1 = 1 to n

i ∈ j

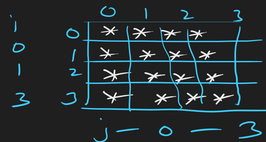
R1 ①
R2 ① ②
1 2 3
1 2 3 4
1 2 3 4 5

(n = 5)

for (i = 1, i <= n, i++)

for (j = 1, j <= i, j++)

print ("%d" j)



(n = 4)

i → rows
j → cols

(Hollow Square Pattern)

" "

4 conditions

(i == 1, i == n
j == 1, j == n)

All 4 cases

if (i == 1 || i == n
|| j == 1 ||
j == n)

Pyramid : n = 4

n	i	spaces	stars
4	1	3 n-i	1 2xi-1
4	2	2	3
4	3	1	5
4	4	0	7

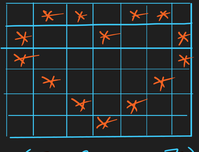
```
for (int i = 1; i <= n; i++) {
  // spaces
  for (int s = 1; s <= n - i; s++)
      printf (" ");
  // stars
  for (int star = 1; s <= 2*i - 1; star++) {
      printf ("*");
  }
  printf ("\n");
}
```

Diamond Pattern

[done]

Homework :

(Static Pattern)

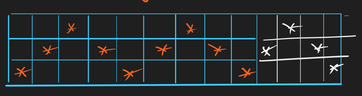


(r = 6, c = 7)

Heart Pattern

(Zig Zag Pattern)

Dynamic Pattern



row → constant = ③

c → val → 9, 13, 17, 21, and so on

* Functions :→ A block of code that does a particular task, when called upon.
It is only invoked / executed when we call it. Else, it stays dead.

* Argument → Actual values that we provide in function call : function (a);

* Parameter → The variables in the function definition -

```
int addition (int a, int b)
{
}
```

parameters

$$\left[SI = \frac{PTR}{100} \right]$$

void simpleInterest (p, t, r)