 ele
= 2 * 4 = 8 Bytes







$$1660 + 8 = 1668$$

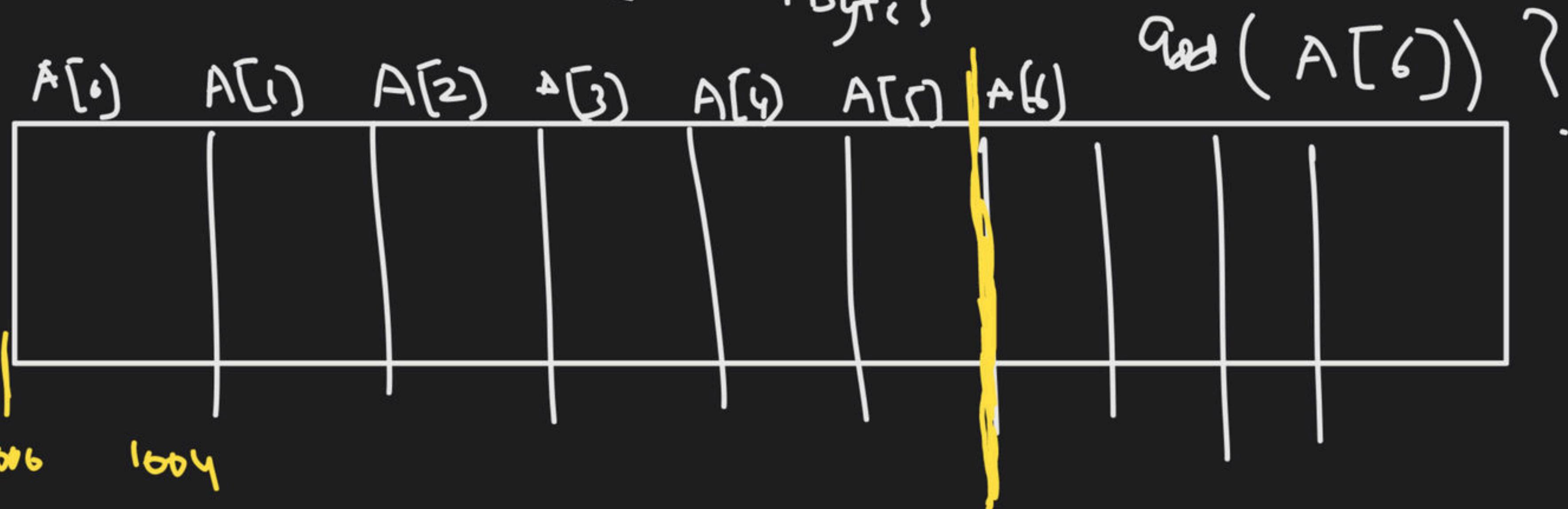
100 to 200 (including both)

last - first + 1

$$200 - 100 + 1 \rightarrow 101$$

Index → 0

'Int A[16]; Base add ⇒ 1000
'3e ⇒ 4 Bytes

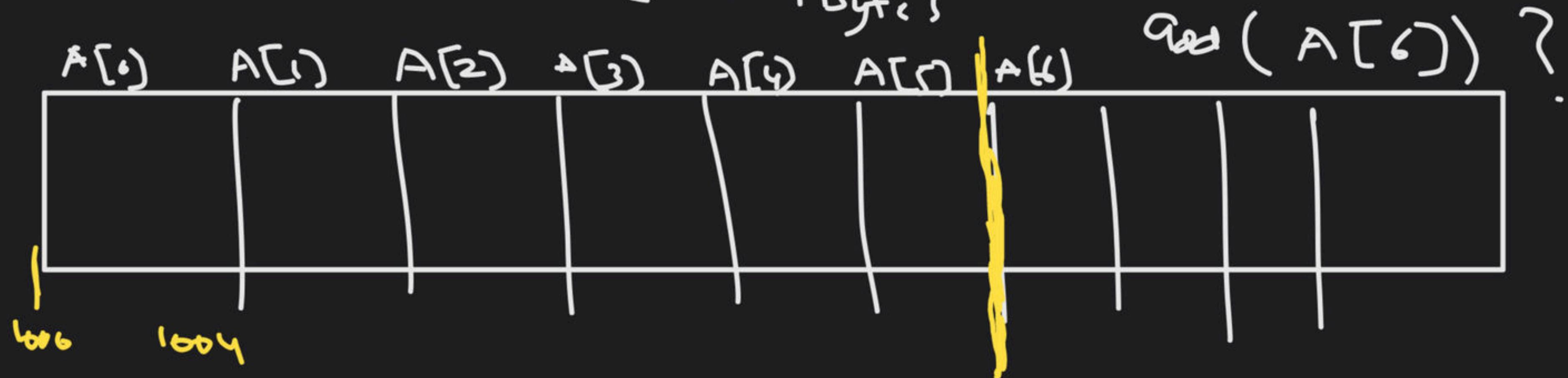


① How many elem. already filled before A[6]?

'Index 0 to 5 = 5 - 0 + 1 = 6 ele'

Index → 0

'Int A[6]; Base add → 1000
 1 byte = 4 Bytes



① How many elem. already filled before $A[6]$?

$$\text{Index } 0 \text{ to } 5 = 5 - 0 + 1 = 6 \text{ ele}$$

② Memory already filled before $A[6] = 6 \times 4 = 24 \text{ Bytes}$



$$BA = 1000$$

$$\text{add}(A[4]) = BA + 24$$

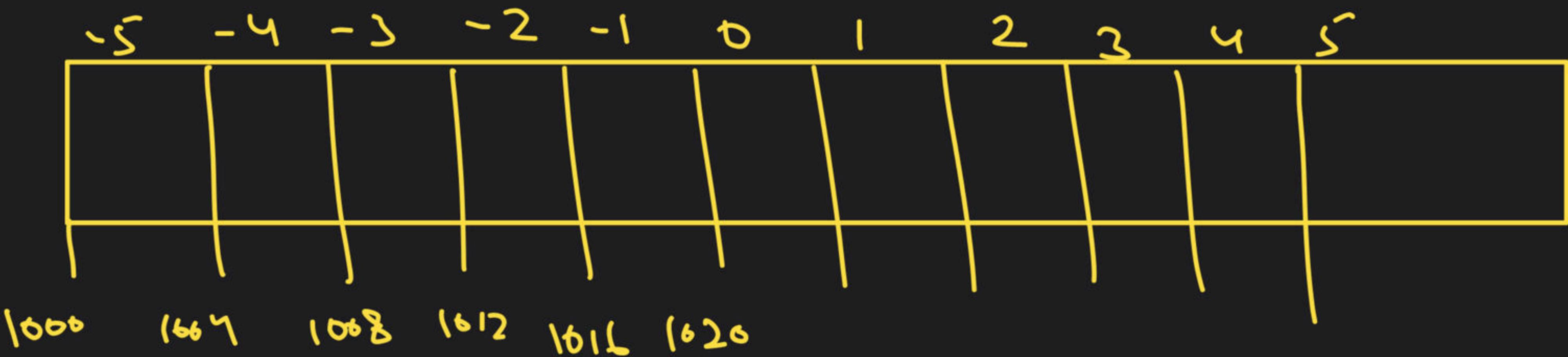
$$= 1000 + 24$$

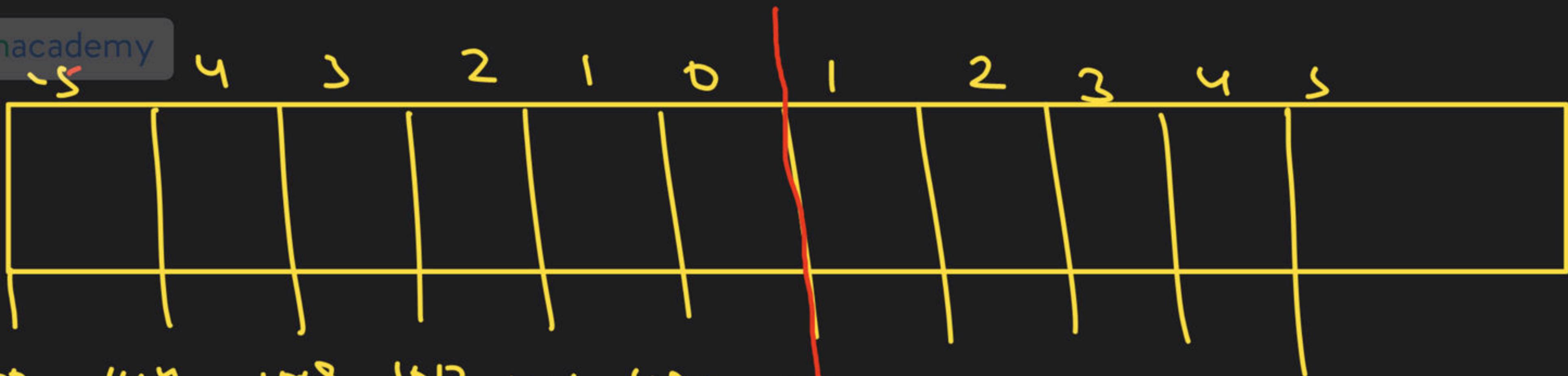
$$= 1024$$

In theory

\Rightarrow index can start from any value

-ve ✓
A[-5..5]



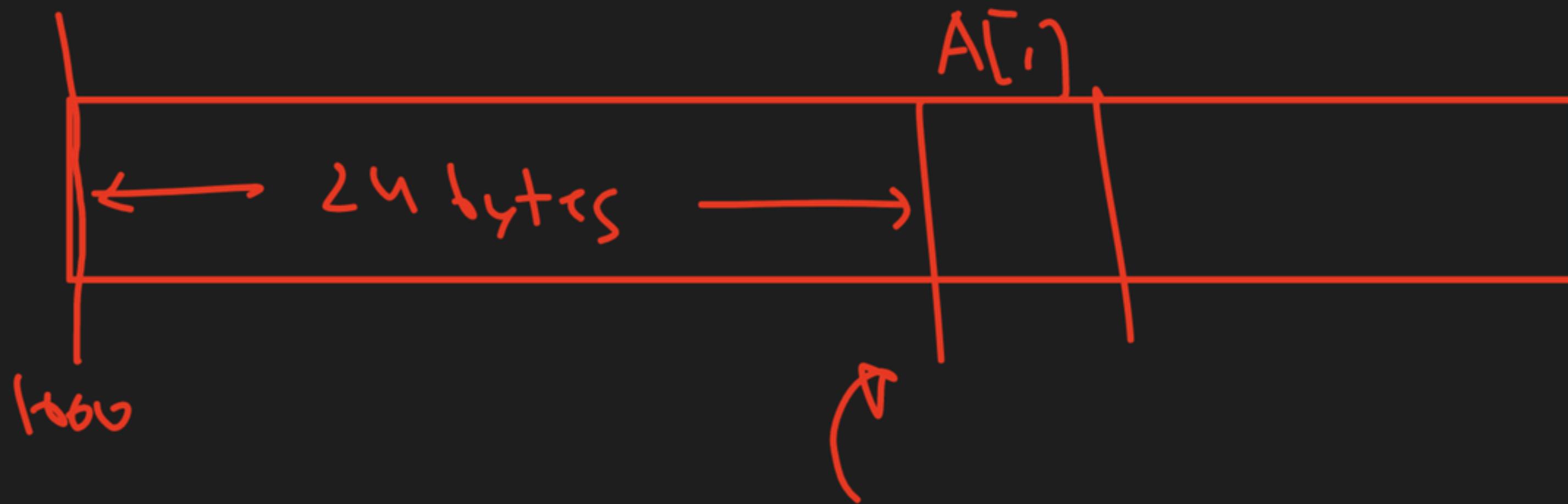


1000 1001 1002 1003 1011 1020

1024

add ($A[i]$)? ① How many ele. already filled before $A[i]$

② Memory already filled = $l \times 4 = 24$ Bytes
 "Index" -5 to 0 $\Rightarrow 0 - (-5) + r = 6$ ele



$$\text{add}(A[i]) = BA + 24 = \underline{l000 + 24}$$

$$= \underline{1024}$$

Q $A[-10..10]$, $\omega = 4$ bytes, $BA \Rightarrow 164$, $\text{add}(A[5])$?
Else already filled before $A[5]$

$$\therefore \text{Index} -10 \text{ to } 5 = 5 - (-10) + 1 = 16 \text{ ele.}$$

Memory already filled before $A[5]$

$$\text{add}(A[5]) = 16 \times 4 = 64 \text{ bytes}$$

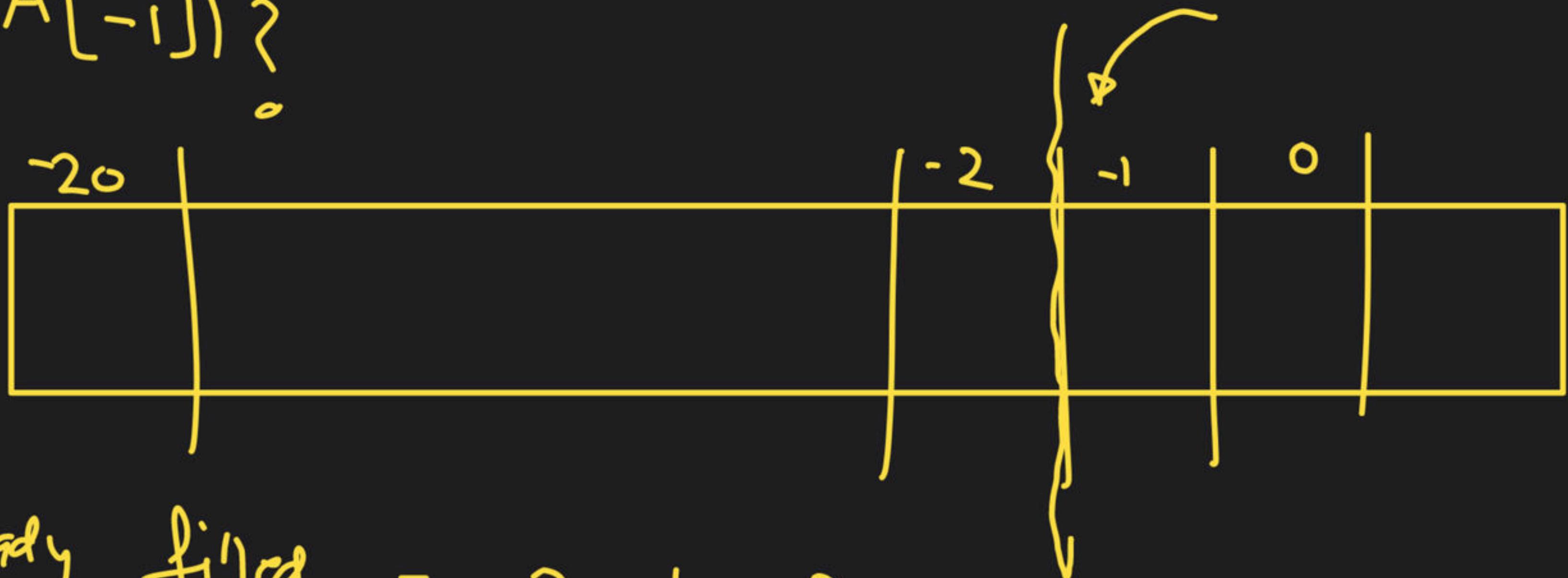
$$= 164 + 14$$

$$= 168$$



$A[-20..20]$, $\omega = 2$ bytes, $BA = 1000$

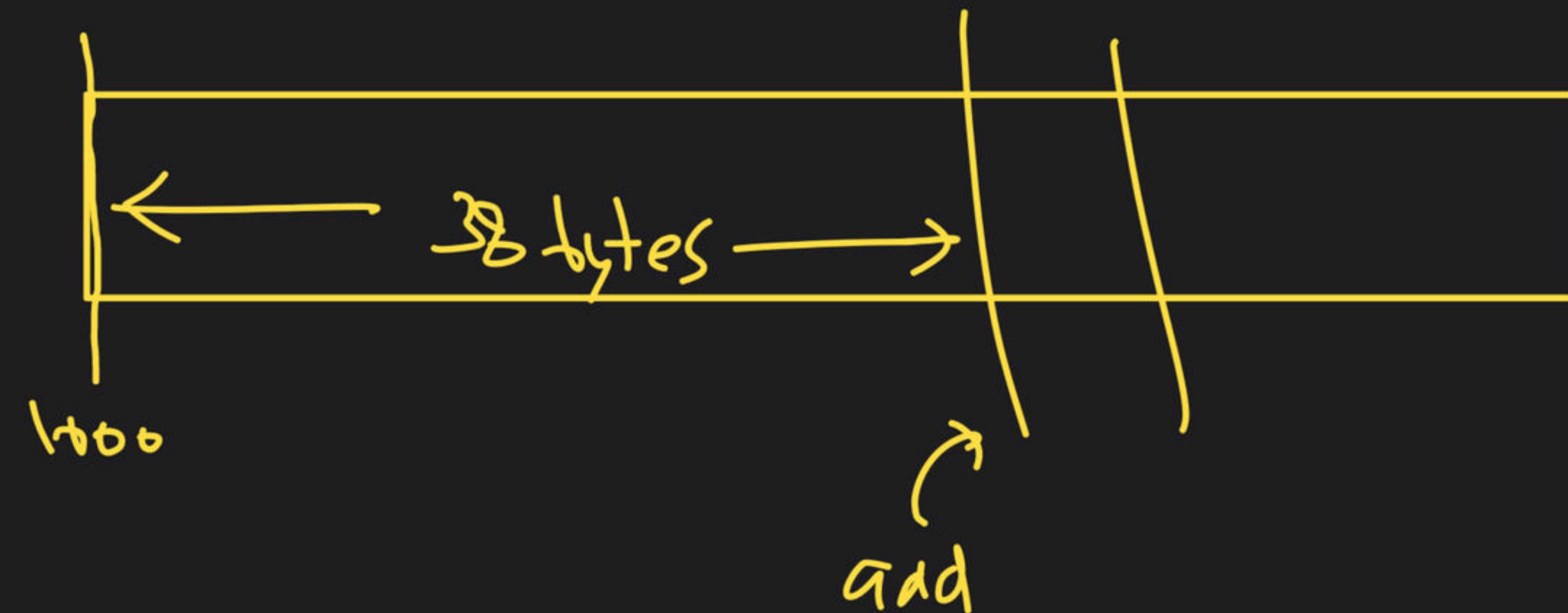
Add($A[-1]$)?



Ele already filled = -20 to -2

$$\therefore -2 - (-20) + 1 = 19 \text{ elem.}$$

Memory already filled = $19 \times 2 = 38 \text{ Bytes}$



$$1000 + 38$$

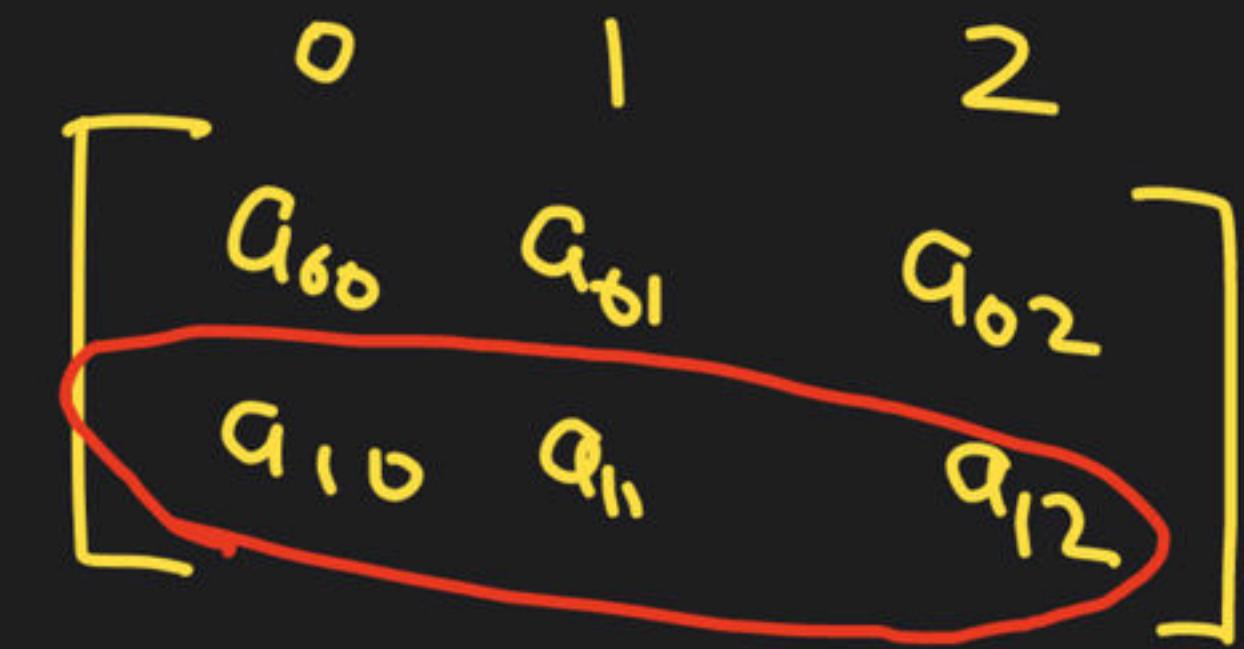
$$\underline{= 1038}$$

```
int a[2][3];
```

0 1 0 1 2

2-D array

row with index 0 \Rightarrow 0
 row with index 1 \Rightarrow 1

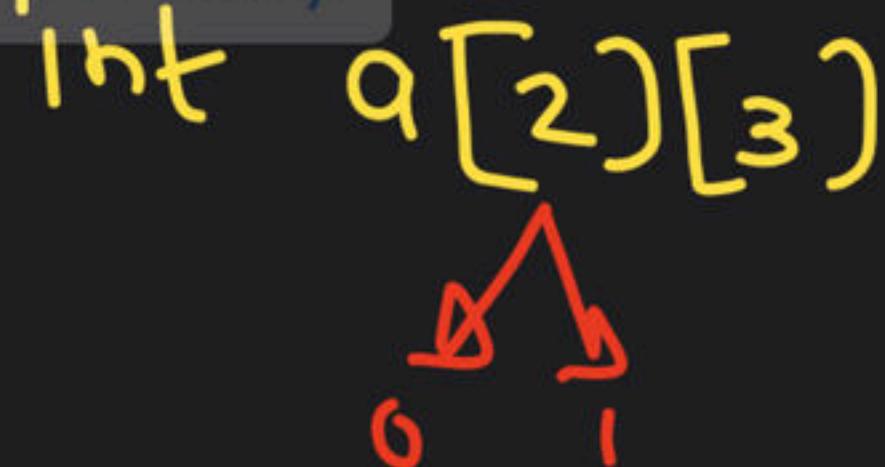


No. of ele in row with index 0 \Rightarrow 3

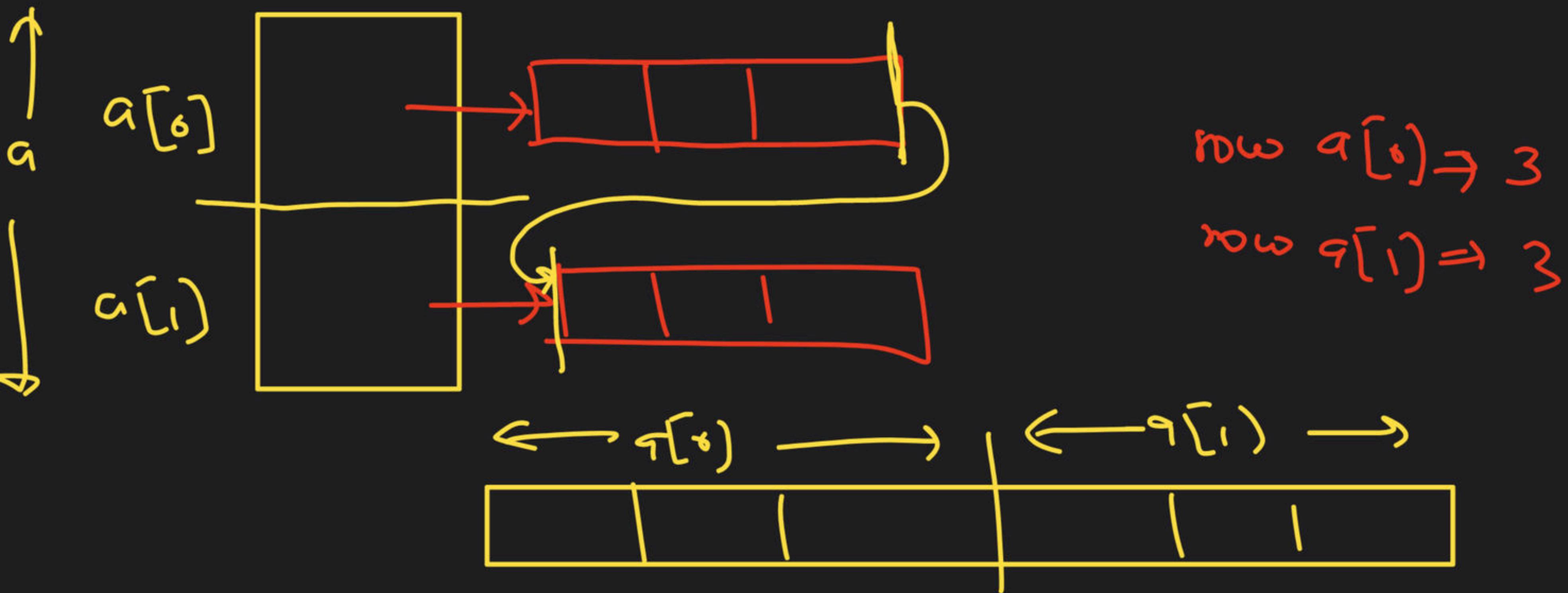
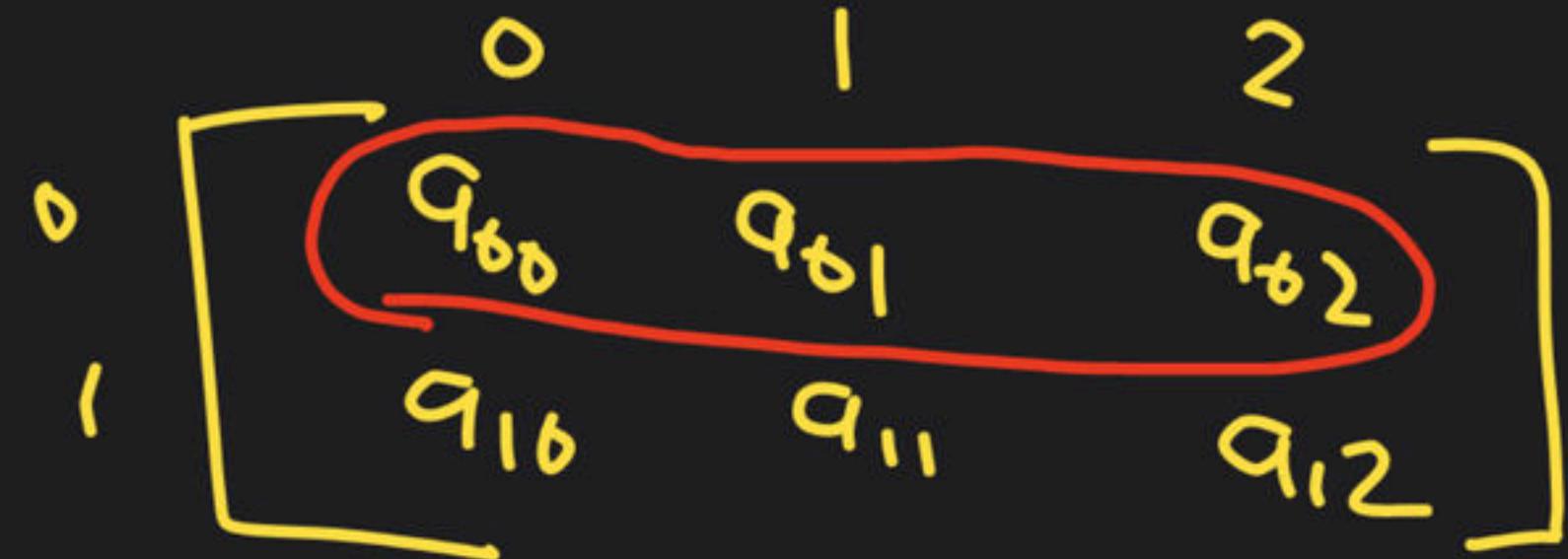
No. of ele " " " " " 1 \Rightarrow 3

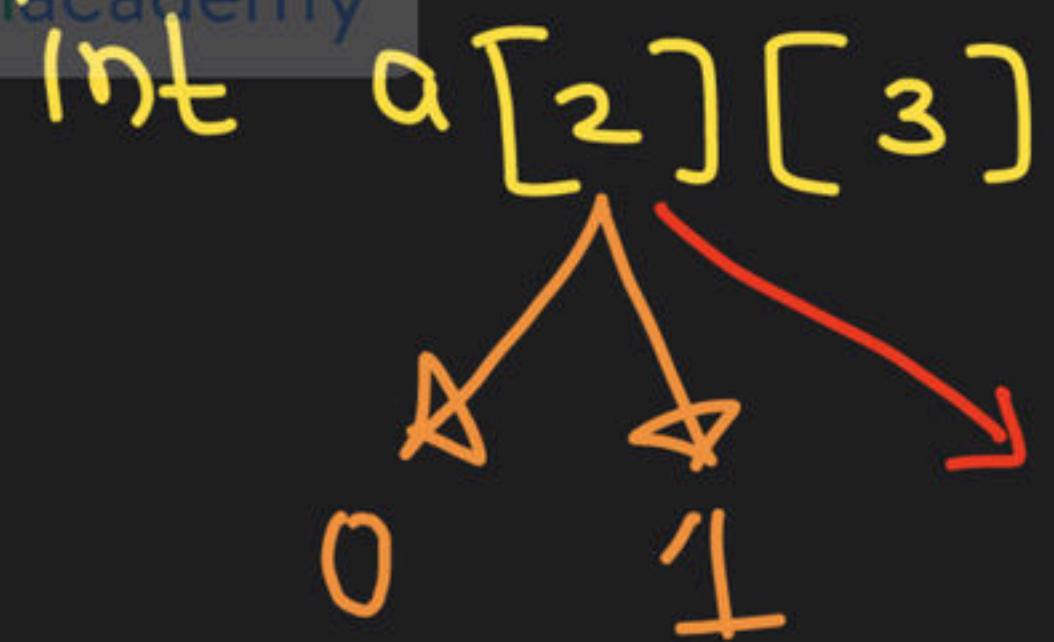
Each row \Rightarrow 3 elements \Rightarrow No. of columns

Int $a[2][3]$



A diagram showing a 2x3 matrix. The top row has indices 0, 1, 2 above it. The bottom row has indices 0, 1, 2 below it. Red arrows point from the text 'Int a[2][3]' to the element $a_{0,1}$ at position (0, 1).





$a[0]$
 Each index/Number
 in this dimension

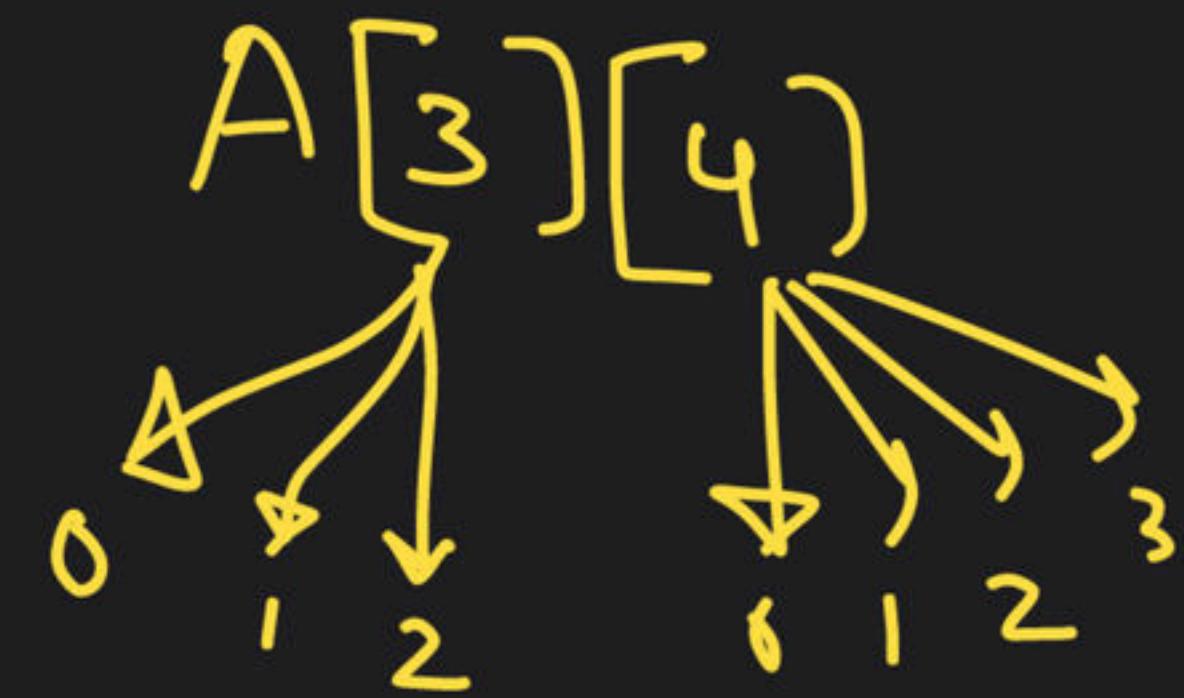


ele in row with index 0 $\Rightarrow 3$
 # ele in row with index 1 $\Rightarrow 3$

$$a[0] \Rightarrow 3$$

$$a[1] \Rightarrow 3$$

-concept

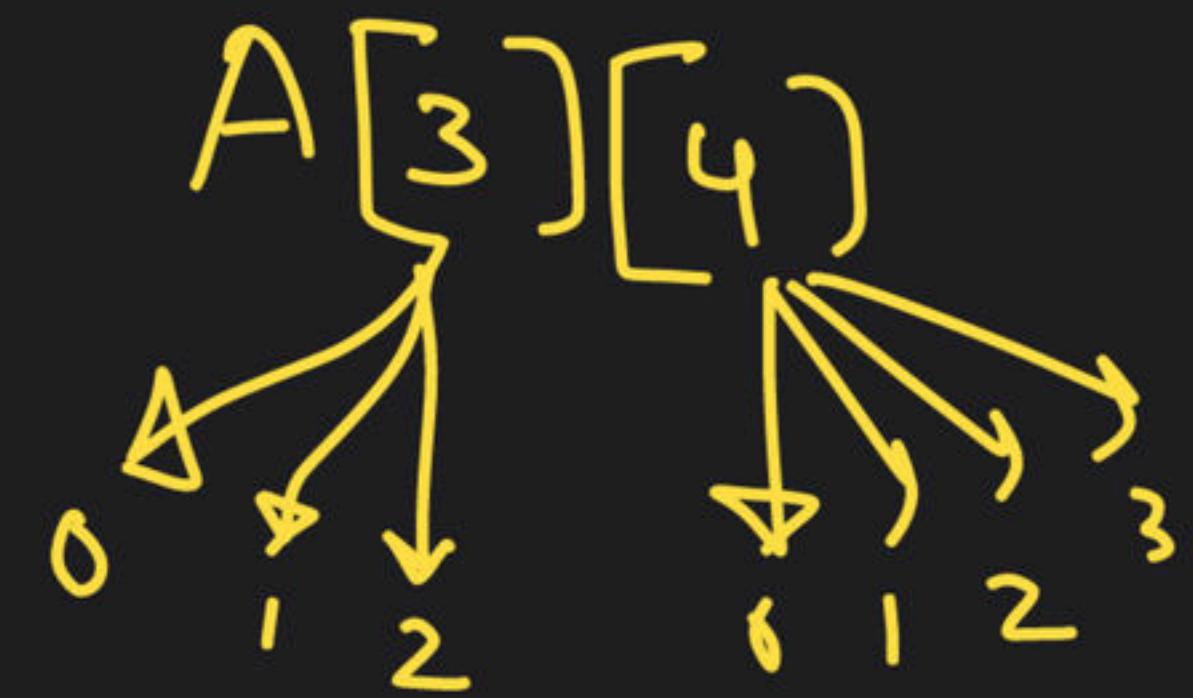


① Random access

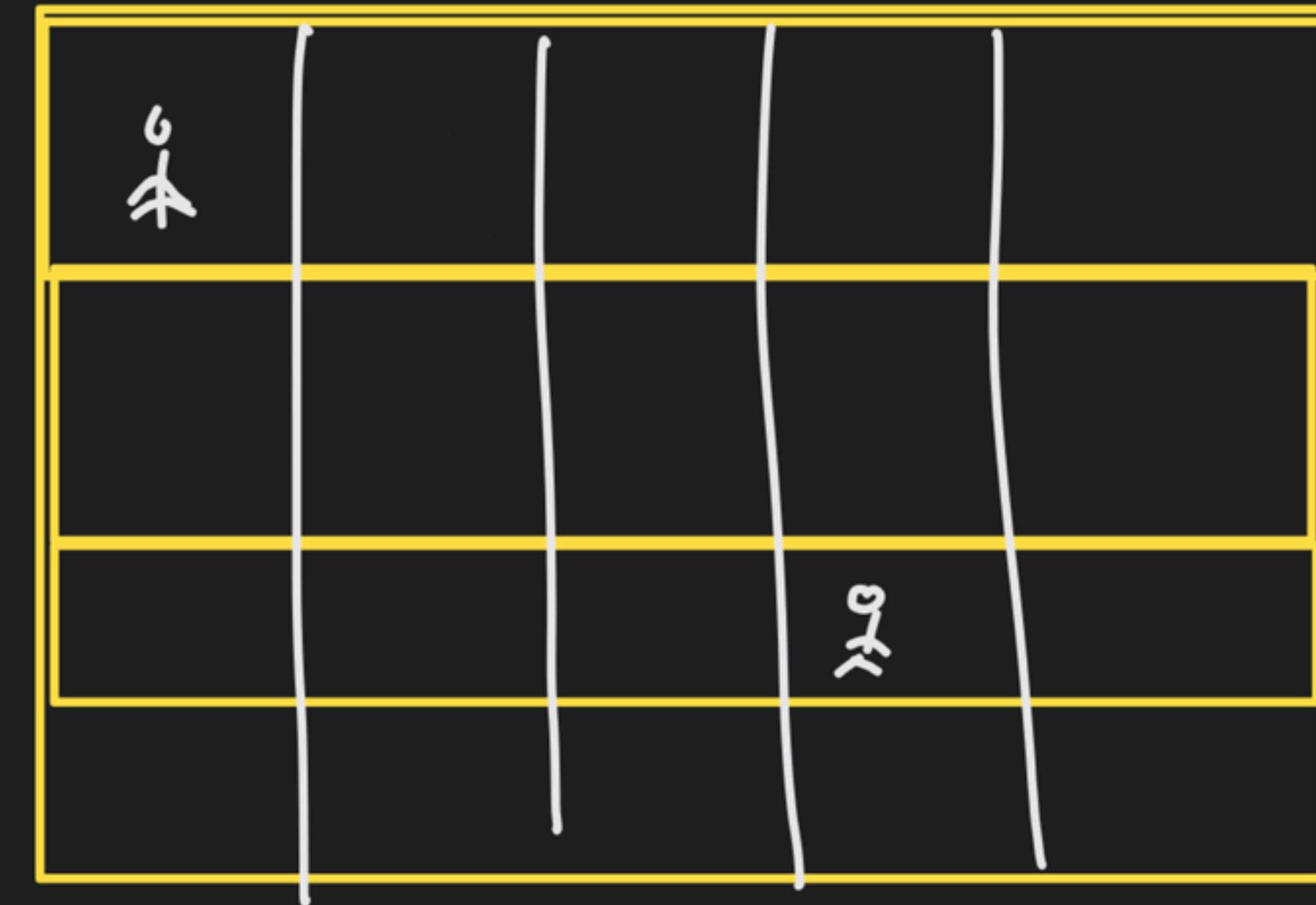
0	q_{00}	q_{01}	q_{02}	q_{03}
1	q_{10}	q_{11}	q_{12}	q_{13}
2	q_{20}	q_{21}	q_{22}	q_{23}

$\text{pf}(".\text{dot}", A[1][2]) \rightarrow \text{Ans}$

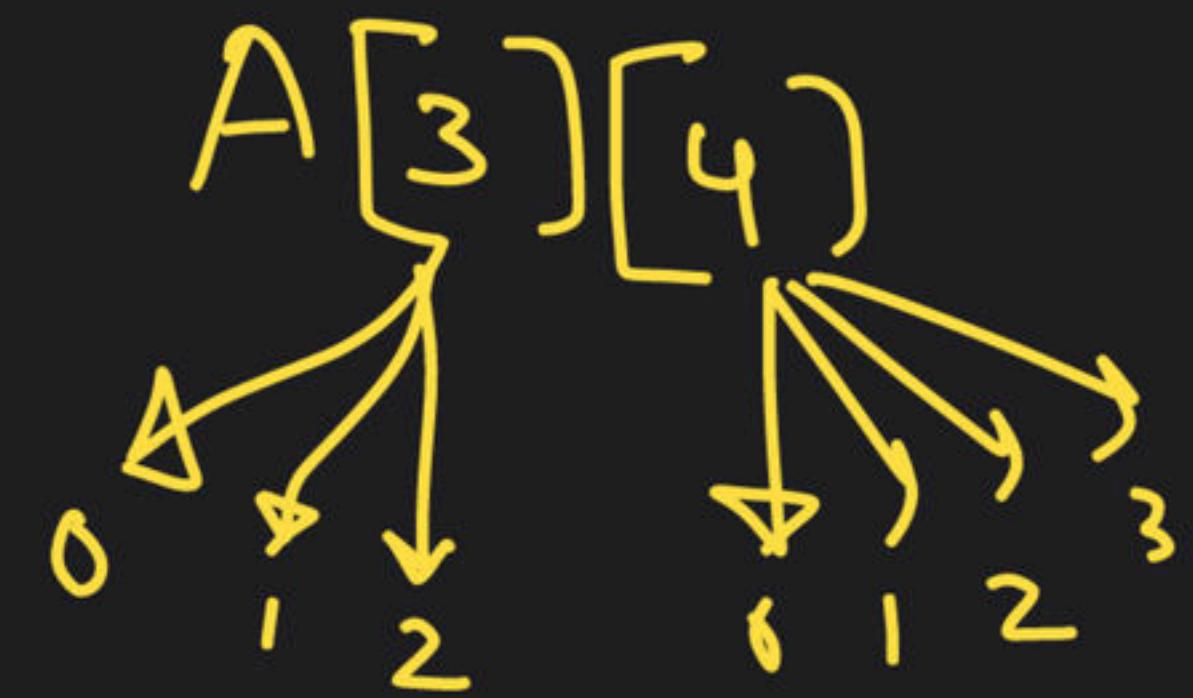
(-concept)



0	q_{00}	q_{01}	q_{02}	q_{03}
1	q_{10}	q_{11}	q_{12}	q_{13}
2	q_{20}	q_{21}	q_{22}	q_{23}



-concept

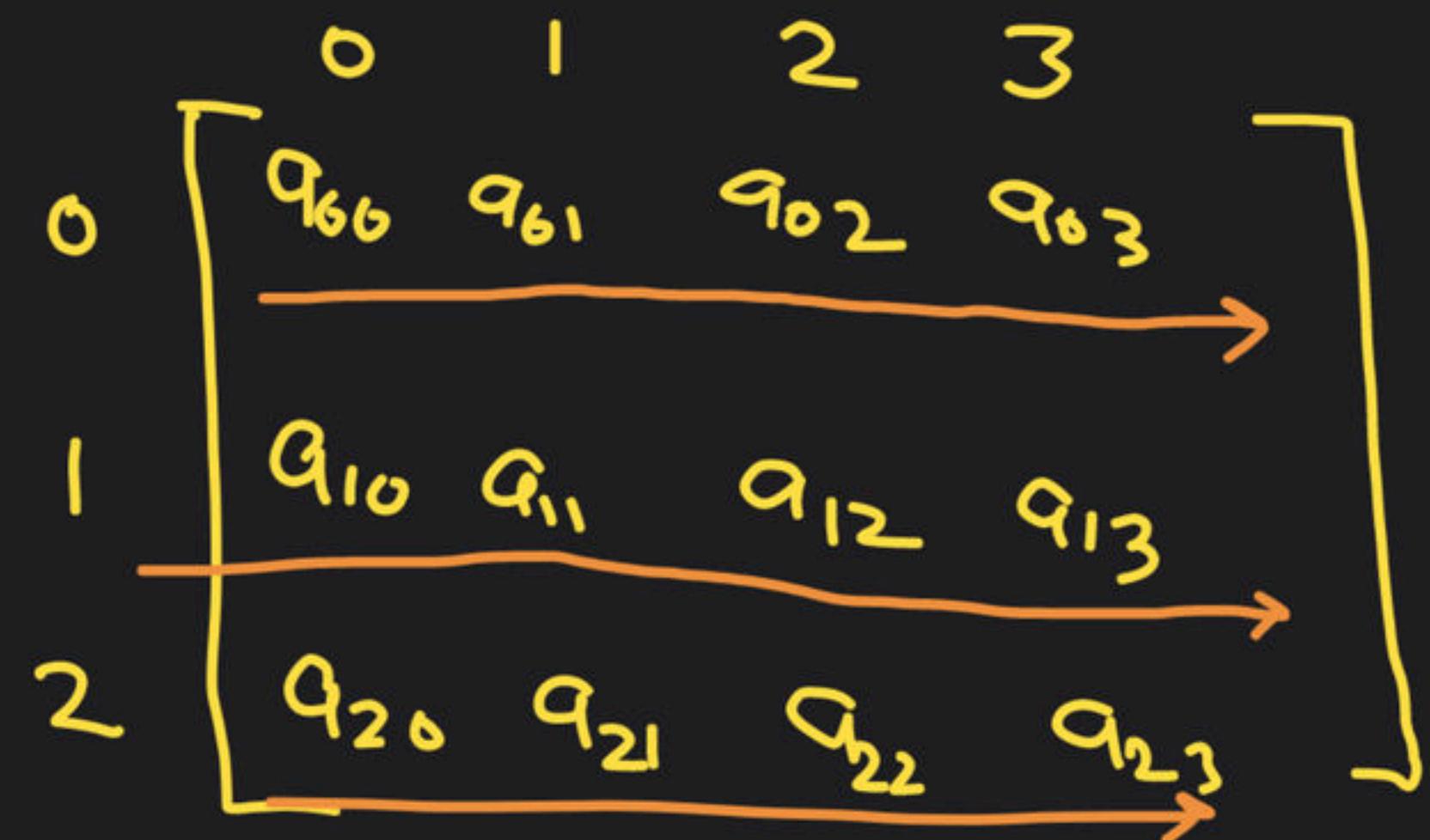
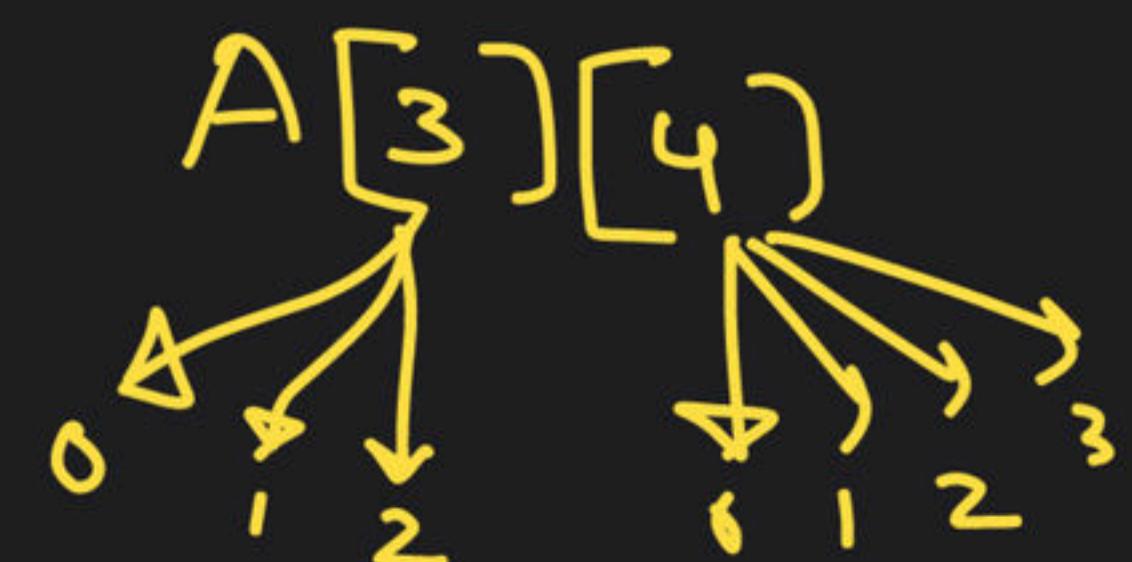


store

- ① Row-wise
- ② column-wise

0	1	2	3	
0	q_{00}	q_{01}	q_{02}	q_{03}
1	q_{10}	q_{11}	q_{12}	q_{13}
2	q_{20}	q_{21}	q_{22}	q_{23}

Row-wise



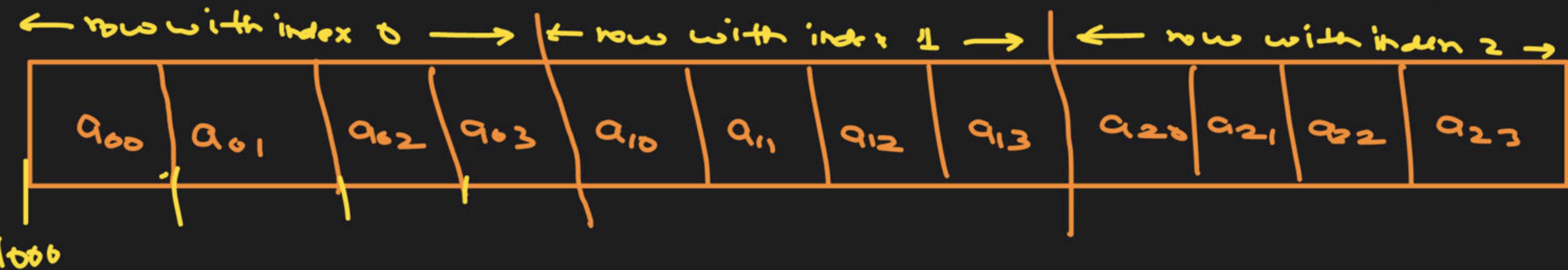
← row with index 0 → ← row with index 1 → ← row with index 2 →

q_{60}	q_{61}	q_{62}	q_{63}	q_{10}	q_{11}	q_{12}	q_{13}	q_{20}	q_{21}	q_{22}	q_{23}
----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------

Row-wise



	0	1	2	3
0	a_{00}	a_{01}	a_{02}	a_{03}
1	a_{10}	a_{11}	a_{12}	a_{13}
2	a_{20}	a_{21}	a_{22}	a_{23}



Add (a_{23})

Every row \Rightarrow 4 ele

1) How many rows
already filled

$$\text{rows with index } 2 = 0 \text{ to } 1 = 1 - 0 + 1 = 2 \text{ rows}$$

$a_{2,3}$
row's index = 2 col's index

	0	1	2	3
0	$a_{0,0}$	$a_{0,1}$	$a_{0,2}$	$a_{0,3}$
1	$a_{1,0}$	$a_{1,1}$	$a_{1,2}$	$a_{1,3}$
2	$a_{2,0}$	$a_{2,1}$	$a_{2,2}$	$a_{2,3}$

rows already filled

Every row \Rightarrow 4 ele

	$\sqrt{0}$	$\sqrt{1}$	$\sqrt{2}$	$\sqrt{3}$
0	a_{00}	a_{01}	a_{02}	a_{03}
1	a_{10}	a_{11}	a_{12}	a_{13}
2	a_{20}	a_{21}	a_{22}	a_{23}

\uparrow

rows already filled

row-index filled

1) How many rows already filled

before row with index 2 = 0 to 1 = 1-0+1 = 2 rows

2) Elements already filled in row with index 2

before a_{23} = Col index 0 to 2

= 2-0+1 = 3 elem.

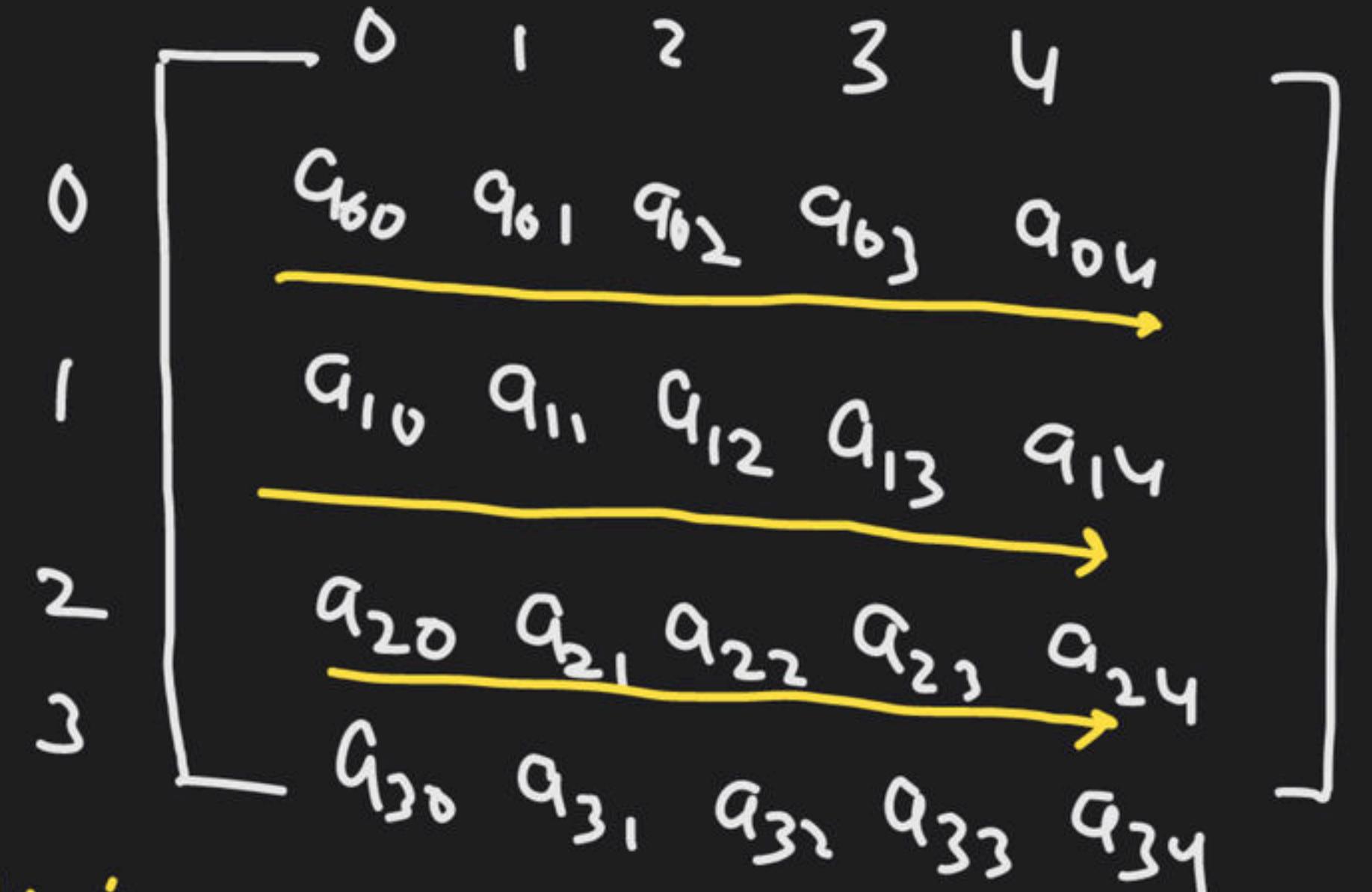
2 rows + 3 cols
X
y

Total elem $\Rightarrow 2 \times 4 + 3 = 11$ ele.

Row-wise

int A[4][5];

Each index in
this dim rep
= 5 elem.



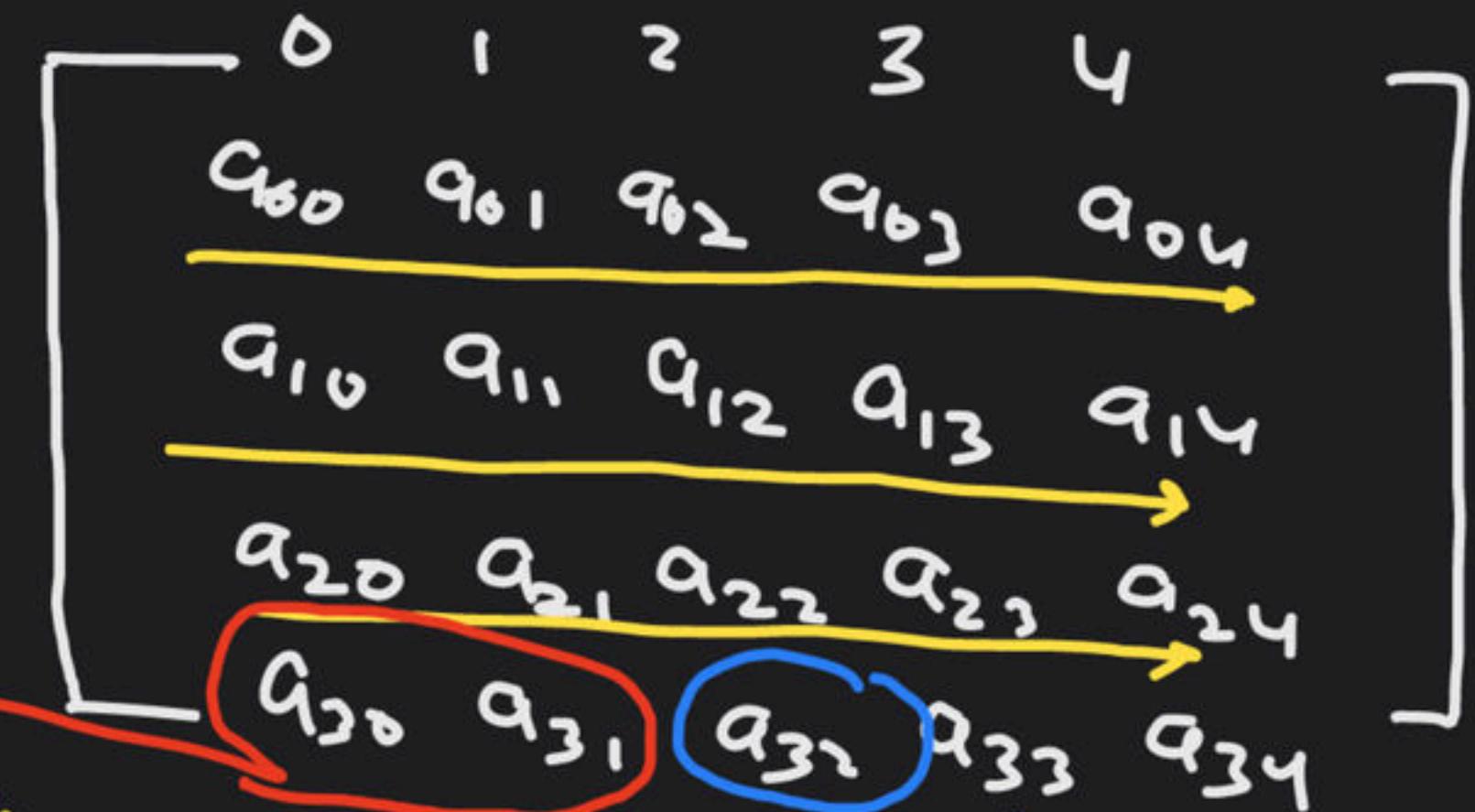
← row with index 0 →	← row with index 1 →	← row with index 2 →	← row with index 3 →
a_{00}	a_{01}	a_{02}	a_{03}

$\text{add}(a_{32})$

Each no./index in
this dim ref
= 5 elem.

'int A[4][5];'

Rows
already
filled
ele. filled
already 4



← row with index 0 →	← row with index 1 →	← index = 2 →	← index = 3 →
a_{00} a_{01} a_{02} a_{03} a_{04}	a_{10} a_{11} a_{12} a_{13} a_{14}	a_{20} a_{21} a_{22} a_{23} a_{24}	a_{30} a_{31} a_{32} a_{33} a_{34}

$\text{add}(q_{32})$

'int A[4][5];'

Each no./index in
this dim ref
= 5 elem.

0	1	2	3	4	
0	q_{00}	q_{01}	q_{02}	q_{03}	q_{04}
1	q_{10}	q_{11}	q_{12}	q_{13}	q_{14}
2	q_{20}	q_{21}	q_{22}	q_{23}	q_{24}
3	q_{30}	q_{31}	q_{32}	q_{33}	q_{34}

← row with index 0 →	← row with index 1 →	← index = 2 →	← index = 3 →
q_{00} q_{01} q_{02} q_{03} q_{04}	q_{10} q_{11} q_{12} q_{13} q_{14}	q_{20} q_{21} q_{22} q_{23} q_{24}	q_{30} q_{31} q_{32} q_{33} q_{34}

1000 1001 1002 1012 1111 1210 1224 1636 1040 1644 1148 1032 1068

$1 \rightarrow 10$

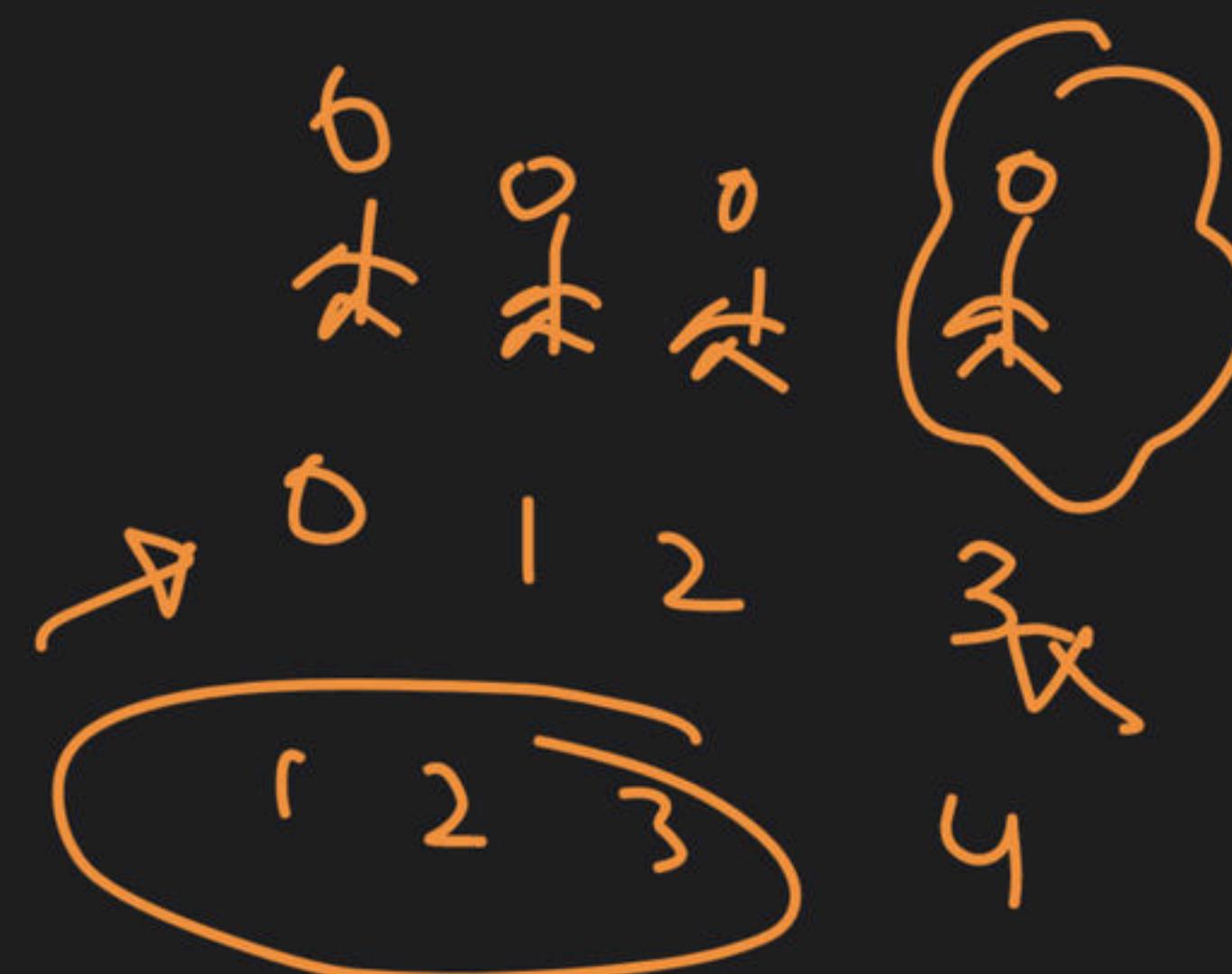


$|b - d|$

$\Rightarrow 10$

$100 + 2\sigma$

last - first + 1





0 to 2

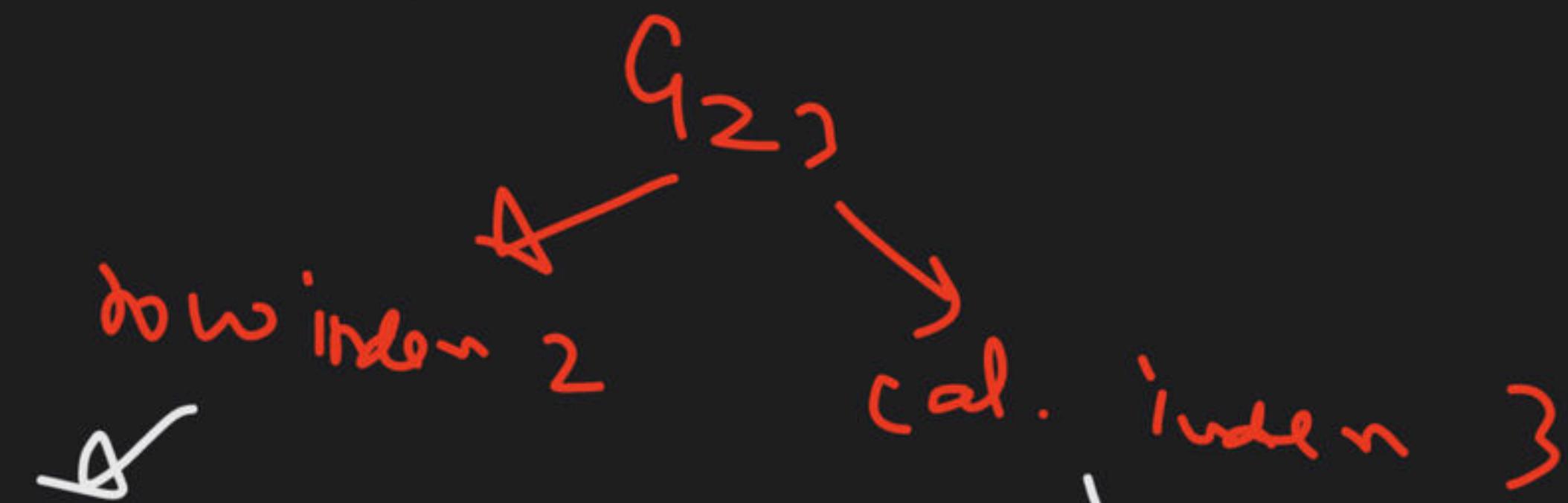
$\Rightarrow \checkmark, \checkmark, \checkmark$

$\Rightarrow 2 - 0 + 1 = 3$ rows

Ex int $A[3][5]$; R640-wise
 $w = 2 \text{ bytes}, DA = 100$

$\text{add}(A[2][3])$

① # ele in each row = 5



rows already filled

$\Rightarrow 0 \text{ to } 1$

$$= 1-0+1 = 2 \text{ rows}$$

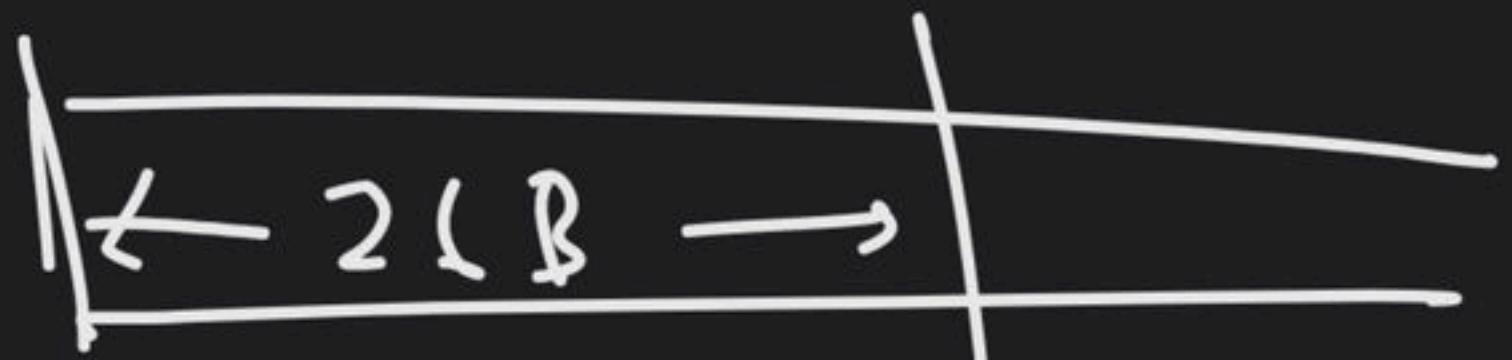
within row index 2, ele.
 already filled = cal index 0 to 2
 $= 2-0+1 = 3 \text{ ele.}$

g_{23} is filled after 2 rows & 3 columns



$$2 \times 5 + 3$$

Total ele. already filled before g_{23} = 13
remainder = 13 $\times 2 = 26$



$$100 + 26 = 126$$

✓ Args, pointers, structures

4x5

add (a_{32})

row with index 3

col 'index'

0	1	2	3	4
a_{00}	a_{01}	a_{02}	a_{03}	a_{04}
a_{10}	a_{11}	a_{12}	a_{13}	a_{14}
a_{20}	a_{21}	a_{22}	a_{23}	a_{24}
a_{30}	a_{31}	a_{32}	a_{33}	a_{34}

- ① Rows already filled before row with index 3
 $= \text{index } 0 \text{ to } 2 \Rightarrow 2 - 0 + 1 = 3 \text{ rows}$
- ② Within row with index 3, ele. already filled before
 $a_{32} = \text{col index } 0 \text{ to } 1 = 1 - 0 + 1 = 2 \text{ ele.}$

After 3 rows & 2 ele q_{j2} is filled

\downarrow \downarrow

$3 \times 5 + 2$

Total ele already filled before $q_{j2} = 17$ ele.

" memory " " " = $17 \times 4 = 68$ byte

$BA = 1000$

1000

6

← 6 bytes →

1000 - 68
= 1038



THANK YOU!

Here's to a cracking journey ahead!