Patter Printing Part-4

Aavance Pattern

①
```
        *
      * * *
    * * * * *
  * * * * *  * *
```

space → *

row	space	star
1	4	1
2	3	3
3	2	5
4	1	7
5	0	9

5-row
↕
n-row

(i) row = 1
(ii) row <= 5 → n
(iii) print space 5-row times n
(iv) print "*" 2×row - 1 , times
(v) row = row+1

$(2 \times row) - 1$

$2 \times 1 - 1 = 1$
$2 \times 2 - 1 = 3$

```
for(row=1; ro<=n; row=row+1)
{
    for( col=1; col <= n-row; col=col+1)
    {
        cout<<" ";
    }
    for(col=1; col <= 2×row-1; col=col+1)
    {
        cout <<"*"<<" ";
    }
}
```

(ii)

```
            1
          1 2 1                              n = 5
        1 2 3 2 1                    space → number.
      1 2 3 4 3 2 1            (5-row) | 1 to row,
    1 2 3 4 5 4 3 2 1                  |        incr.
                                       | (row-1) to 1
                                       |        decr.
```

(i)   row = 1
(ii) row <= 5
(iii) print space 5-row times.
(iv) print 1 to row increasing order
(v)   print row-1 to 1 decreasing order.
(vi) row = row + 1.

```
for ( row=1 ; row<=n ; row=row+1)
    for( col=1 ; col<= n-row ; col=col+1)
    {
        cout<<" ";
    }
    for( col=1 ; col <= row ; col=col+1)
        cout << col <<" ";
    for( col=row-1 ; col >= 1 ; col=col-1)
        cout << col <<" ";
    cout << endl;
}
```

(iii)

```
*  *  *  *  *  *  *  *  *
   *  *  *  *  *  *  *
      *  *  *  *  *
         *  *  *
            *
```

n = 5

space → *

row		space		*
5	1	0		9
4	2	1		7
3	3	2		5
2	4	3		3
1	5	4		1

① row → $2(n-row)+1$

$2(5-1)+1$
$8+1 = 9$

$2(5-2)+1$
$= 7$

$2(n-row)+1$

② 5 star

$n \times 2 - 2row =$

Alternate

$5 \times 2 - 2 \times 1 - 1$

$\dfrac{5 \times 2 - 2 + 1}{\textcircled{9}}$

→ (row-1) → ① space

$(2\,row - 1)$ → ② star

$$\dfrac{9 - (2\,row - 1)}{2}$$ → ① space

Method - I

(i) row = 5 ← n

(ii) row >= 1

(iii) print space
$$\dfrac{9 - (2\,row - 1)}{2}\ \text{times}$$

(iv) print * $2\,row - 1$

(v) row = row - 1

$(2 \times n - 1) - 2(row - 1)$
$\Rightarrow \dfrac{2n - 2\,row}{2} = (n - row)$

Method - II

(i) row = 1

(ii) row <= n

(iii) print space
row - 1 times.

(iv) print *
$(2 \times (n - row) + 1)$
times.

(v) row = row + 1

↓

Debug n = 5

```
*  *  *  *  *  *  *  *  *
   *  *  *  *  *  *  *
      *  *  *  *  *
```

$(2 \times (5-1) + 1) ← row = 1 (5 \checkmark)$
$← row = 2 (<5 \checkmark)$
$(2 \times (5-2) + 1) ← row = 3 (<5 \checkmark)$
$\textcircled{7}\ (2(5-3) + 1)$

Method -1

```
for( row = n; row >= 1; row = row +1)
{
    for(col = 1; col <= n-row; col++)
    {
        cout << " ";
    }
    for(col = 1; col <= 2*row-1; col++)
    {
        cout << "* ";
    }
    cout << endl;
}
```

Method → 2

```
for(row = 1; row <= n; row++)
{
    for(col = 1; col <= row-1; col++)
        cout << " ";
    for(col = 1; col <= (2*(n-row)+1); col++)
        cout << "* ";

    cout << endl;
}
```

(iv)   n = 4

```
*   *   *   *   *   *   *
*   *   *   -   -   *   *   *
*   *   -   -   -   -   *   *
*   -   -   -   -   -   *        * → space → *
*   -   -   -   -   -   *
*   *   -   -   -   -   *   *
*   *   *   -   -   *   *   *
*   *   *   *   *   *   *   *
```

part - ①

part - ②

part - 1

space	row	*	← (2× row)
0	4 ← →	4 , 4	
2	3 ← →	3 3	
4	2 ← →	2 2	
6	1 ← →	1 1	

(i) 2×n - 2×row
    2(n - row)

(O)

part - I

(i)   row = 4, n
(ii) row >= 1
(iii) Print * row time
(iv) Print space 2(n - row) time
(v)   print * row time
(vi) row = row - 1

part - 2

```
*                        *
*  *                  *  *
*  *  *            *  *  *
*  *  *  *      *  *  *  *
```

print → space → print

part - II

space	row	star	
6	1	1 1 → 2×row	
4	2	2 2	
2	3	3 3	
0	4	4 4	

8 - 2 row
2×n - 2 row
2(n - row)

(i)   row = 1

(ii)   row <= n

(iii) Print * row times

(iv)   Print space 2(n-row) times.

(v)   Print * row times.

(vi)   row = row+1.

```cpp
for( row = n; row >= 1; row--)
{ for( col = 1; col <= row; col++)
      cout << "*";
  for( col = 1; col <= 2(n-row); col++)
      cout << " ";
  for( col = 1; col <= row; col++)
      cout << "*";

  cout << endl;
}
for( row = 1; row <= n; row++)
{
    for( col = 1; col <= row; col++)
        cout << "*";
    for( col = 1; col <= 2(n-row); col++)
        cout << " ";
    for( col = 1; col <= row; col++)
        cout << "*";
    cout << endl;
}
```

(v)   *   *   (n = 4)

part → 1

```
   *              *  *
   *  *           *  *  *
   *  *  *        *  *  *  *  *  *  *
```

part - 2

```
   *  *  *        *  *  *
   *  *           *  *
   *              *
```

\# → space → *

(part-1)

row	space	*	
1	6	1, 1	2 ← row
2	4	2 2	
3	2	3 3	
4	0	4 4	
	↓		
	2(n-row)		

(i)   row = 1
(ii)   row <= n
(iii)   print * row times
(iv)   print space 2(n-row) times
(v)   print * row times
(vi)   row = row+1

(part - 2)

row	space	*	
3	2	3 , 3	2row
2	4	2, 2	
1	6	1 1	

① row = n-1
② row >= 1
③ row print * row times
④ print space (2(n-row)) times
⑤ Print * row times
⑥ row = row -1

```
for( row=1; row<=n; row++)
{
    for(col=1; col<=row; col++)
        cout<< "*" ;
    for( col=1; col <= 2x(n-row); col++)
        cout<< " ";
    for(col=1; col <= row; col++)
        cout<< "*" ;
    cout<< endl;
}
for( row=n-1; row>=1; row--)
{
    for( col=1; col <= row; col++)
        cout<< "*";
    for( col=1; col <= 2(n-row); col++)
        cout<< " ";
    for( col=1; col <= row; col++)
        cout<< "*";
    cout<< endl;
}
```

(vi) Part-1 n=4

Part-2

space → *

Part-1.
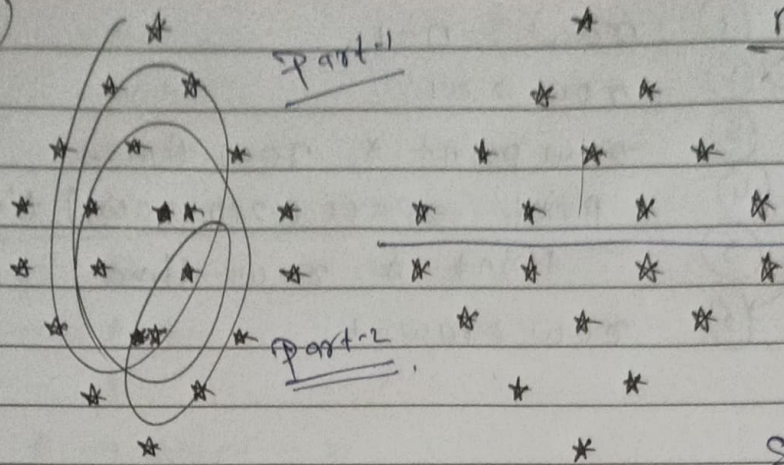
row	Star (*-)	space (-)
1	1	4 - row = 3
2	2	2
3	3	1
4	4	0 (n - row)

(i) row = 1
(ii) row <= n
(iii) Print space n-row time
(iv) Print * row times
(v) row = row+1.

Part-2.

row	Star (*-)	space (-)
4	4	0
3	3	1
2	2	2
1	1	3
		(n-row)

ⓘ row = n

→ⓘⓘ row >= 1

ⓘⓘⓘ print space   n-row times.

ⓘⱽ print *   row times

ⱽ row = row-1.

```
for( row=1; row<=n; row++)
{
    for( col=1; col<=n-row; col=col+1)
        cout <<" ";
    for( col=1; col<= row; col=col+1)
        cout<<"* ";
    cout << endl;
}
for(row=n; row>=1; row--)
{for(col=1; col<=n-row; col=col+1)
        cout <<" ";
    for(col=1; col<= row; col=col+1)
        cout << "* ";
    cout<< endl;
}
```