

DAY #21

Wednesday, June 17, 2025 8:30 PM

NUMER PATTERN

```
int rows = 5;
int no = 1;
for (int i = 0; i < 5; i++) // 0 1 2 3 4
{
    for (int j = 0; j <= i; j++)
    {
        printf("%d", no);
        no++;
    }
    printf("\n");
}
```

Row 0 - 1
Row 1 - 2 3
Row 2 - 4 5 6
Row 3 - 7 8 9 10
Row 4 - 11 12 13 14 15

no = 1 2 3 4 5 6 7

i=0	0 < 5	True	j=0	0 <= 0
i=1	1 < 5	True	j=1	1 <= 1
i=2	2 < 5	True	j=2	2 <= 2
i=3	3 < 5	True	j=3	3 <= 3

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

```
int rows;
char ch = 'A'; // 65
for (int i = 0; i < 5; i++)
{
    for (int j = 0; j <= i; j++)
    {
        printf("%c", ch);
        ch++;
    }
}
```

A B
A B C

A
B C
D E F
G H I J
K L M N O

65 → A
66 → B
67 → C
68 → D

Continue statement in C

- ↳ It's a keyword (have reserve meaning)
- ↳ can't use 'continue' as variable also
- ↳ It is used in loops (for, while, do-while)
- ↳ It continues the loop ignoring code written after it.

Syntax:

```
for (int i = 0; i < 5; i++) // 0 1 2 3 4
{
    if (i % 2 == 0)
        continue;
    printf("Hello");
}
```

i=0	0 < 5	TRUE	0 % 2	0 == 0	TRUE
i=1	1 < 5	TRUE	1 % 2	1 == 0	FALSE
i=2	2 < 5	TRUE	2 % 2	2 == 0	TRUE
i=3	3 < 5	TRUE	3 % 2	3 == 0	FALSE
i=4	4 < 5	TRUE	4 % 2	4 == 0	TRUE
i=5	5 < 5	FALSE			

Hello
*
Hello }

Pyramid Pattern in C
 int rows = 5;

for (int i = 1; i <= rows; i++)

{
 // print spaces
 for (int j = 1; j <= rows - i; j++)
 {
 printf(" ");
 }
 for (int k = 1; k <= (2 * i - 1); k++)
 {
 printf("* ");
 }
 printf("\n")
}



```

1
 1 1
 1 1 1
 1 1 1 1
 1 1 1 1 1
  
```

4 spaces 1 * Row 1
 3 spaces 3 * Row 2
 2 spaces 5 * Row 3
 1 space 7 * Row 4
 0 space 9 * Row 5

i = 1, 1 <= 5 TRUE j = 1 1 <= 4 T → - - - - *
 j = 2 2 <= 4 T - - - * * *
 j = 3 3 <= 4 T - - * * * *
 j = 4 4 <= 4 T - * * * * *
 j = 5 5 <= 4 F

k = 1 1 <= (2 * 1 - 1) 1 <= 1 T
 k = 2 2 <= (2 * 1 - 1) 2 <= 1 F

i = 2, 2 <= 5 TRUE j = 1 1 <= 3 T
 j = 2 2 <= 3 T
 j = 3 3 <= 3 T
 j = 4 4 <= 3 F

k = 2 1 <= (2 * 2 - 1) ⇒ 1 <= 3 T

}