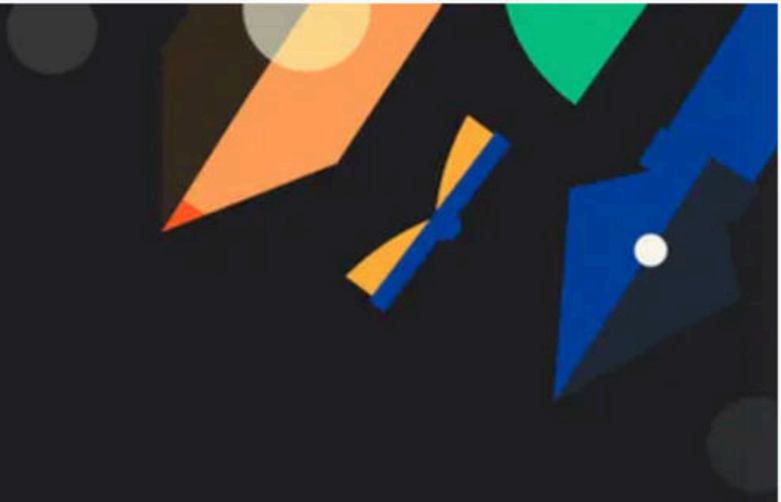


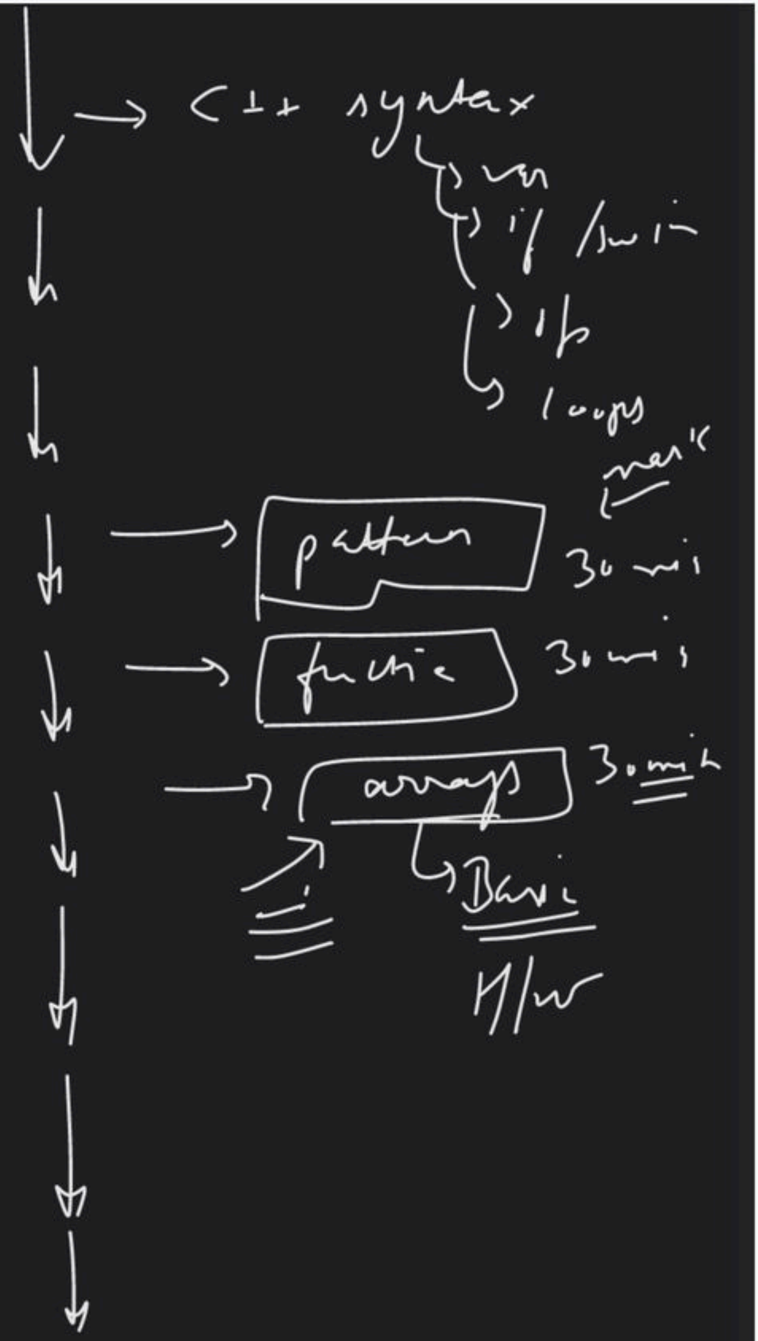
Doubt Clearing Session - Part II

Foundation Course on Data Structures & Algorithm - III

Love Babbar • Lesson 2 • Apr 24, 2022

→ Doubts :-

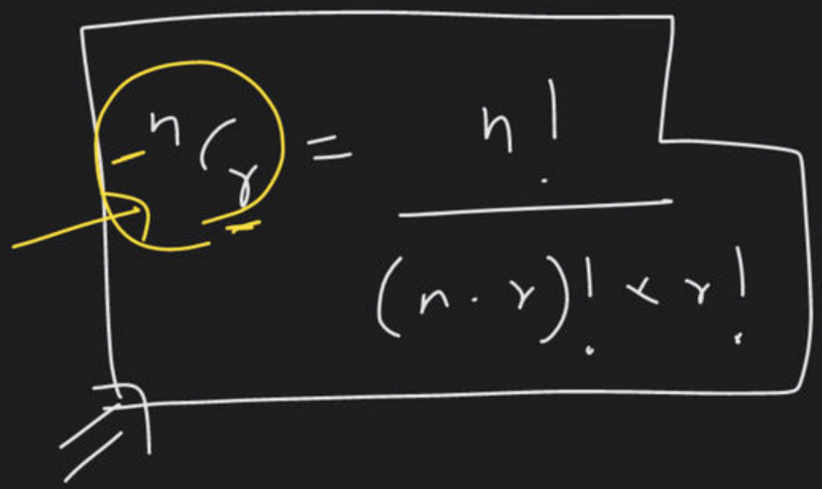
vector



→ Pascal triangle (Pattern)

r → / c ↓	0	1	2	3	4
0	1				
1	1	1			
2	1	2	1		
3	1	3	3	1	
4	1	4	6	4	1

row → 4



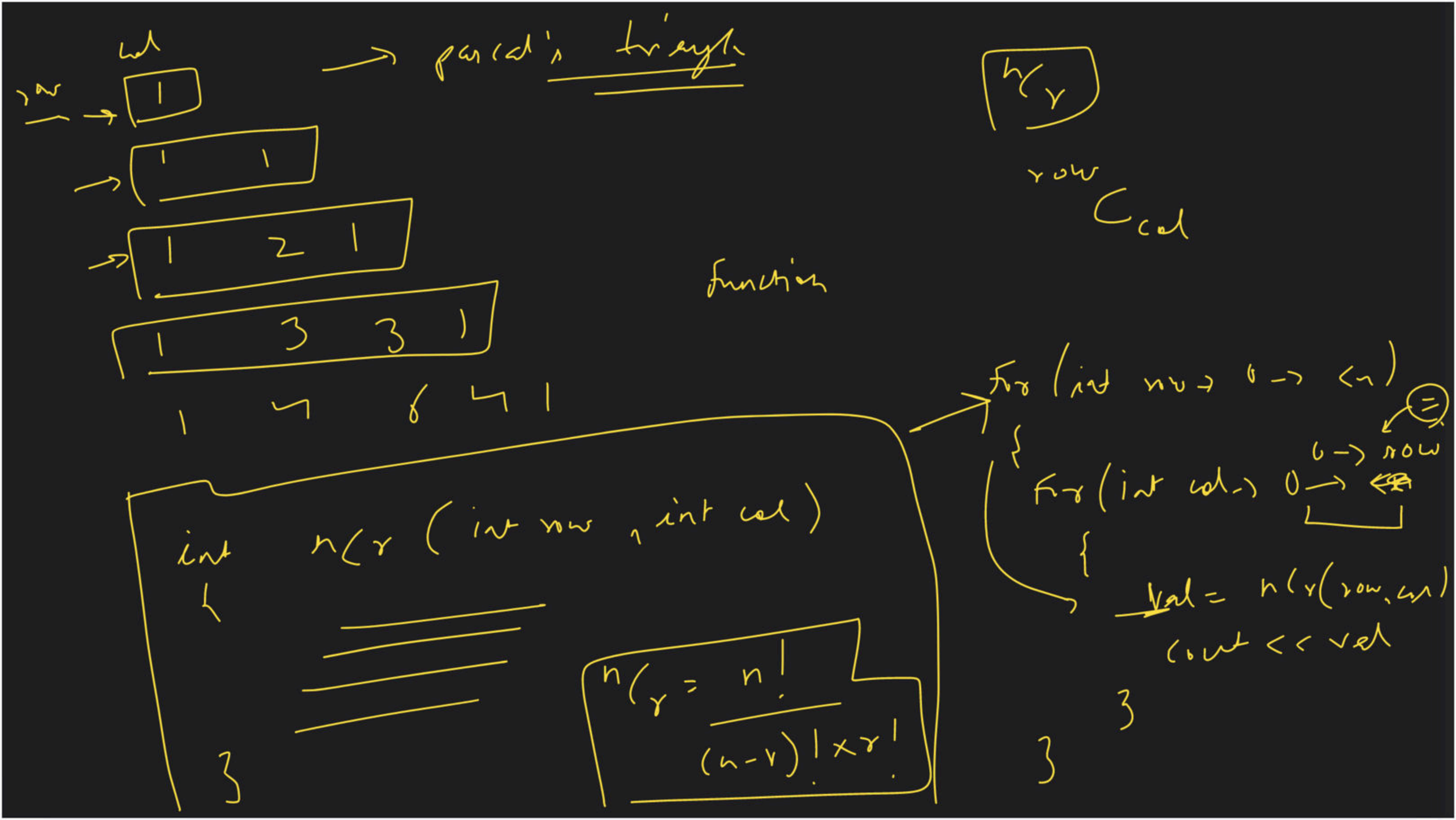
$^{0}C_0 = \frac{0!}{0! \times 0!} = 1$

$^{1}C_0 = 1$ $\quad$ $^{1}C_1 = 1$

$^{2}C_0 = 1$ $\quad$ $^{2}C_1 = \frac{2!}{1! \times 1!} = 2$

$^{4}C_2 = \frac{4!}{2! \times 2!}$ $\quad$ $\frac{4 \times 3 \times 2 \times 1}{2 \times 1 \times 2 \times 1} = 6$

$^{4}C_2 = \frac{4!}{(4-2)! \times 2!} = \frac{4!}{2! \times 2!}$


col
row
pascal's triangle
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
function
int nCr (int row , int col)
{
}
nCr = n!
(n-r)! x r!
for (int row -> 0 -> <n)
{
for (int col -> 0 -> row
{
Val = nCr(row,col)
cout << val
}
}
nCr
row
Ccol

task → int nCr (int n, int r)

{

function

2 min Break;

int factorial (int n)

{

}

$nCr = \frac{(n)!}{(n-r)! \times r!}$

Limitation

→ Big no ke liye factorial nahi laa paenge

```
int factorial (int n)
{
    int ans = 1
    for (i = n; i >= 1; i--)
    {
        ans = ans * n
    }
    return ans;
}
```

```
int fact (int n, int v)
{
    ~~int ans = fact (n~~
    int val1 = fact (n)
    int val2 = fact (n-v)
    int val3 = fact (v);
    int ans = ( val1 )
              ( val2 * val3 )
    return ans;
}
```

Int Overflow

→ 2 D array

→ Butterfly pattern

n = 3

Logic

for (int row → 1 ... <= n)
{
// each row
star → for (1 → row)
space → for (1 → n-row)
}

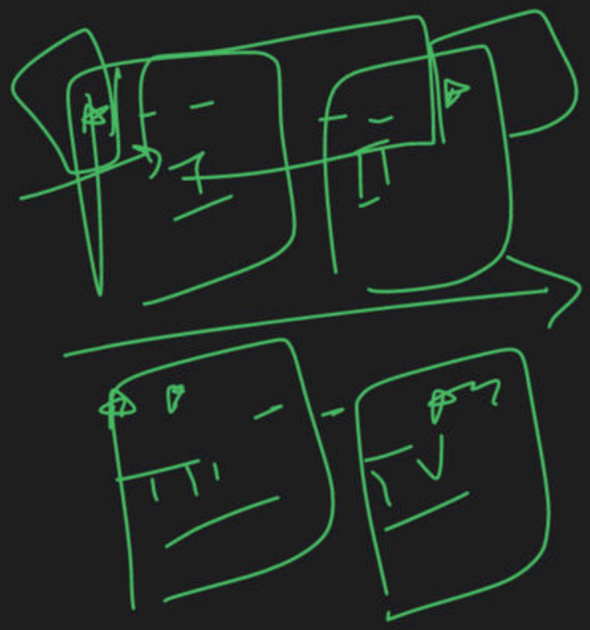
	star	space
1st row	1	2
2nd row	2	1
3rd row	3	0
	(row)	(n-row)

n = 3

row 1

2

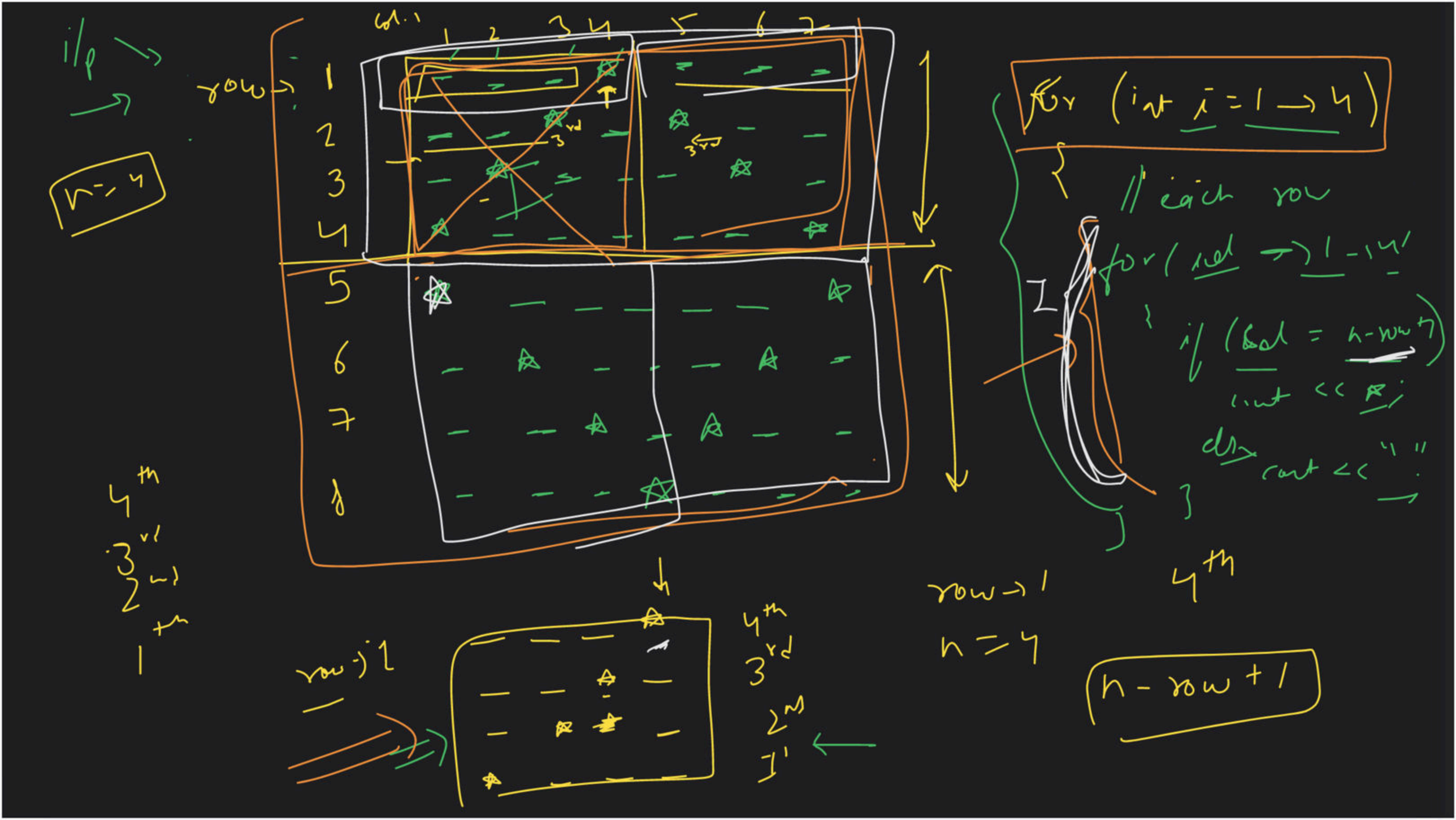
n-row

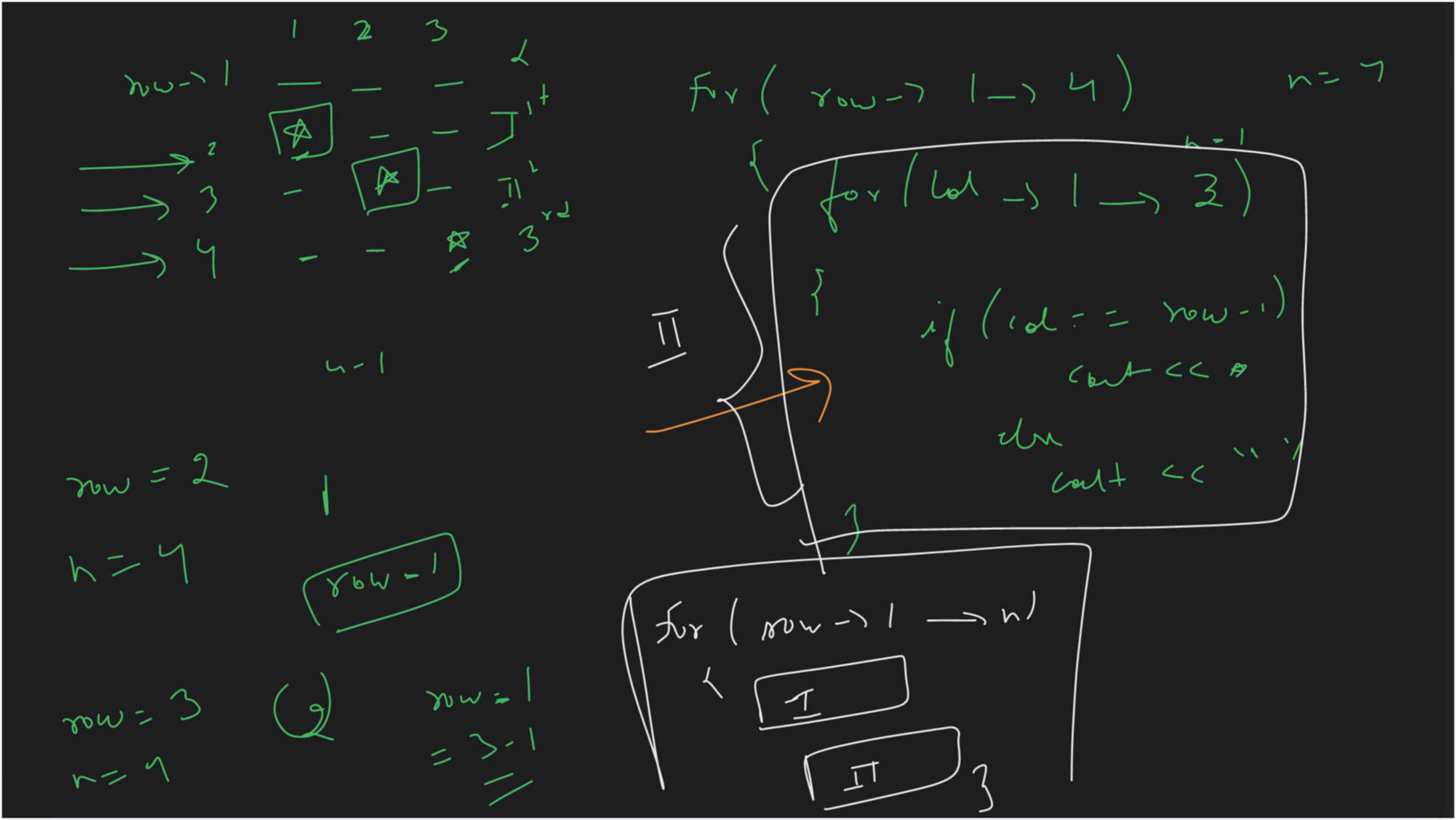


```
for (int row--> 1  <=n)
{
    star  -> for { }
    space -> for { }
    star  -> for { }
}
cout <<endl
For( row--> 1 --> <=n)
{
    [ ]
    [ ]
    [ ]
}
```

T.C

S.C


i/p
n = 4
row -> 1 2 3 4 5 6 7
1
2
3
4
5
6
7
8
for (int i = 1 -> 4)
{
// each row
for (int j -> 1 - i)
{
if (j == n - row)
cout << "*";
}
cout << "\n";
}
4th
3rd
2nd
1st
row -> 1
n = 4
row - 1
4th
3rd
2nd
1st
n - row + 1

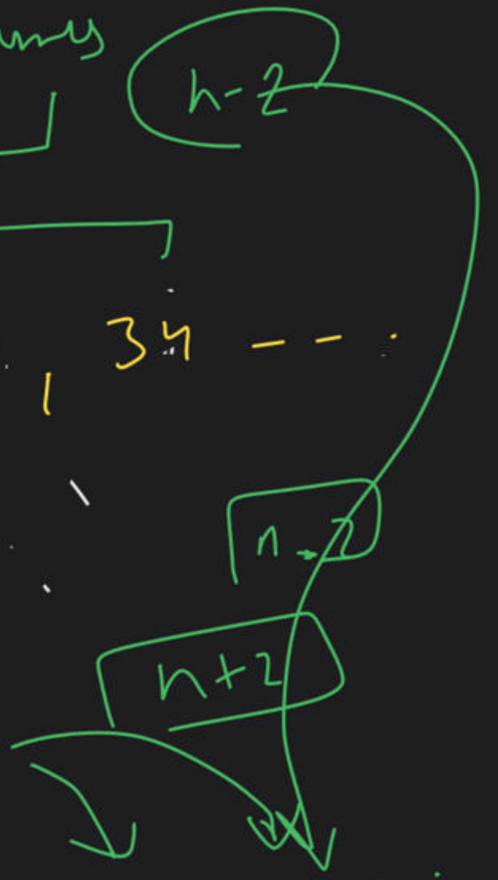

1 2 3
row->1
2
3
4
4-1
row = 2
n = 4
1
row-1
row = 3
n = 4
row = 1
= 3-1
= 2
for (row -> 1 -> 4)
n = 4
{
for (col -> 1 -> 3)
}
if (col == row-1)
cout << *
else
cout << " "
I
II
for (row -> 1 -> n)
{
I
II
3

→ functions :-

fib series

n terms

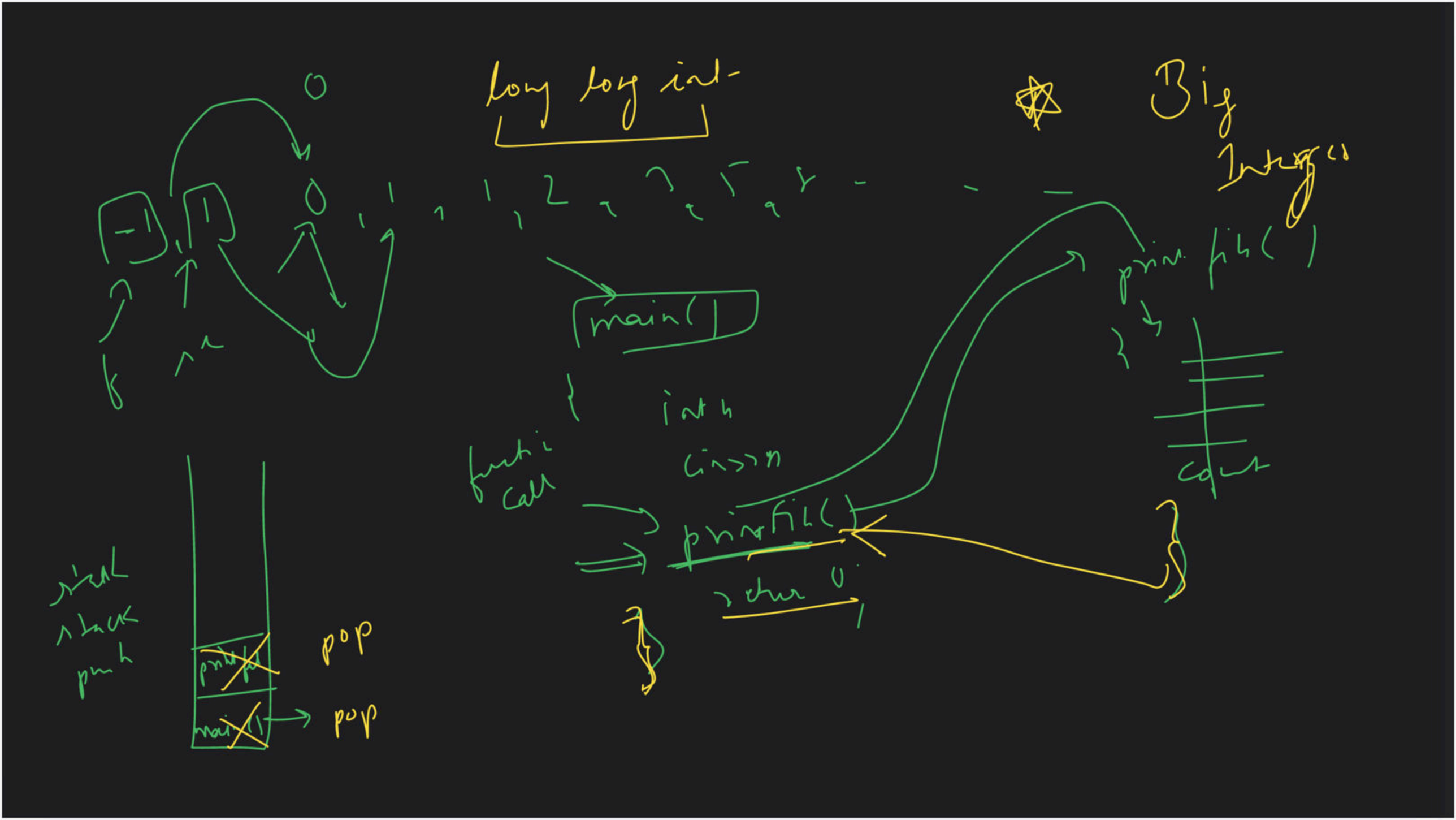
0, 1, 1, 2, 3, 5, 8, 13, 21, 34 - - -




```
void printfibonacci ( int n)
{   int first = 0;
    int second = 1;
    cout << 0 << " " << 1 << " ";
```

2 prints

```
{ for( int i=1; i<=n-2; i++)
    int sum = first + second;
    cout << sum << " ";
    first = second;
    second = sum;
}
```


long long int
Big Integer
main()
int n
cin>>n
fact(n)
return 0;
pop
pop

int fun2()

{

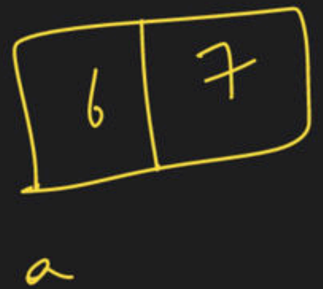
return 0;

}

int n = 6

n

2 -> pair<int, int> a



a

⇒ Gagan :- Writing

16 weeks

```
int xyz (int n)
{
  :
  :
}  luvbabbar
```

→ functions

↳

return type

int / char / void / string / float

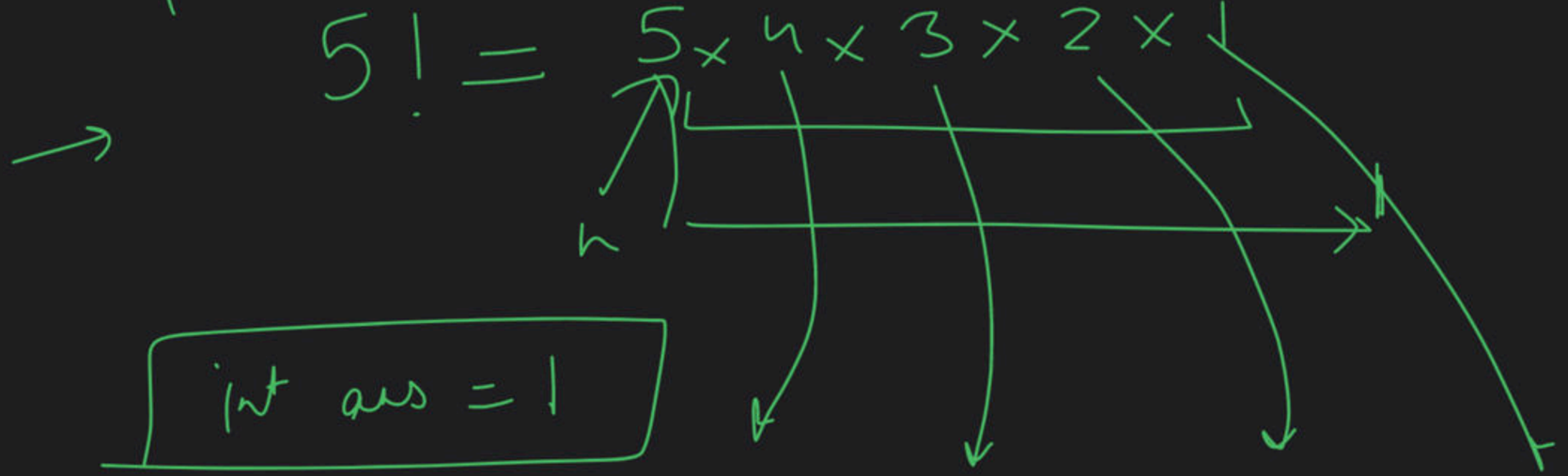
function name ( )

{

}

n = 5

$5! = 5 \times 4 \times 3 \times 2 \times 1$



int ans = 1

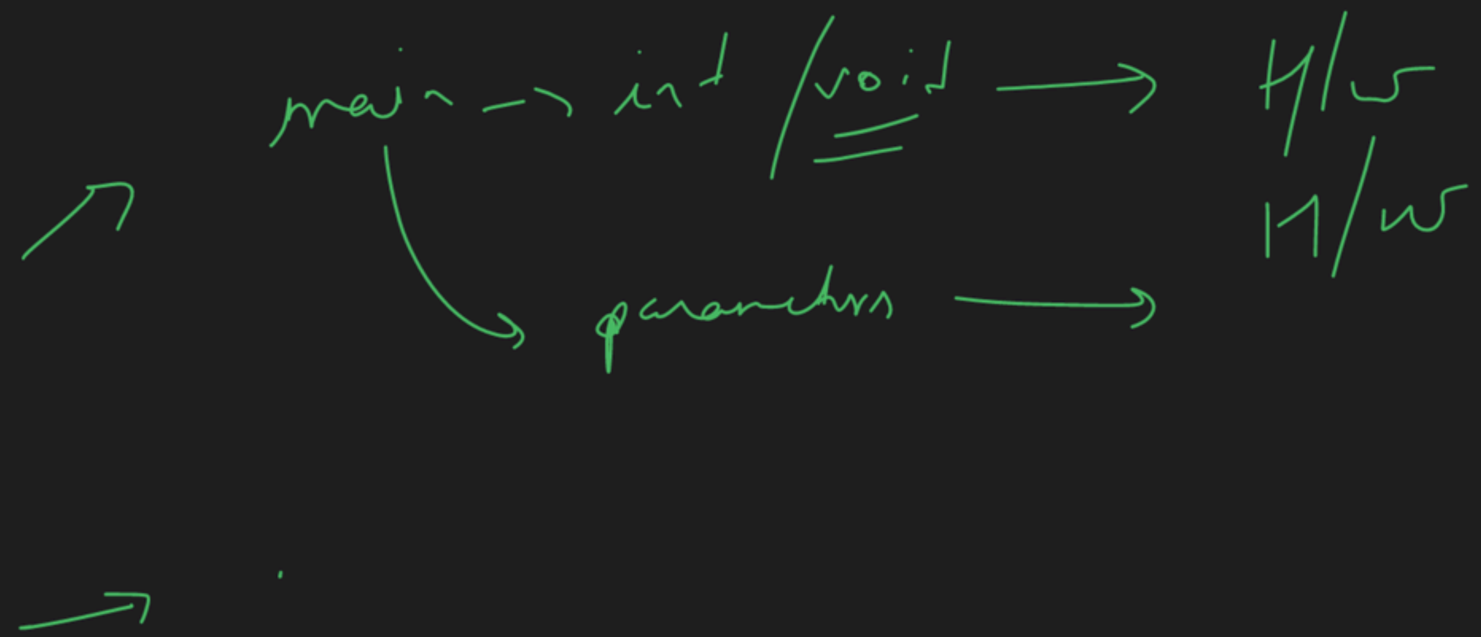
ans : $1 \times 5 = 5 \times 4 = 20 \times 3 = 60 \times 2 = 120 \times 1 = 120$

~~for(int i = n ; i >= 1 ; i--)~~

```
{
    ans = ans * i
}
```

void xyz()

```
{
// No value to return
}
```


main -> int / void
parameters

```
int f = -1
int s = 1
for (int i=0; i<n; i++)
{
    int sum = f + s;
    cout << sum;
    f = s;
    s = sum;
}
```

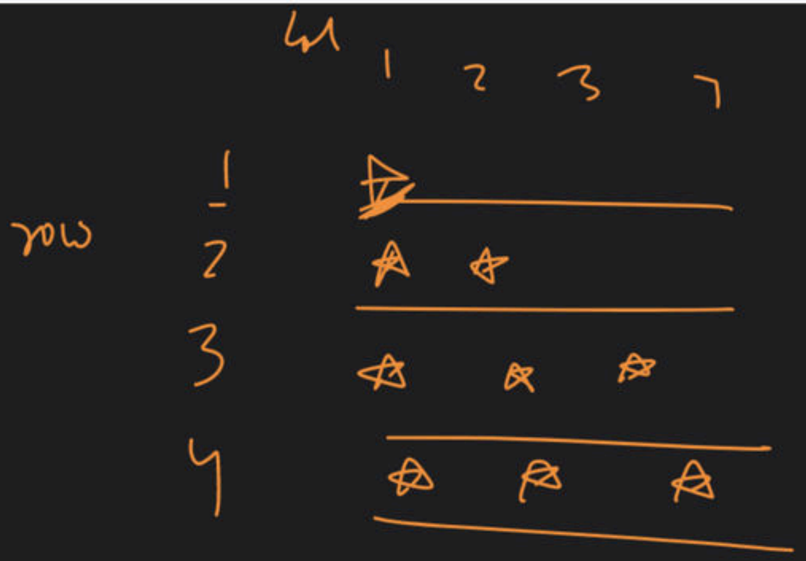
n = 5

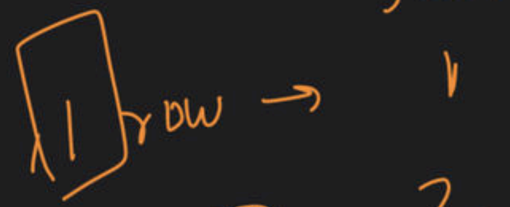
i = 0
0 < 5 → T
i = 1
1 < 5 → T
i = 2
2 < 5 → T
i = 3
3 < 5 → T
i = 4
4 < 5 → T
i = 5
5 < 5 → F

o/p → 0, 1, 1, 2, 3

→ doubts :-

Pattern






```
for (int row 1 → 4)
{
    for (int col → 1 → row)
    {
        cout << *
    }
}
```

n = 4

	space	star
1 row	3	1
2 row	2	2
3 row	1	3
4 row	0	4
	n - row	row

Bhot

2 Σ+

23

18

59

D/H

Google

Hint

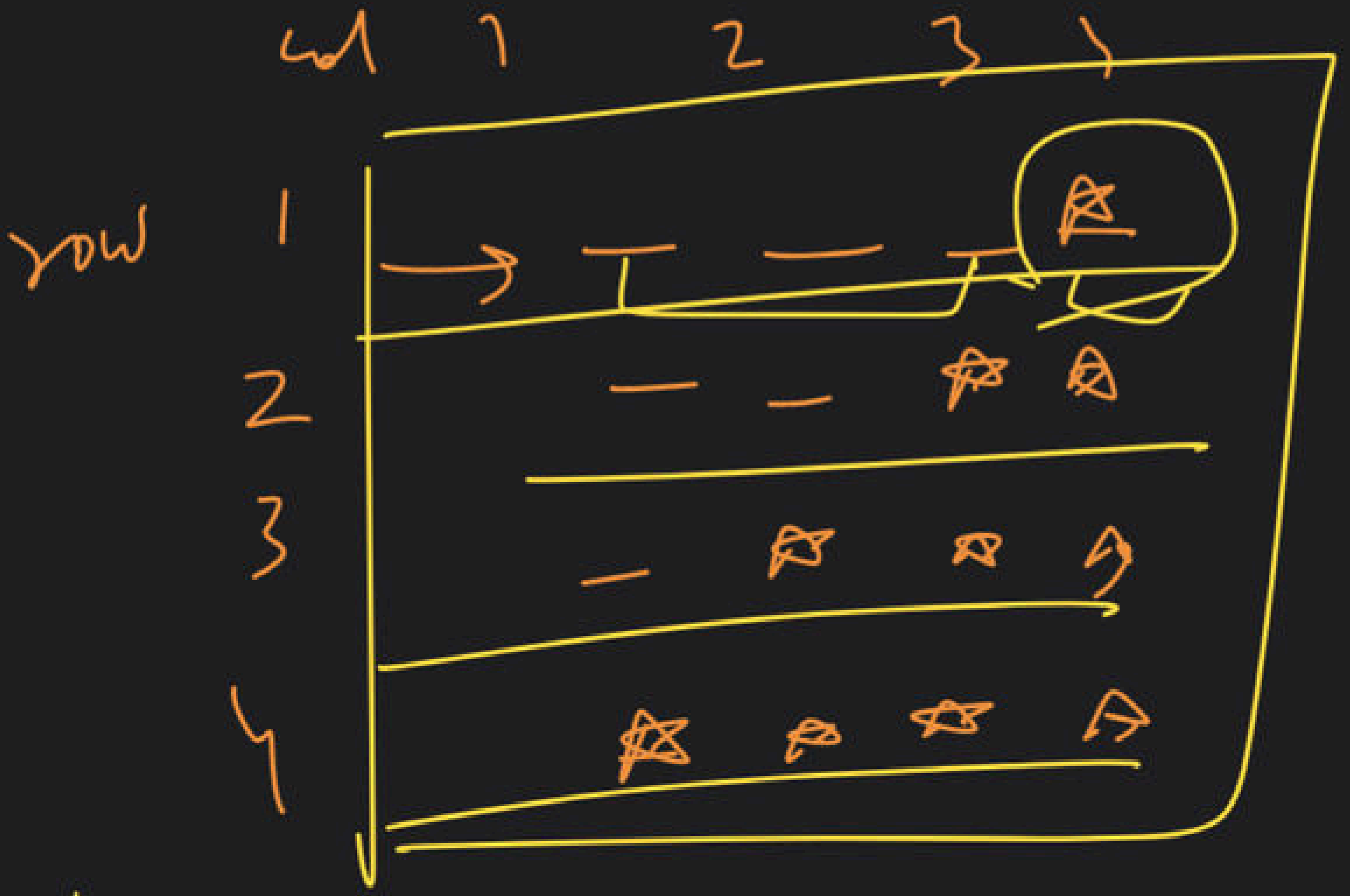
Dimaag

row = 1

n = 4

3

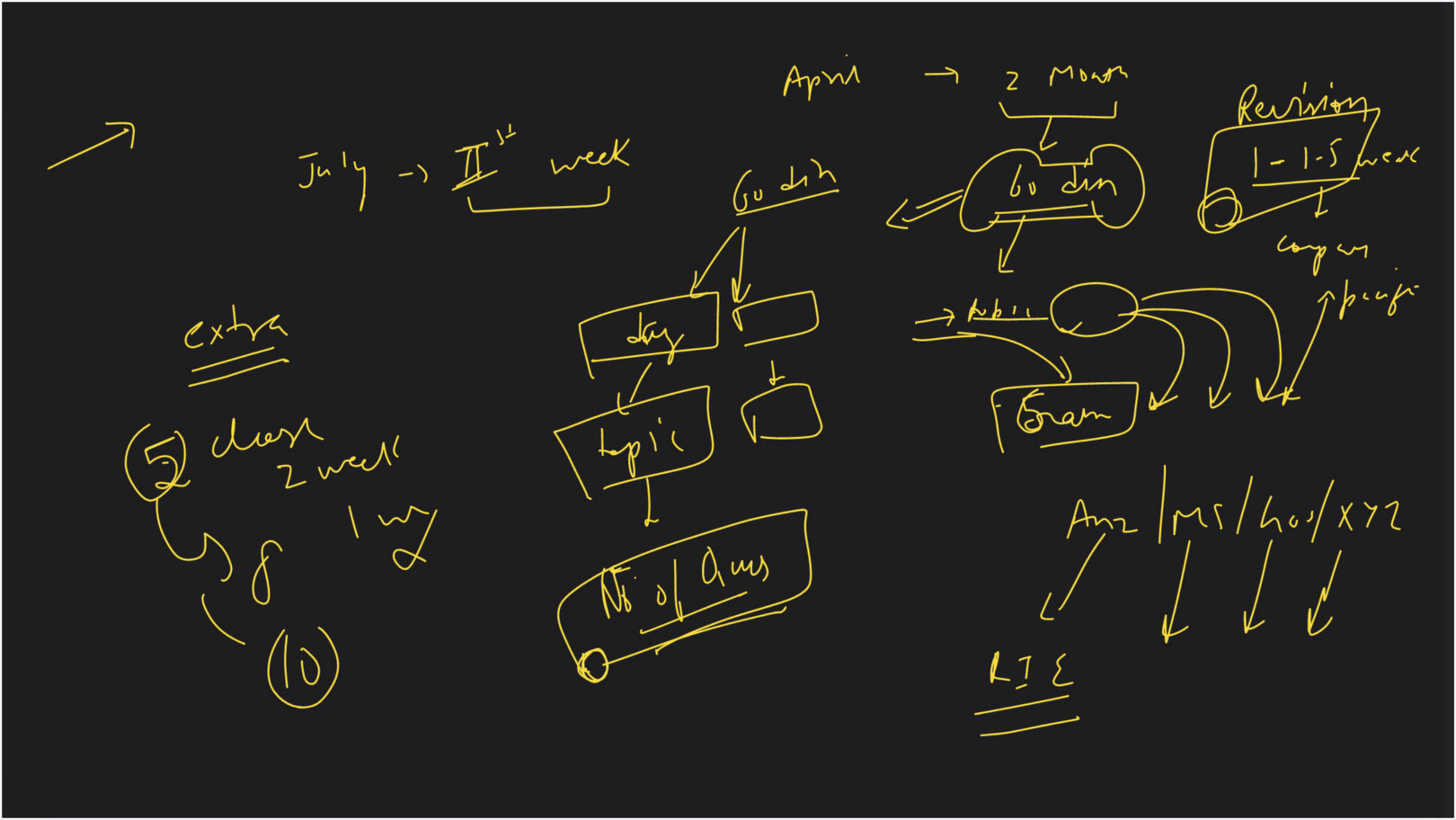
n - row

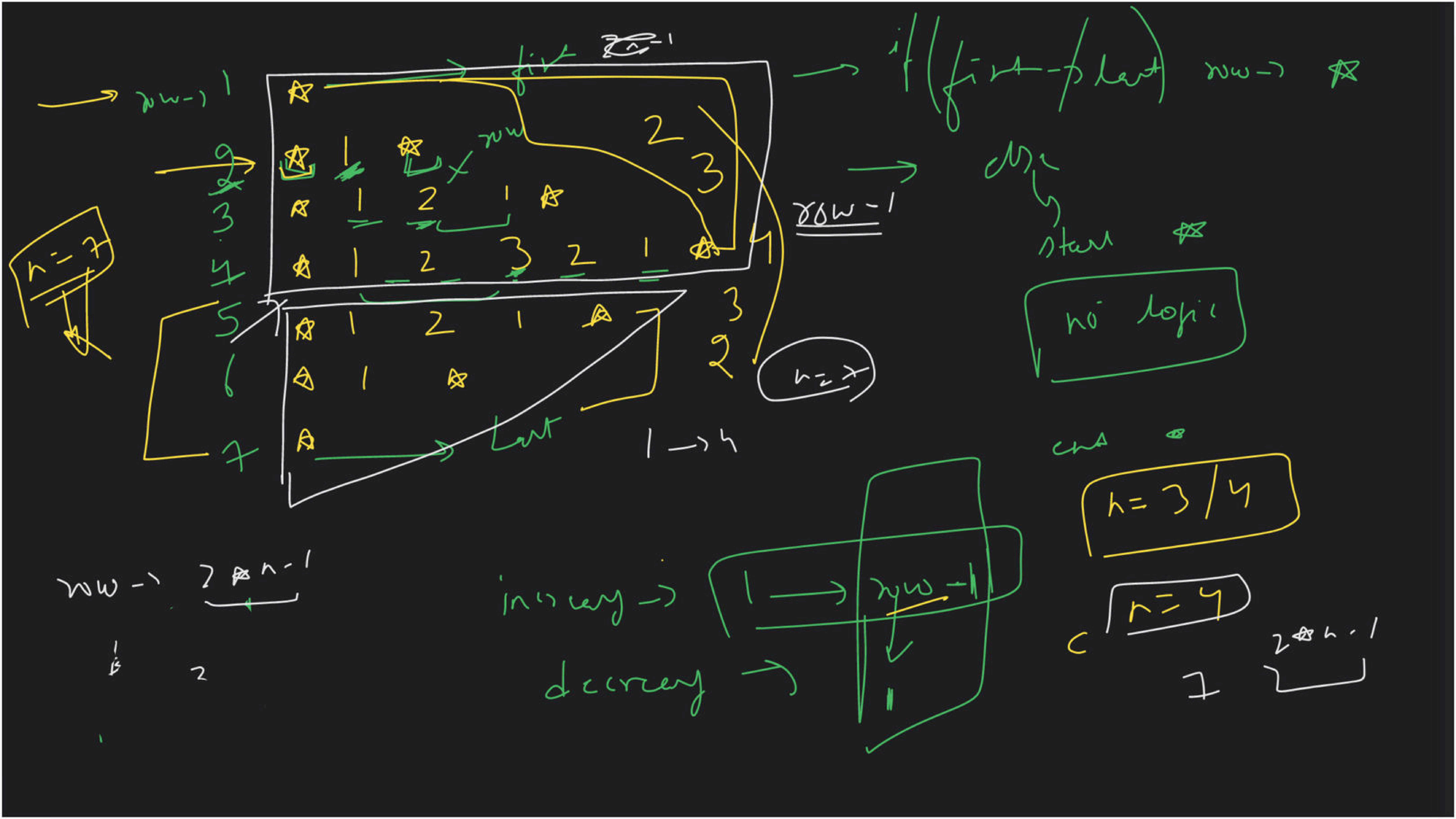


0 < n

1 <= n

```
for (int row -> 1 ___ <= n)
{
    // each row
    for (int col -> 1 -> n-row)
    { cout << " "; }
    for (int col -> 1 -> row)
        cout << *;
}
```


April → 2 Months
60 days
Revision
1-1-5 weeks
July → II 1st week
60 days
day
topic
No of Qns
Extra
5 days
2 weeks
1 hr
10
Aptitude
Gram
Company specific
Any MS/Gov/XYZ
RTE



row -> 1
first
row -> 2*n-1
if (first -> last) row -> ☆
else
row - 1
start ☆
no logic
end ☆
n = 3 / 4
n = 4
n = 7
Last
1 -> 4
increasing -> 1 -> row - 1
decreasing ->

r:7

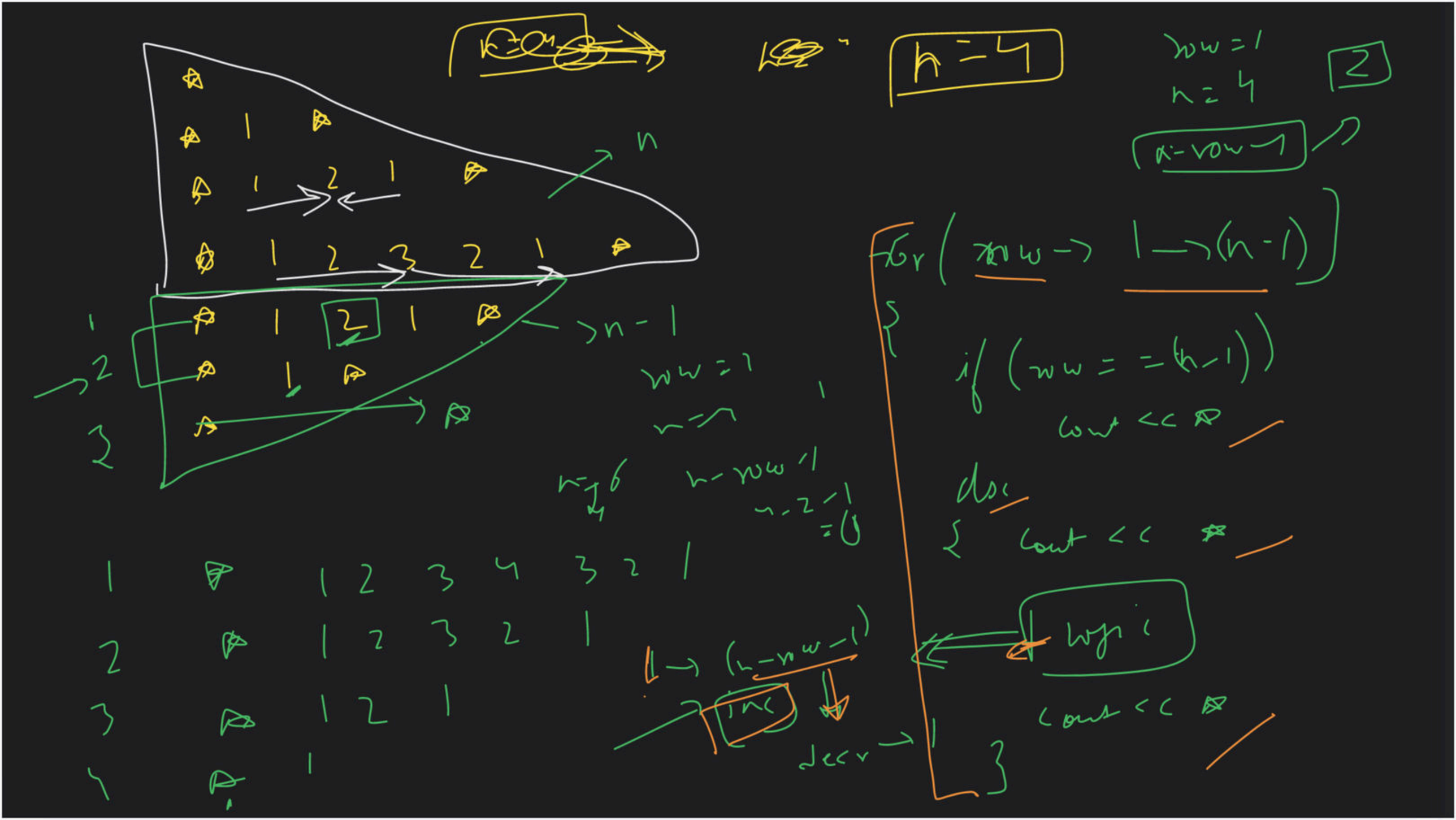
```
for (row= 0 -> <n)
{
    if ( row==0 || row==n-1 )
    {
        cout << "*";
    }
    else
    {
        cout << "*";
        Logic
        cout << "*";
    }
}
```

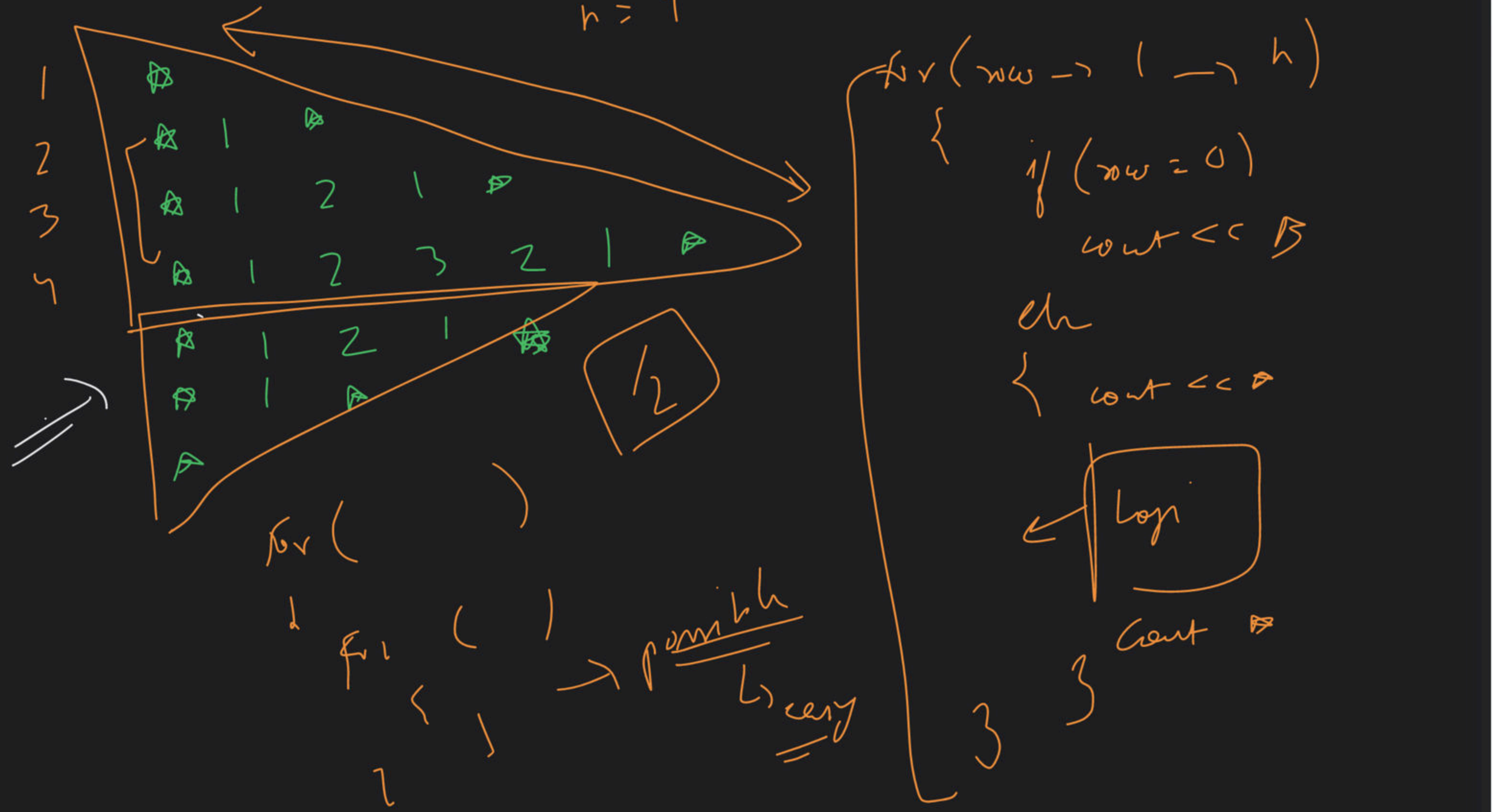
1 -> row-1 -> 1

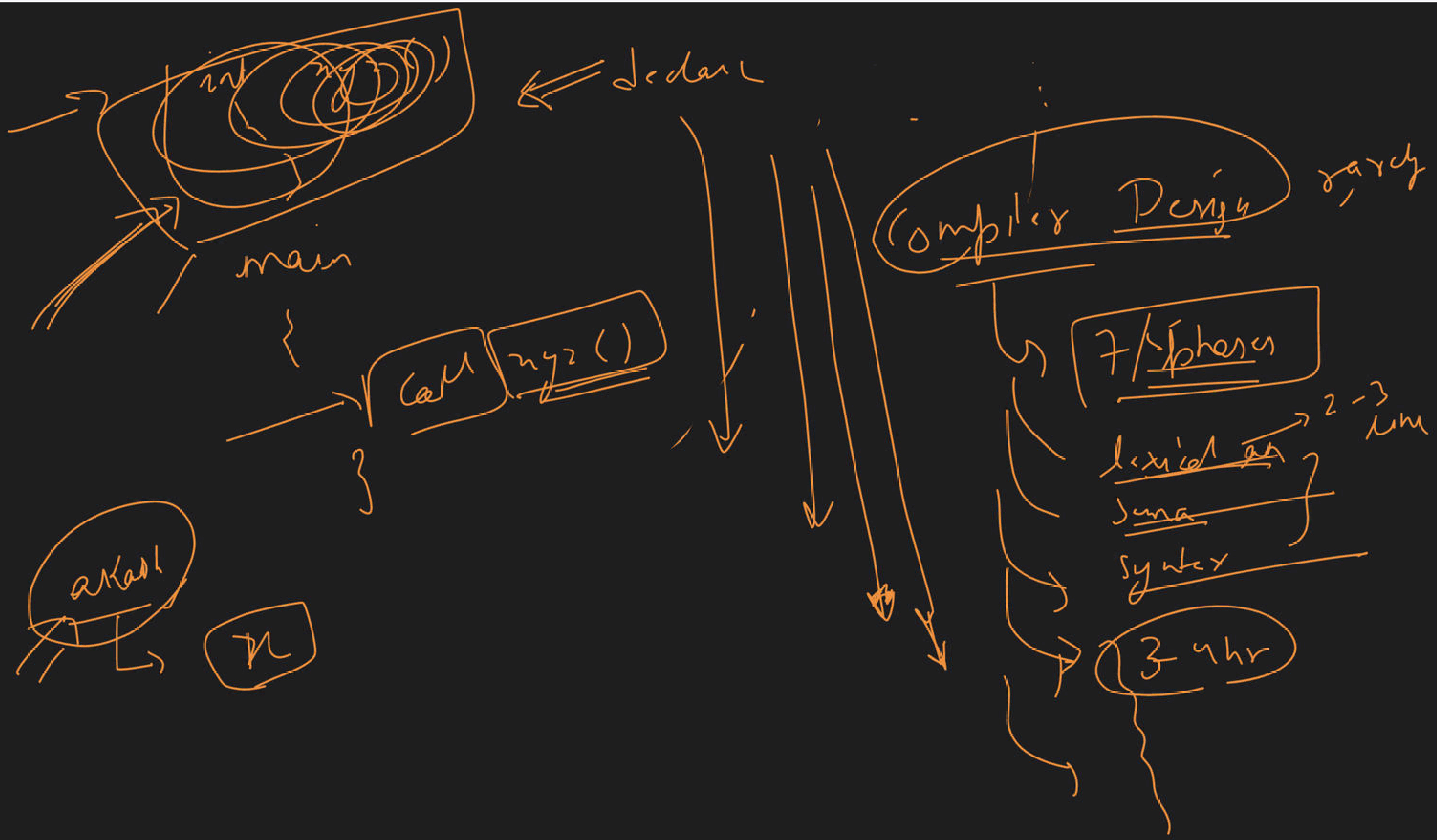
in dec

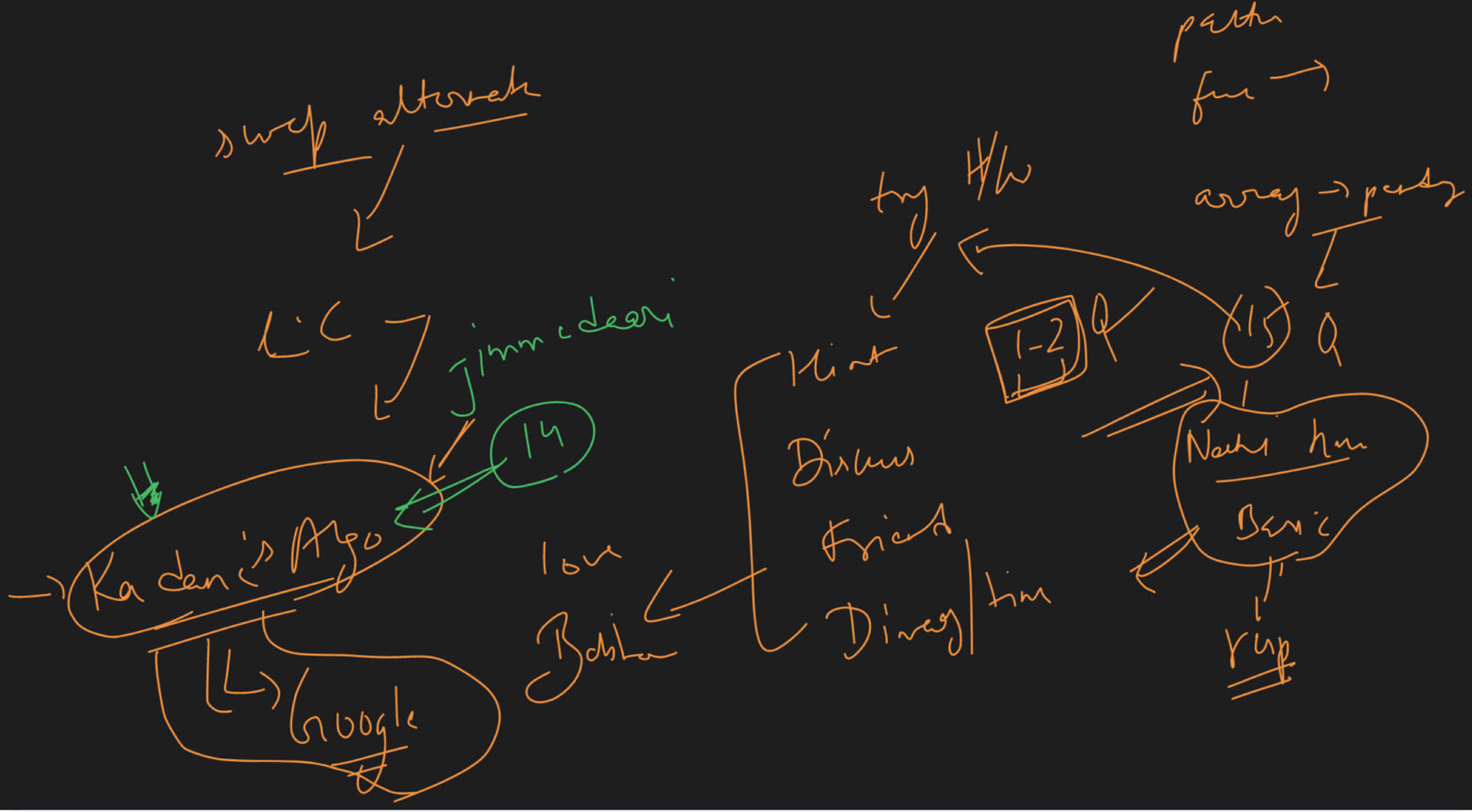
Logic

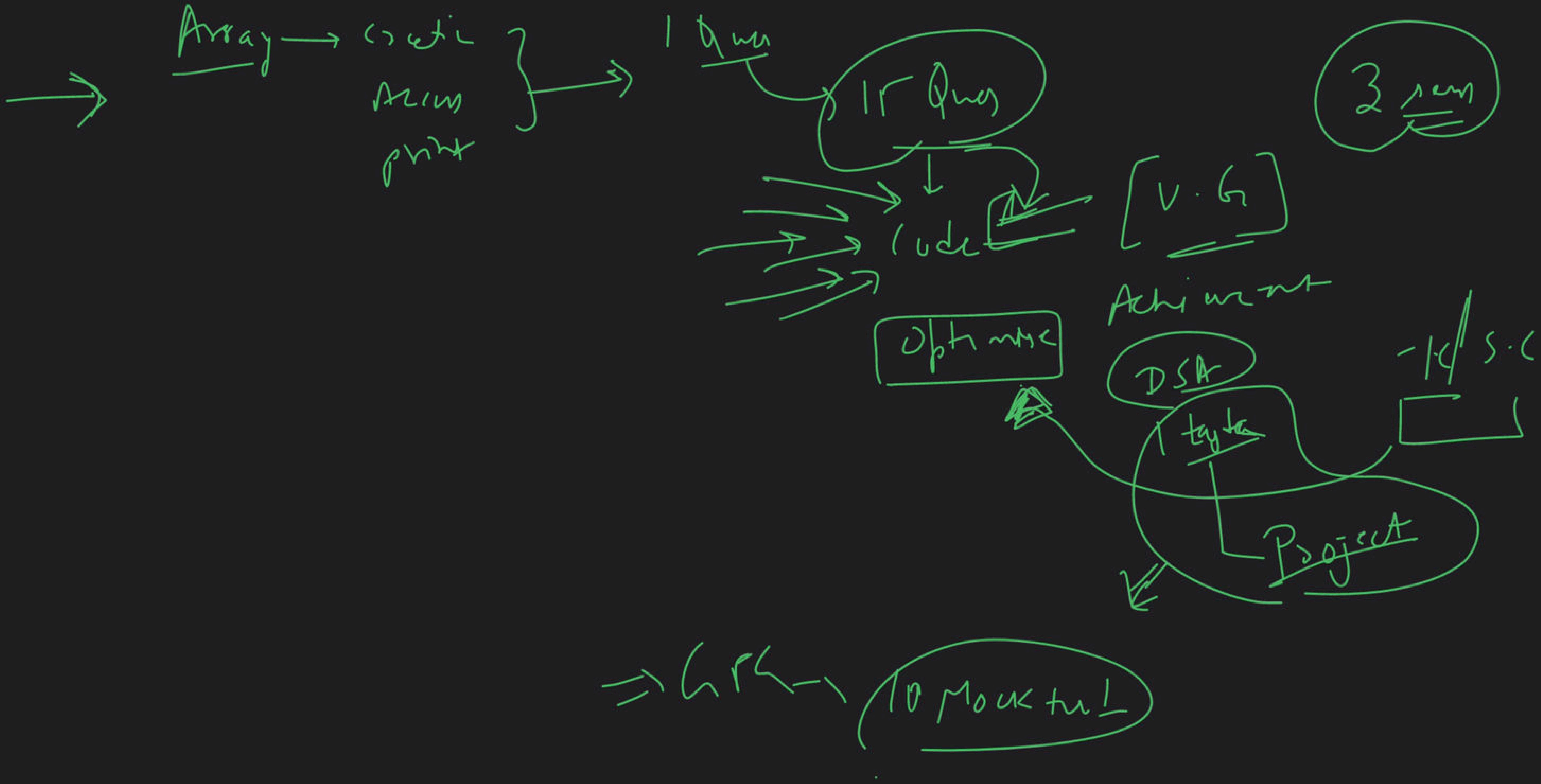
```
int counter=1
while (counter!=row-1)
{
    cout << counter
    counter++
}
while (counter >0)
{
    cout <<counter
    counter--;
}
```

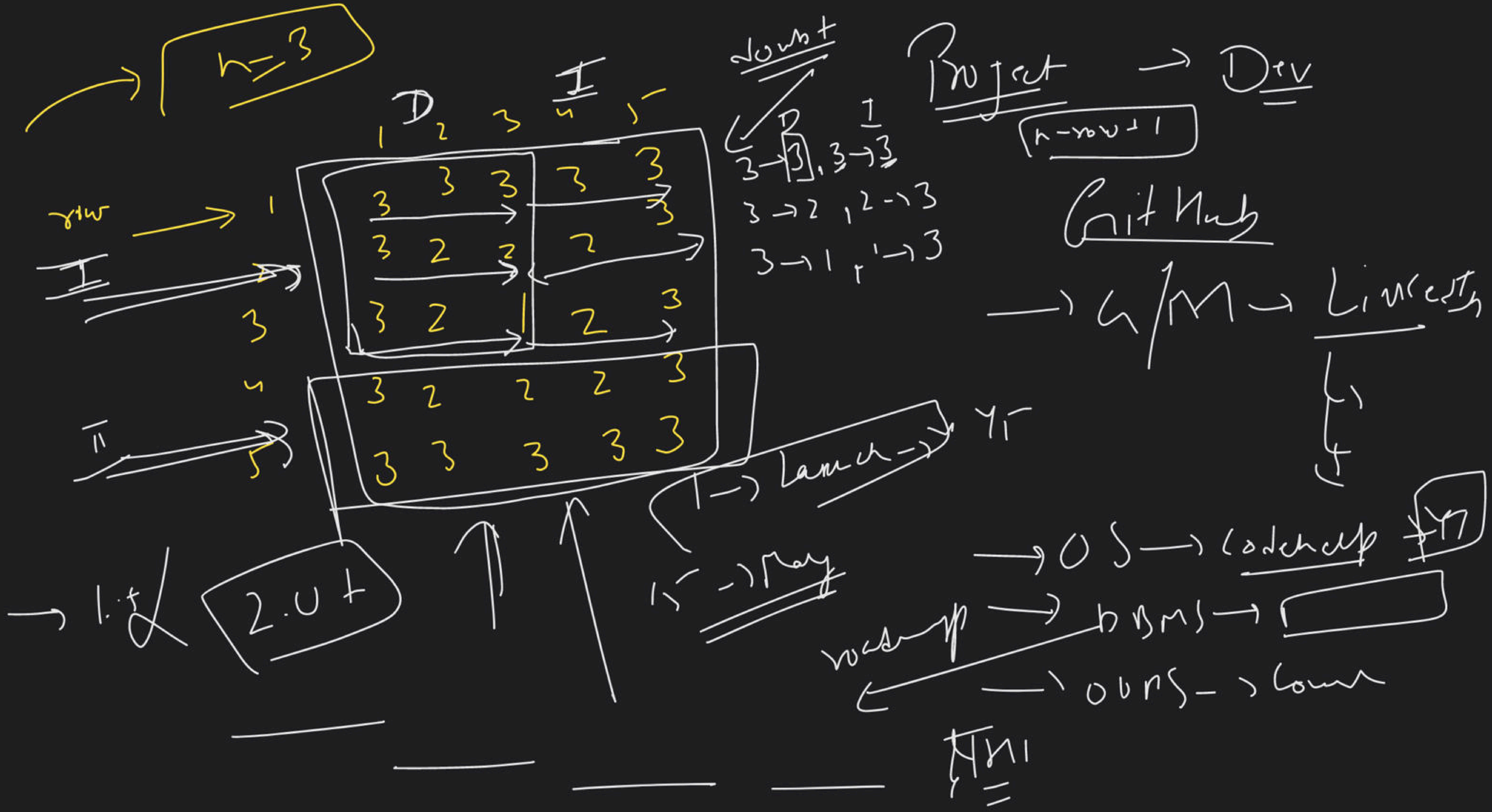

n=4
n
1
1 2 1
1 2 3 2 1
1 2 1
1 2 1
n-1
1
2
3
row=1
n=4
1
n-row-1
n-2-1
=0
1 2 3 4 3 2 1
1 2 3 2 1
1 2 1
1
1 -> (n-row-1)
inc
decr
h=4
row=1
n=4
2
n-row-1
for(row -> 1 -> (n-1))
{
if (row == (n-1))
cout << *
do{
{ cout << *
logic
cout << *
}


n = 7
1
2
3
4
1
1 2 1
1 2 3 2 1
1 2 1
1
12
for ( )
for ( )
{ }
possible
easy
for (row -> 1 -> n)
{
if (row = 0)
cout << B
else
{ cout <<
Logic
Count
3 3


int
main
{
call
xyz ( )
}
Compiler Design
phases
lexical
syntax
3-4hr
2-3


swap alternate
Kadane's Algo
14
Google
love Babbar
Hint
Discuss
Friend
try H/W
1-2 Q
15 Q
array → pattern
Basic


Array
print
1 Ques
15 Ques
Code
V.G
Optimise
3 sem
DSA
1 topic
Project
S.C
10 Mock test


n = 3
Project → Dev
GitHub
OS
DBMS
HTML