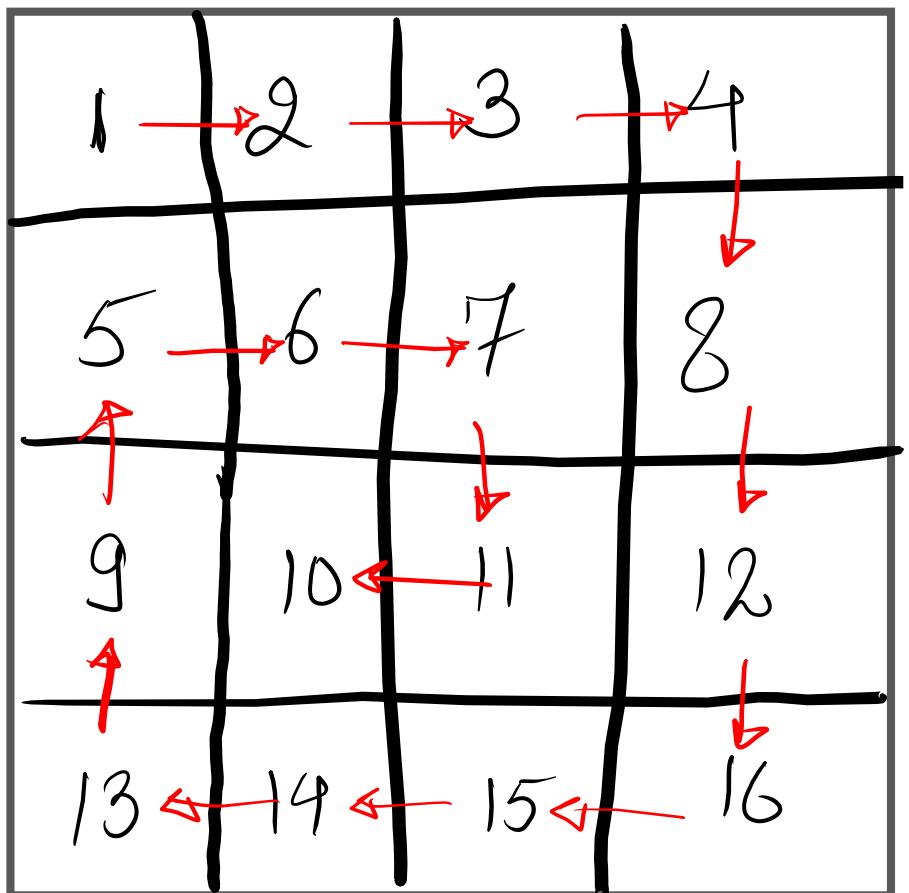


~~4/6~~

* SPIRAL MATRIX



O/P :

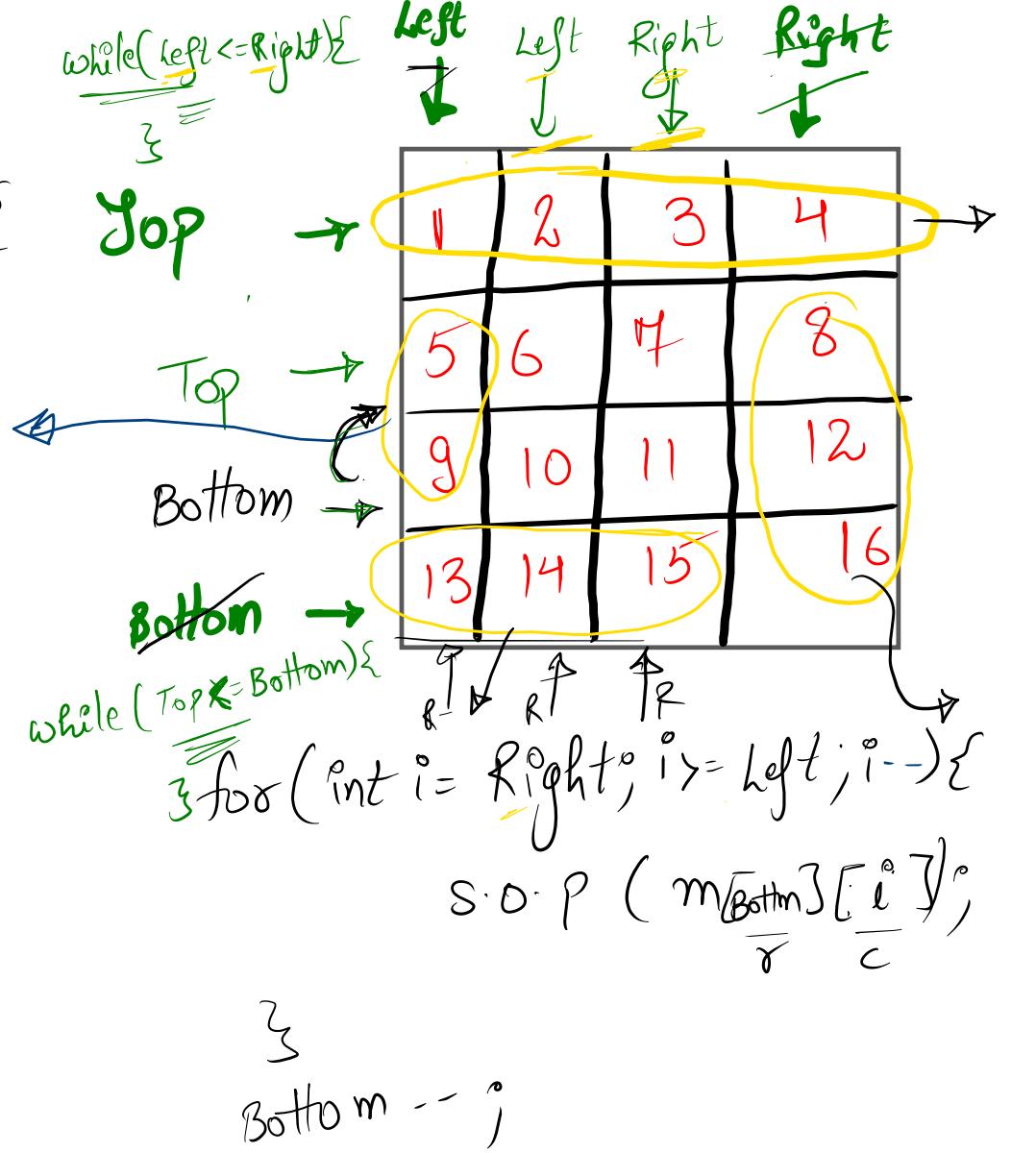
1 2 3 4 8 12 16 15 14
13 9 5 6 7 11 10

```

for( int i= Bottom; i >= Top ; i-- ) {
    s.o.p( m[ i ][ Left ] );
}

```

Left ++;



Top = 0 ; left = 0 ; Right = n - 1 ; Bottom = n - 1 ;

```

for( i = left ; i <= Right ; i++ ) {
    s.o.p( m[ Top ][ i ] );
}
Top++;
for( int i = Top ; i <= Bottom ; i++ ) {
    s.o.p( m[ i ][ Right ] );
}
Right--;

```

row col.

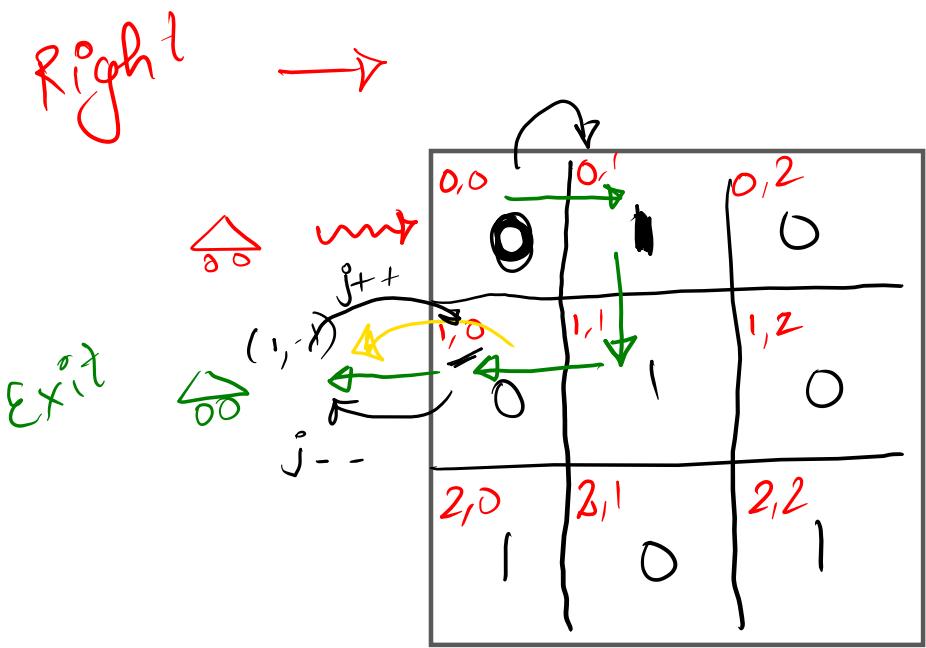
Condition

while (left <= right & Top <= Bottom) {
 if (left <= right) (for loop)
 ①
 ②
 if (Top <= Bottom) (for loop)
 ③
 ④
 if (Right >= Left) (for loop)
 if (Bottom >= Top) (for loop)

}

Find The Way

exit
 $\text{dir} = \text{"left"}$
 $[1, -1]$
 $j++$
 $i--$
 exit
 $[1, 0]$
return cell

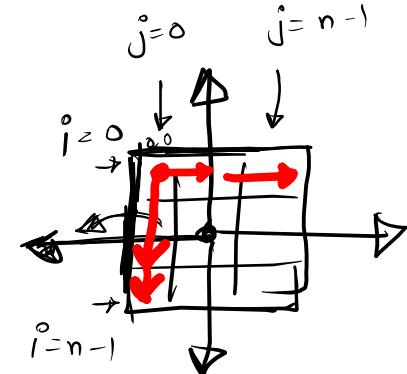


0/0 \Rightarrow $[1, 0]$

outside
of the matrix

false

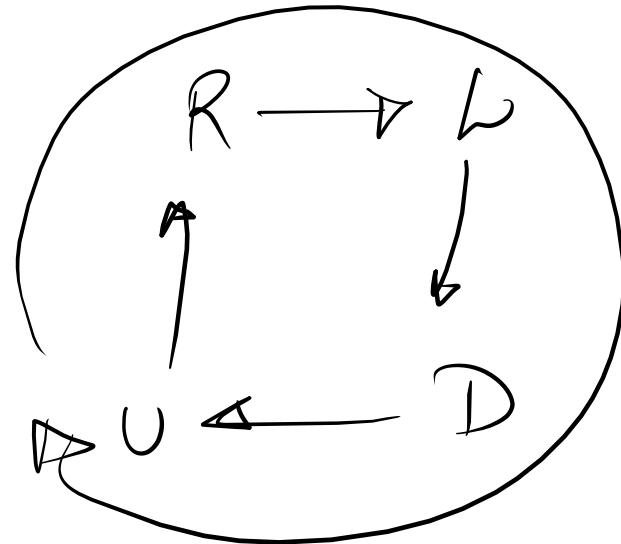
while ($i >= 0$ && $i < n$ && $j >= 0$ && $j < m$);



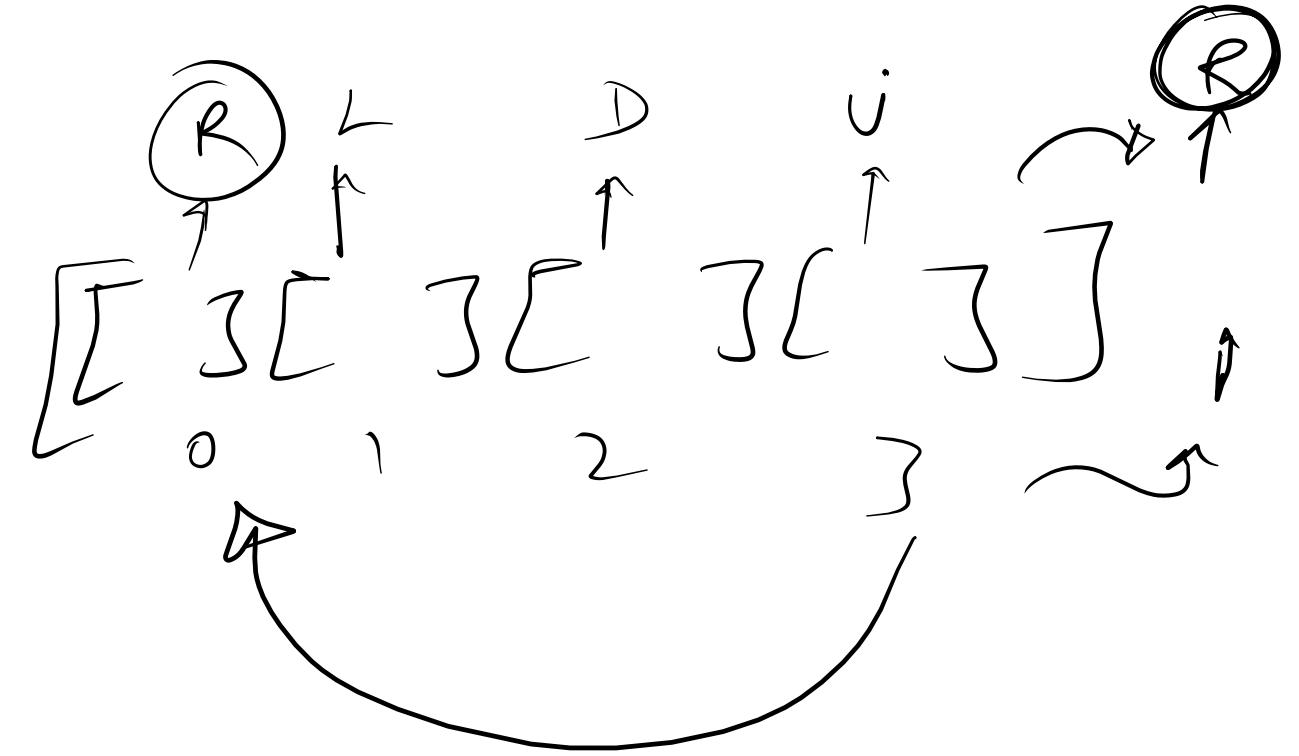
// Inside the matrix

=
{

#



using
Remainder



$$\underline{3+1} \Rightarrow 4 \% 4 \Rightarrow 0$$

switch Cases

x @ \$

switch(Expression) {

+ → case - 1 : // statement

+ → case - 2 : // process

+ → case - 3 : // process

+ → default : // process

→ "Please Enter Number"

}

→ WAP

to Read the gender (M/F/NPS)

~~use~~ ~~o/p~~

'f' 'F'



Point = '

you are a Female

'm' 'M'



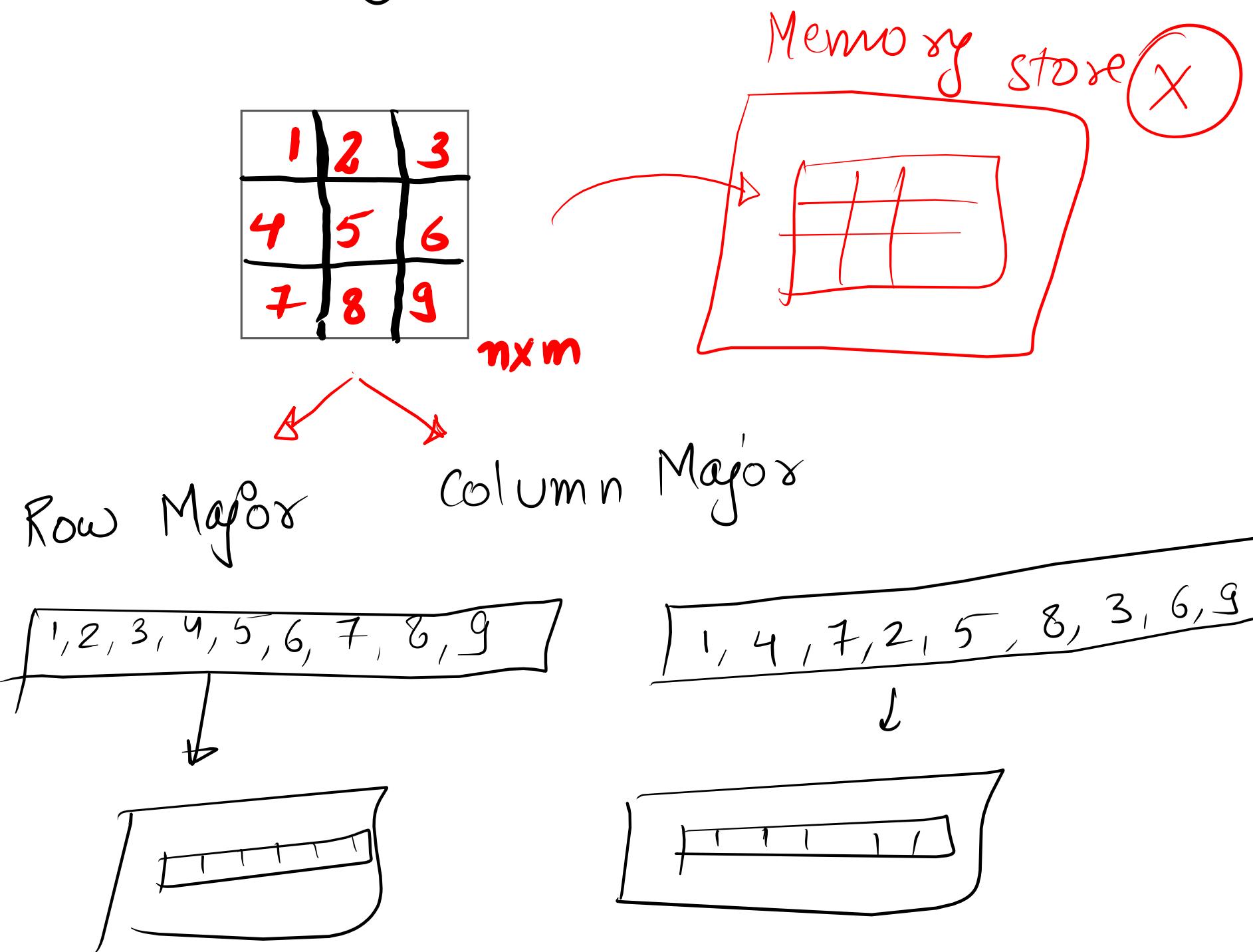
you are a Men.

T/L

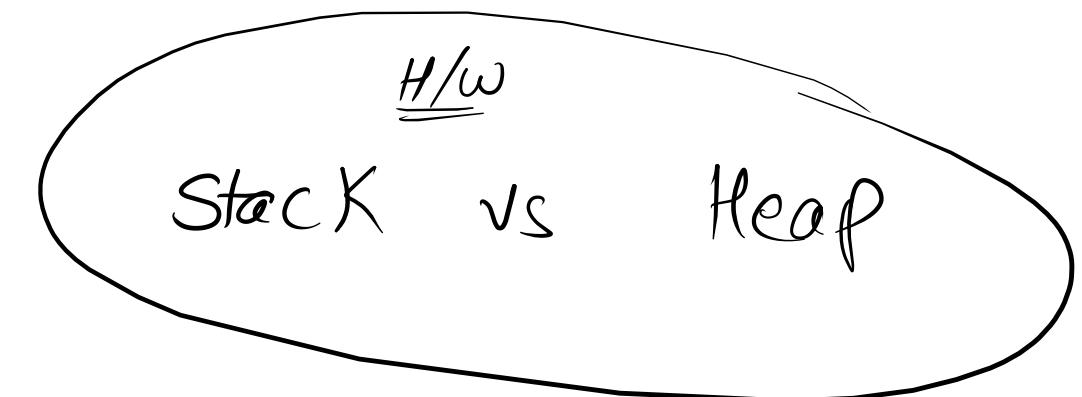
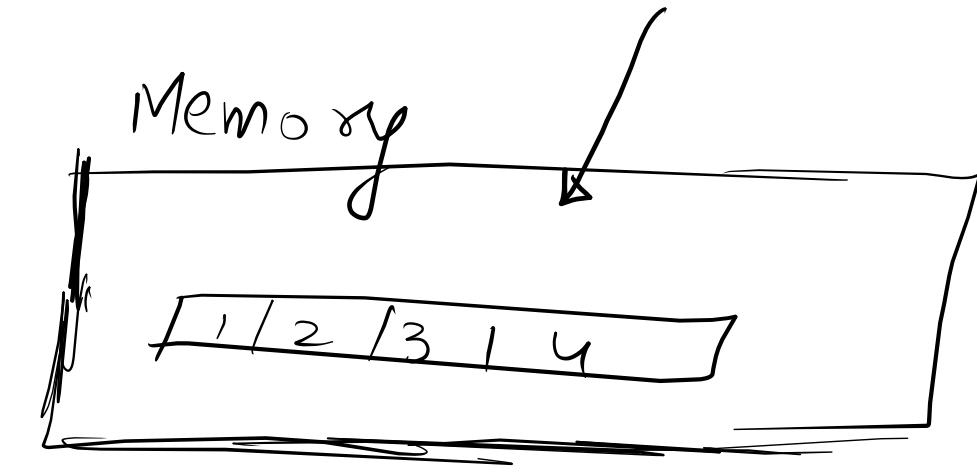


you are Neither a Men
Nor a female

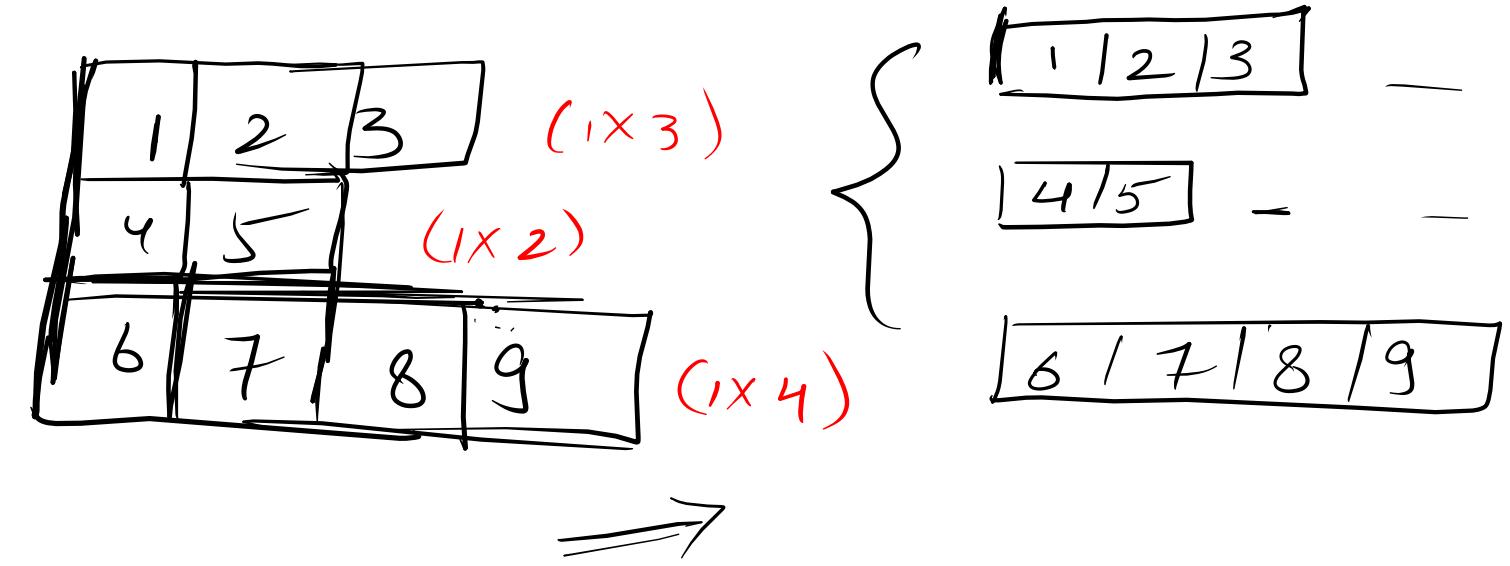
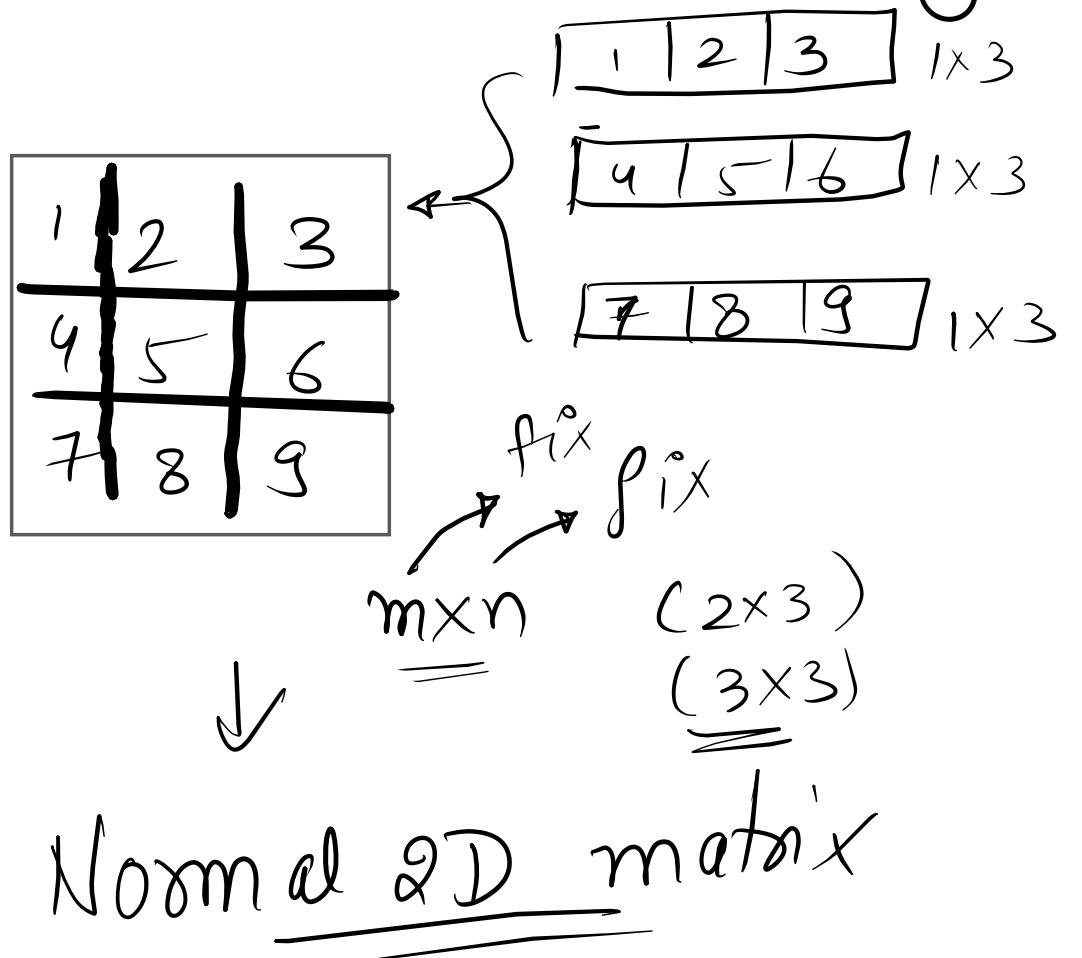
Memory Allocation of 2D - Array



1-D
arr \Rightarrow 1 1 2 1 3 1 4



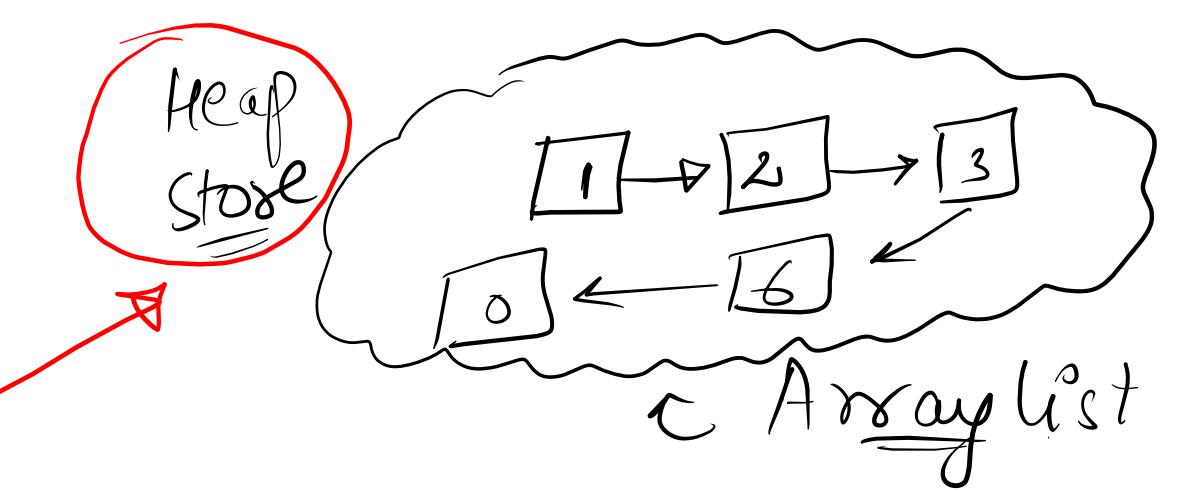
what do you mean by Jagged M.?



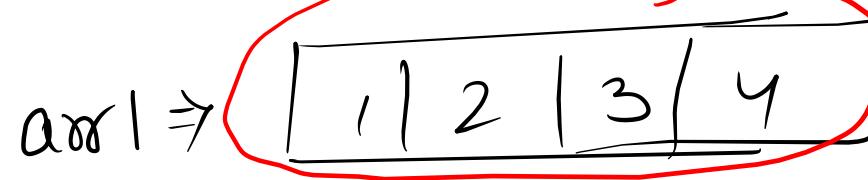
Individual row size different

Jagged Matrix

* ArrayList

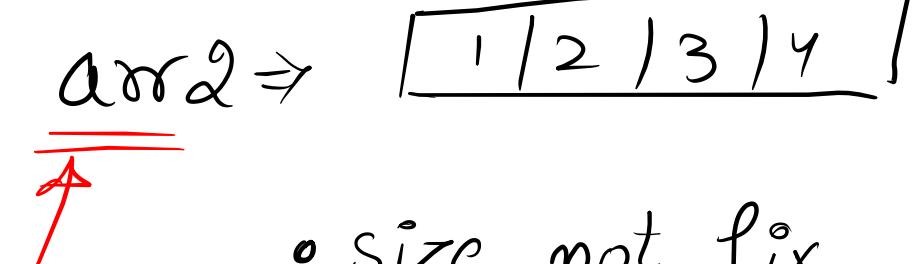


Normal 1D array



variable
store in
stack.

- Size fix
- continuous memory
-



- size not fix
- Non-Continuous memory.
-

variable
store in
stack.

ArrayList

III - - - - -

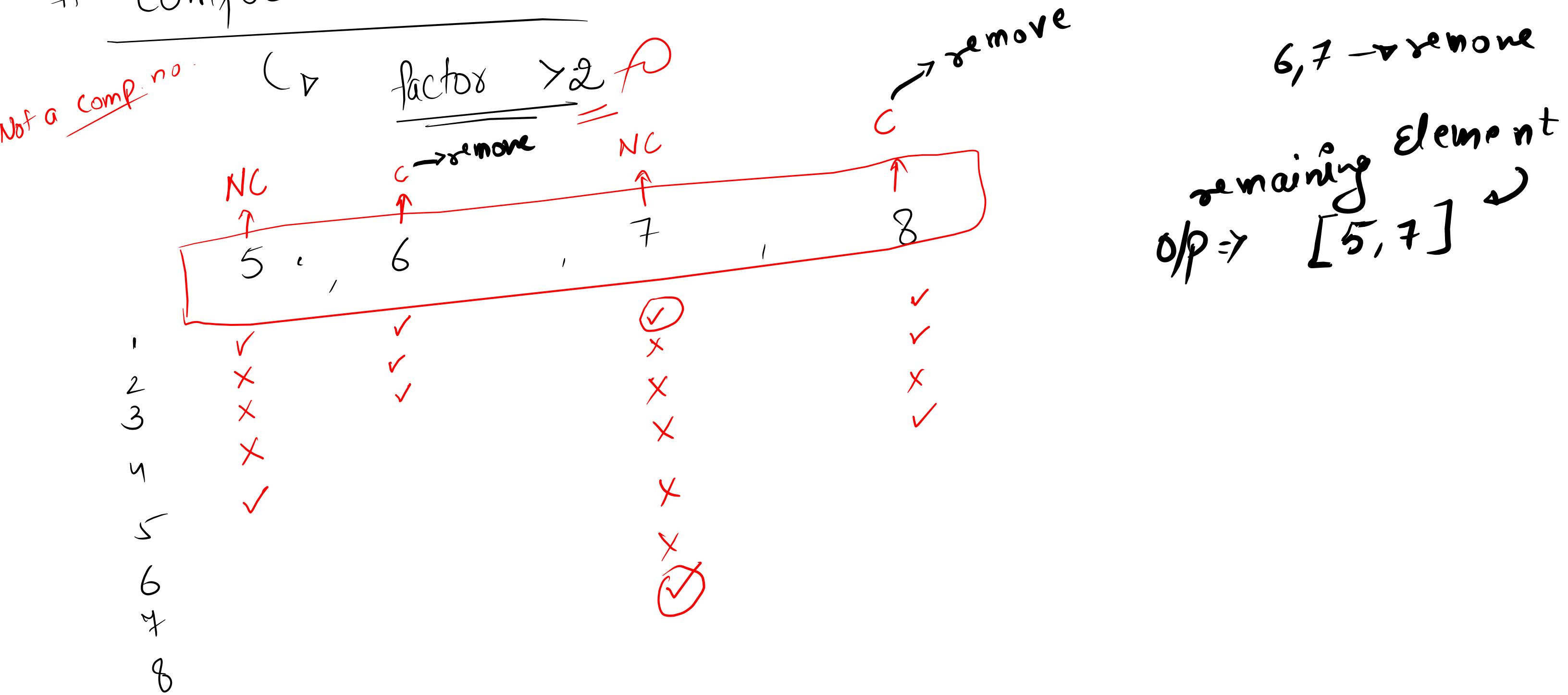
① Create * `ArrayList < Integer > list1 = new ArrayList < >();`
* `ArrayList < String > list2 = new ArrayList < >();`

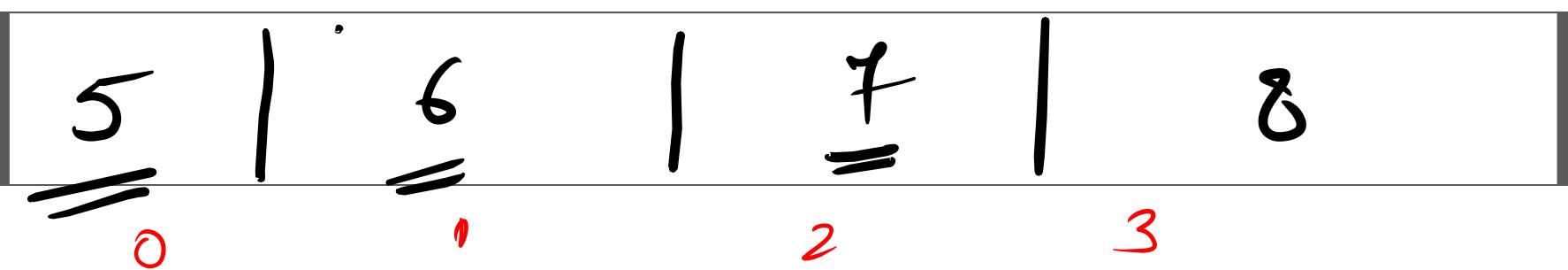
`import java.util.*;`

② operation →

`ArrayList < Integer > list1 = new ArrayList < >();` ←

Composite Numbers





$i=0$

⑤

1, 2, 3, 4, 5
✓ ✗ ✗ ✗ ✓

$i=1$

⑥

1, 2, 3, 4, 5, 6
✓ ✓ ✓ ✓ ✓

$i=2$

⑦

1, 2, 3, 4, 5, 6, 7
✓ ✓ ✓ ✓ ✓ ✓

$i=3$

⑧

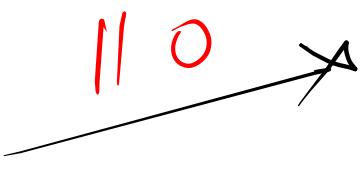
1, 8
✓ ————— ✓

5 % 1 == 0 ✓
5 % 2 == 0 X
5 % 3 == 0 X
5 % 4 == 0 X
5 % 5 == 0 ✓

for ($i=0$; $i < n$; $i++$) {
 for (int j=1 — arr[i];
 if (arr[j] == 0)
 {
 }
 }
}

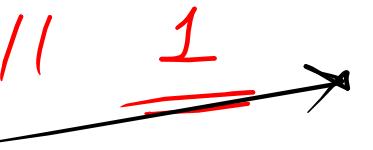
ArrayList

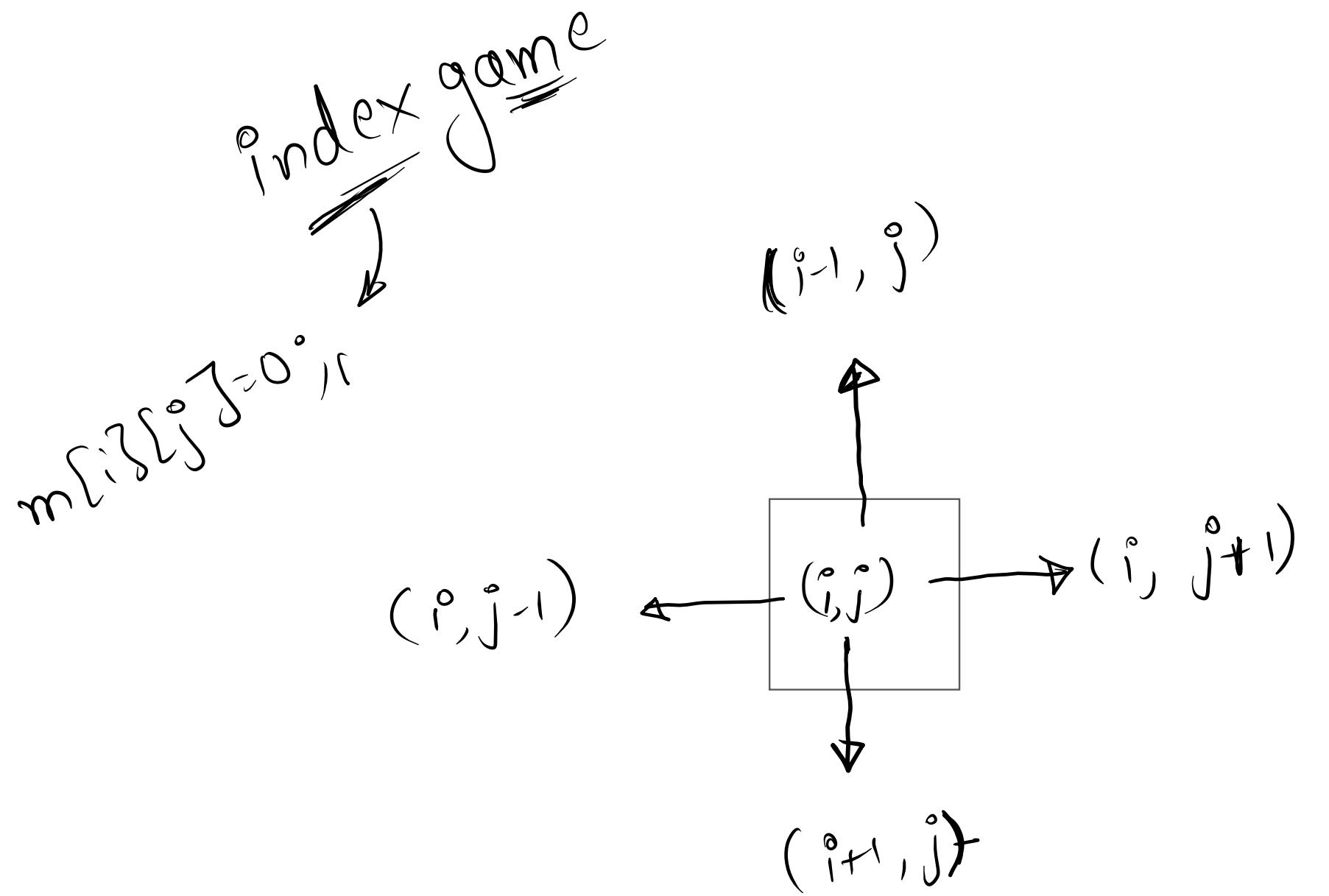
→ Create



$$\left(\rightarrow m[i][j] = 0 \right)$$

No change \downarrow in direction:





direction game

prev. dir.

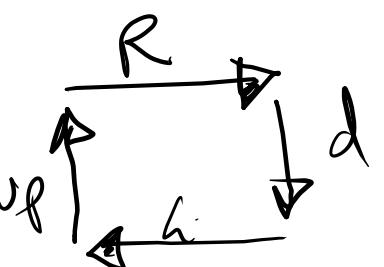
$\text{dir} = 'R'$ \Rightarrow $\text{dir} = \text{down}$

$\text{dir} = 'd'$ \Rightarrow $\text{dir} = \text{left}$

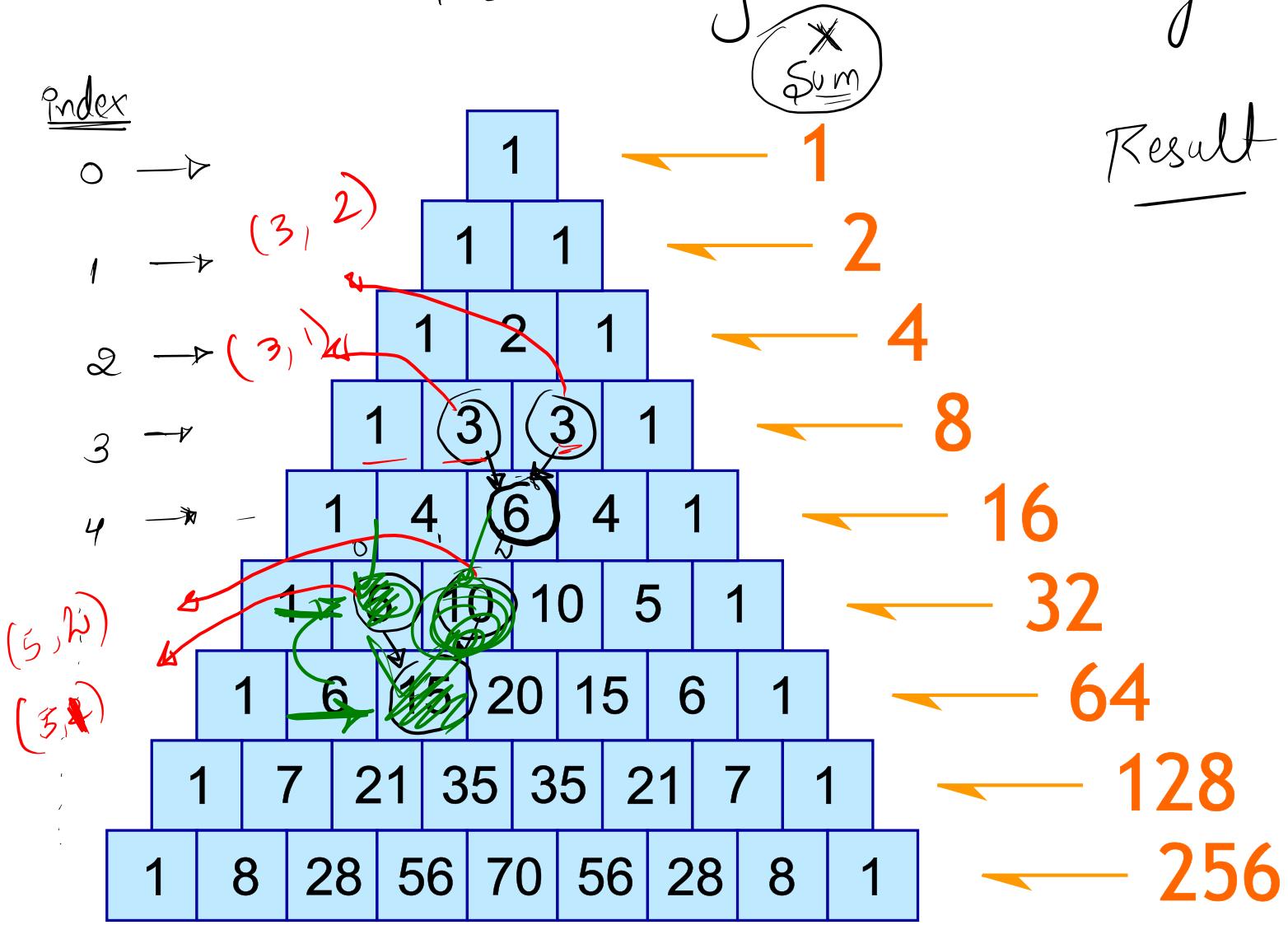
$\text{dir} = 'l'$ \Rightarrow $\text{dir} = \text{up}$

$\text{dir} = 'v'$ \Rightarrow $\text{dir} = \text{Right}$

dir updated

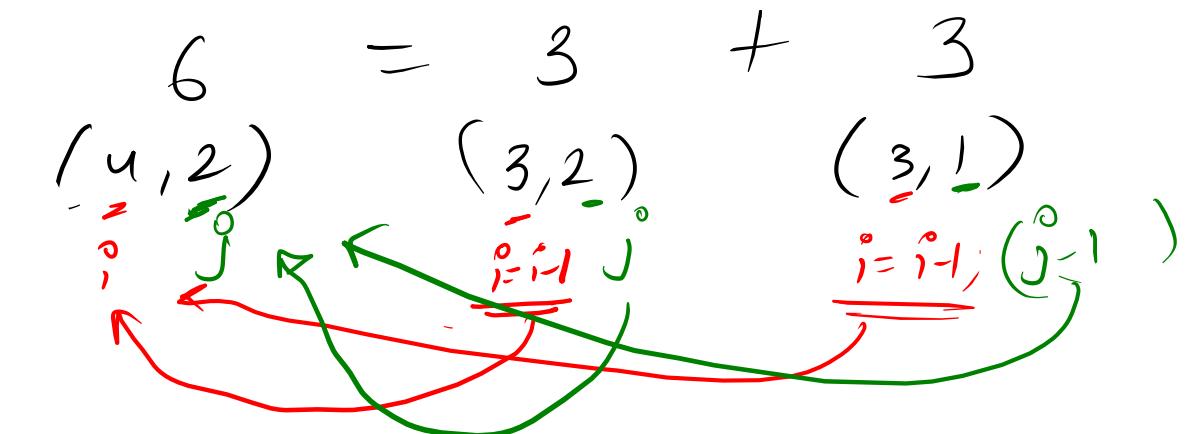


Pascal Triangle \rightarrow Actually it is sum of 2 element.



$$\text{arr}[i][j] = \text{arr}[i-1][j] + \text{arr}[i-1][j-1]$$

Pattern find



#Pascal Triangle

0	1	2	3	4
1				
1	1			
1	2	1		
1	3	3	1	
1	4	6	4	1

$$6 = 3 + 3$$

for $i=0$ to n

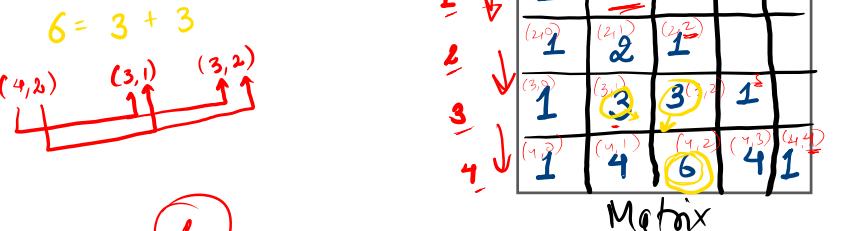
if ($j=0$)

mat[i][j] = 1;

or

if ($i=j$) {
mat[i][j] = 1;}

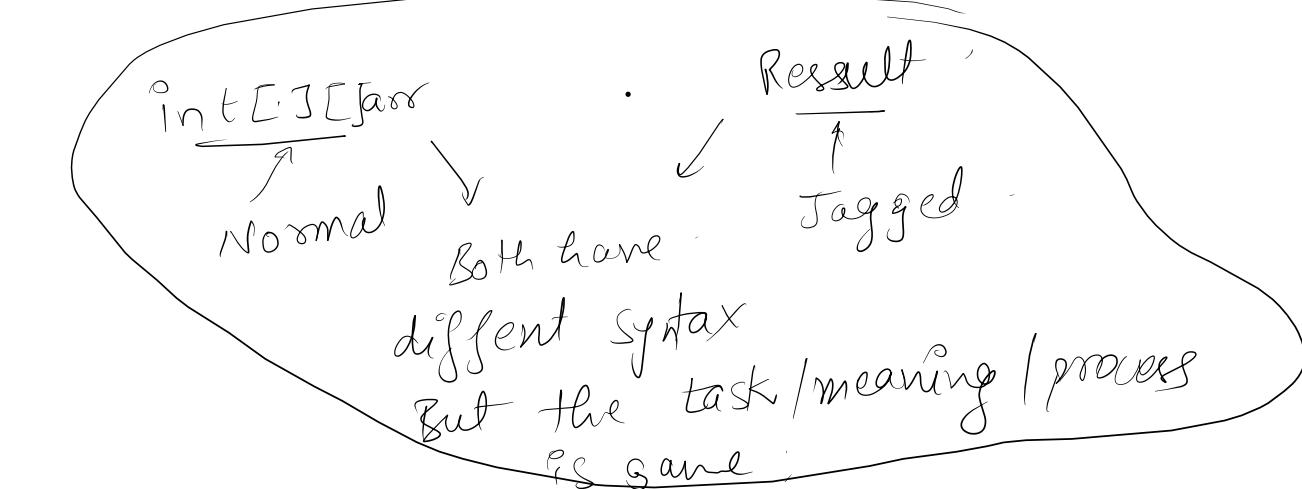
}



$i=0 \quad j=0$
 $i=1 \quad j=0, 1$
 $i=2 \quad j=0, 1, 2$
 $i=3 \quad j=0, 1, 2, 3$
 $i=4 \quad j=0, 1, 2, 3, 4$

- This is a Jagged Matrix
 - The value at each row, with starting column and last column is 1.
 - The value at particular cell is the sum of above row with same column and previous col.
- Let take a General case -
- $$\text{arr}[i][j] = \text{arr}[i-1][j] + \text{arr}[i-1][j-1]$$

`ArrayList<ArrayList<Integer>> Result = new ArrayList<>();`



for ($i=0$; $i < n$; $i++$)

for ($j=0$; $j \leq i$; $j++$) {

Result[i][j] = matrix[i-1][j] + matrix[i-1][j-1];

}

}
return Result;

Two Sum (H/w) ↗

↙ # Three sum (H/w) ↗
↓
c/w Triplet sum to X ;

Triplet Sum to X

[- - -]

$$a + b + c = \cancel{X}$$

cnt ++

return

$[0, 1, 2, 3, 4, 5, 6]$ $x = 12$

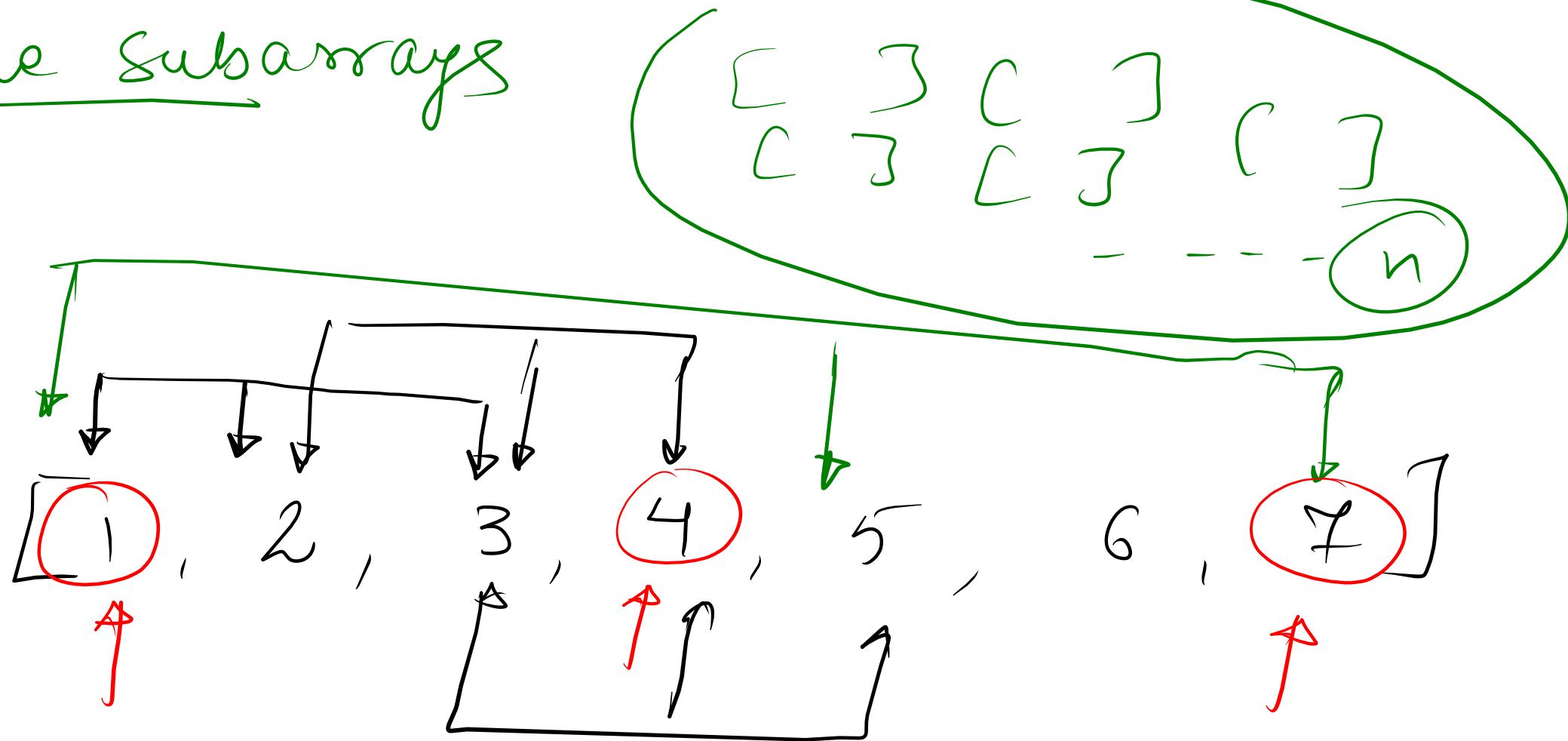
$i, \textcircled{2}, 3, \textcircled{4}, 5, \textcircled{6}, 7$

↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑

Cnt = 0

$$2 + 4 + 6 = \underline{12} \quad \underline{\text{Cnt}++}$$

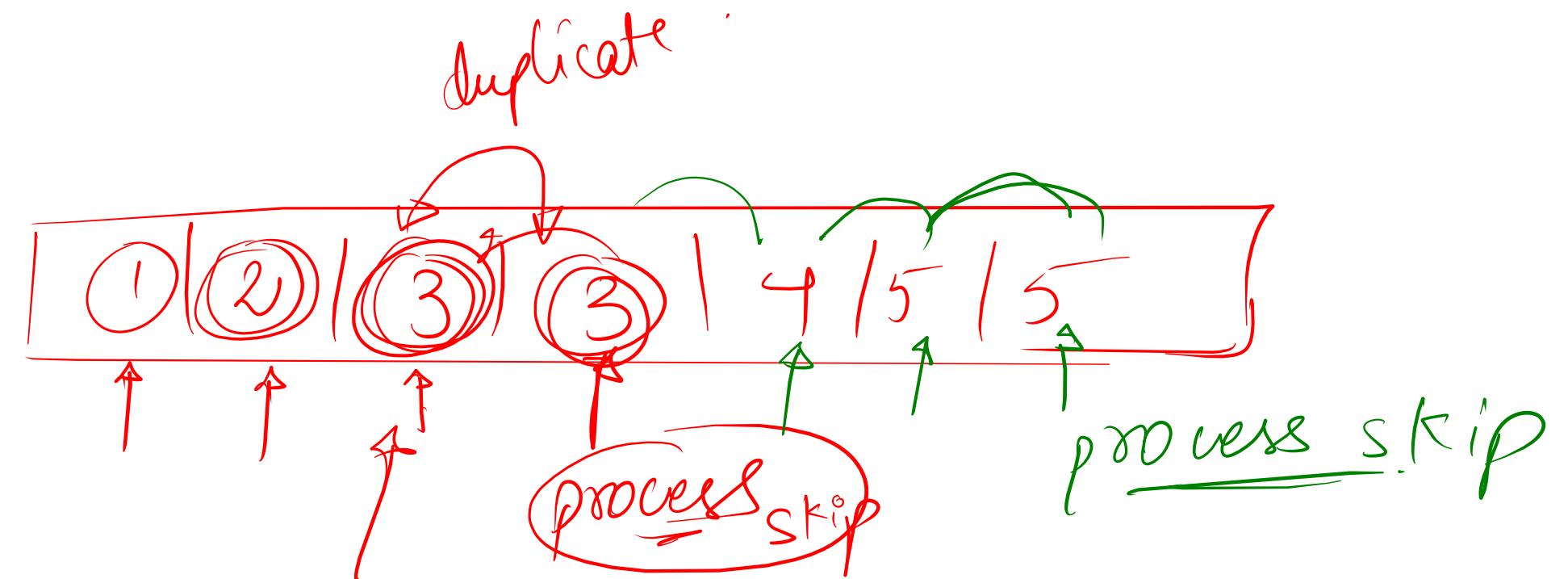
I need all the subarrays



$a | b | c$

$\Rightarrow \text{sum} += \text{arr}[i]$
 $\text{if}(\text{sum} == X) \text{cnt}++$

~~sorted~~



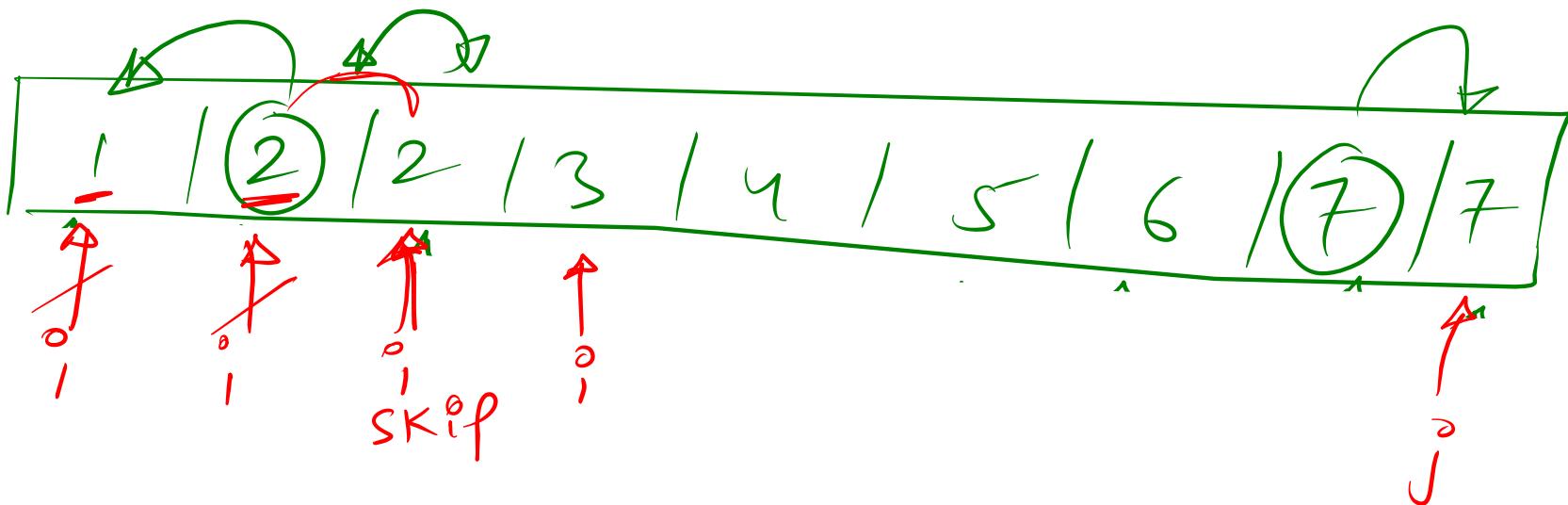
already

we process

for the element(3)

at index ~~2~~

* It is same as previous
Value



{ if ($\text{arr}[i] == \text{arr}[i-1]$) $i++$ }
if ($\text{arr}[j] == \text{arr}[j+1]$) $j--$

#

$$\frac{2}{fix} + \frac{b}{\cancel{fix}} + \frac{c}{\cancel{fix}} = 12$$

while ($j < k$) { } }

