n = 3 n = Total No-

Row = 1 , <= 3

Row → 1

col = 1, Col <=

Col → 3

1
2
3
4 <= 3 → Stop =

Row 1 2 3

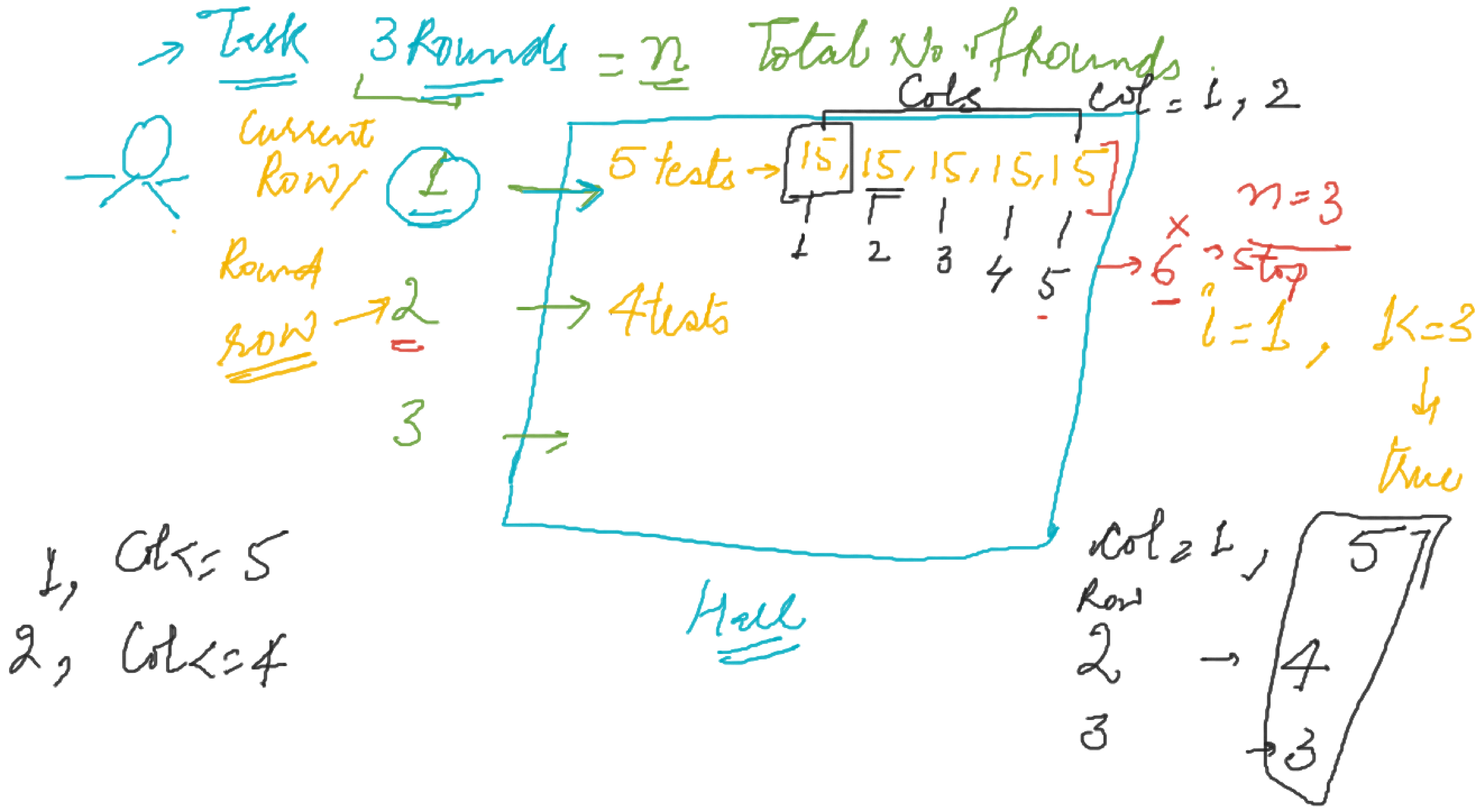
[1 1 1 1] 3

2 - -

3 -

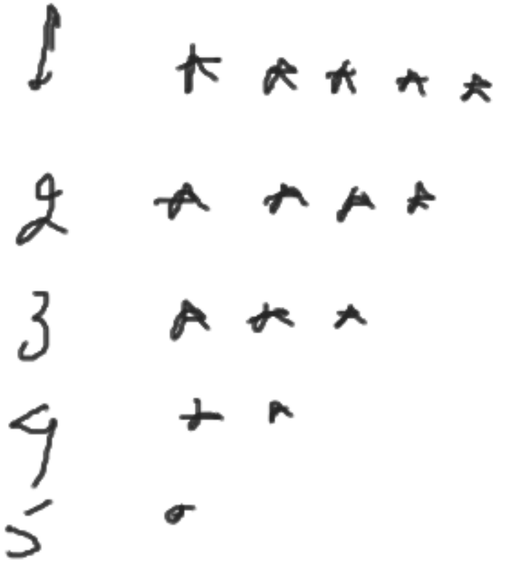
Row times Loop

1 3


→ Task
3 Rounds = n Total No. of Rounds
Cols
col = 1, 2
Current Row/ 1
5 tests → 15, 15, 15, 15, 15
1 2 3 4 5
n=3
→ 6 → stop
Round
Row → 2
4 tests
i=1, k=3
true
3
Hall
1, Col<=5
2, Col<=4
col=1,
Row
2 → 4
3 → 3
5

row	col (n+1) - row
1	5
2	4
3	3
4	2
5	1

$n = 5$



$5 - 1 \quad 4$

$n - 1 =$

$(n+1) - 1 = 5$

$(n+1) - 2 = 4$

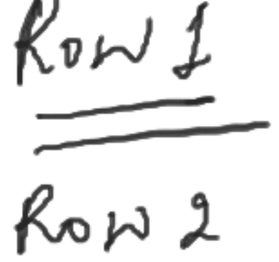
$(n+1) - 3 = 3$

col 2, col 5 = (n+1) - i

Turn

Row 1

Row 2



R-1 →

Round → Round

Turns

1

2 2

3 3 3

4 4 4 4

```
for (row1 ; row<=4 ; row++ ) {
    for(col=1; col<= row)
        Syso("row");
}
```

Row	Col
1	1
2	1 2
3	1 2 3
4	1 2 3 4

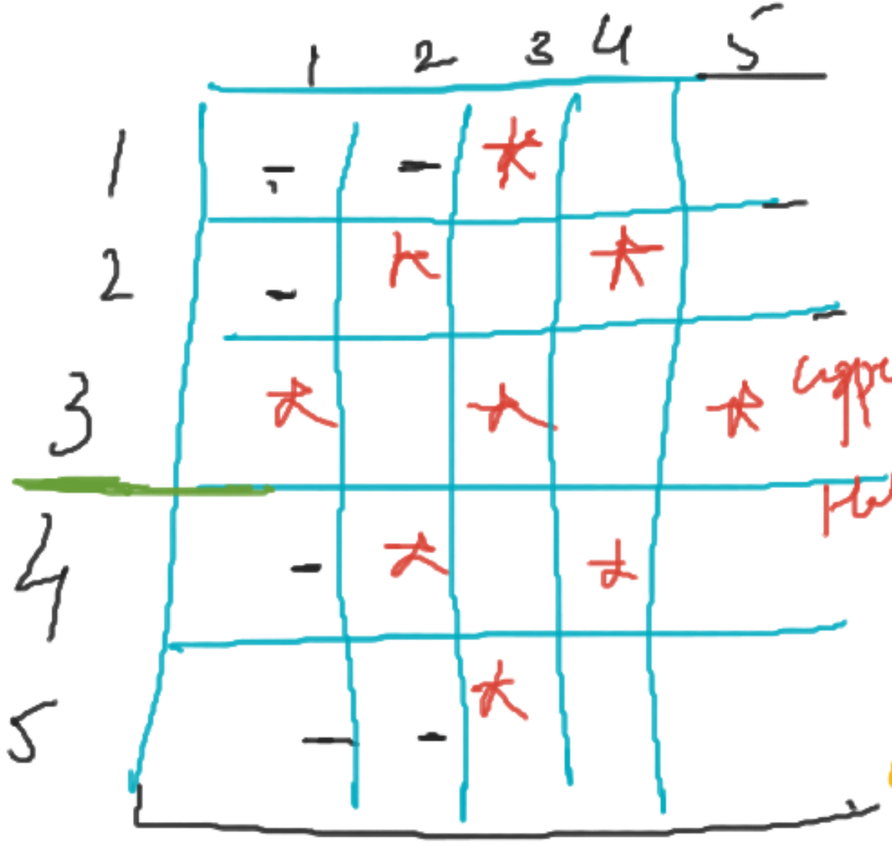
times

Row	Col
1	1
2	2

$n = 5$

$Half = (n+1)/2$

$= (5+1)/2 = 3$



Loop 1 → Row | Loop 2 → Spaces | Loop 3 → Stars

Row	Spaces	Stars
1	2	1
2	1	2
3	0	3
4	1	2
5	2	1

3 - row half

sp

if

Row - 1, 2, 3 <= half (3)

Row 4, 5 > half

else

SP → spaces

St → stars

```
int n = 5;  int sp, st = 0;

for( int row = 1; row <= n; row++) {
    if (row <= half) {  → upperhalf }  → if part row → 1, 2, 3
        sp = half - row;
    } else {
        sp++;
    }
    for( int space = 1; space <= sp; space++)
    {
        Syso ("-");
    }

    for( int star = 1; star < 2*row; star++)
    {
        Syso ("*");
    }
    Syso println();
}
```

Row 4, 5 ←

half upper
Lower half

upper
Lower Half

half = (3 - row)

① Row | Space

Row	Space
1	2
2	1
3	0

Row <= half → if part
upperHalf

half - row