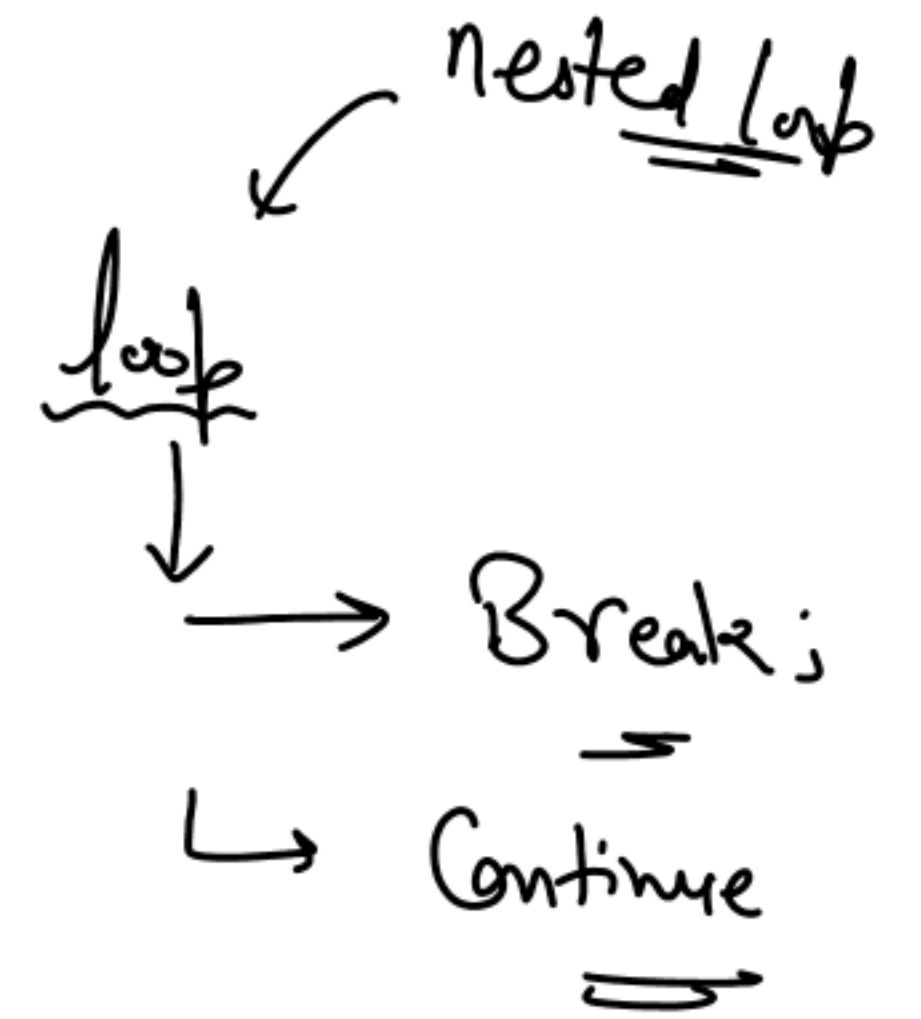


Lecture - 17

Programming in C

Flow of Control (Part-05)



Break Statement :

↳ Used to terminate the loop.

Conditions

Ex `for (int i=0; i<10; i++) {`
 `↳ printf ("%d", i);`
 `⇒ if (i==3) {`
 `break;`
 `}`

terminate the loop

$i=0, i<10$

① $i=0, i<10$ (T)
 $↳ i==3$ (F)

$i++ \rightarrow 1$

② $i=1, i<10$ (T)
 $↳ i==3$ (F)

$i++ \rightarrow 2$

③ $i=2, i<10$ (True)
 $↳ i==3$ (False)
 $i++ \rightarrow 3$

④ $i=3, i<10$ (True)
 $↳ i==3$ (True)

$↳ \text{break}$ ← Terminate

o/p
0

1

2

3

Ex

```
int i = 0;
for ( ; i < 10; i++ ) {
    if ( i > 4 ) {
        break;
    }
    else {
        printf ("%d", i);
    }
}
```

Initialise (points to `i = 0`)
Valid (points to `i < 10`)

$i = 0, i < 10$ O/P

① $i = 0, i < 10$ (True)
 $\hookrightarrow i > 4$ (False)
 $i++ \Rightarrow 1$ 0

② $i = 1, i < 10$ (True)
 \hookrightarrow if ($i > 4$) \rightarrow false
 \hookrightarrow else \Rightarrow 1
 $i++ \Rightarrow 2$

③ $i = 2, i < 10$ (True)
 \hookrightarrow if : $i > 4$ (False)
 \hookrightarrow Else
 $i++ \rightarrow 3$ 2

④ $i = 3, i < 10$, True
 \hookrightarrow if : $i > 4$ (False)
 \hookrightarrow Else,
 $i++ \rightarrow i = 4$ 3

⑤ $i = 4, i < 10$ (True) \rightarrow if ($i > 4$) \leftarrow false
 \hookrightarrow Break 4 \neq 4

Continue Statement :

↳ if skip the current iteration

Ex

```
for (int i = 0, i < 10; i++) {  
    if (i % 2 == 0) {  
        continue;  
    }  
    printf("o/d", i);  
}
```

Skip all the next codes inside loop

i = 0 o/p
① $i = 0, i < 10$
↳ $i \% 2 == 0 \rightarrow \text{Continue}$
 $i++ \rightarrow 1$

② $i = 1, i < 10$ (True)
↳ $i \% 2 == 0$ (False)
→ Print
 $i++ \rightarrow 2$ 1

③ $i = 2, i < 10$ (T)
↳ $2 \% 2 == 0$ (True) → Continue (Skip)
 $i++ \rightarrow 3$

④ $i = 3, i < 10$ (T)
↳ $3 \% 2 == 0$ (False)
↳ $i++ \Rightarrow 4$ 3

⑤ $i = 4, 4 < 10$ (True)
↳ $4 \% 2 == 0$ (True) → Continue (Skip)
 $i++ \rightarrow 5$

⑥ 5
⋮

o/p
1
3
5
7
9
↖ 10 < 10
 $i < 10$

Nested Loop:

↳ Loop inside the loop

```
for ( — ) {  
    for ( — ) {  
        for ( — ) {  
            ⋮  
            n for ( — ) {  
            }  
        }  
    }  
}
```

← nested
for = loop

← Nested while loop

```
while ( condition ) {  
    while ( condition ) {  
        while ( condition ) {  
            ⋮  
        }  
    }  
}
```

Write a program to print the following pattern :

```
1 *
2 * *
3 * * *
4 * * * *
  ⋮
n n (*)
```

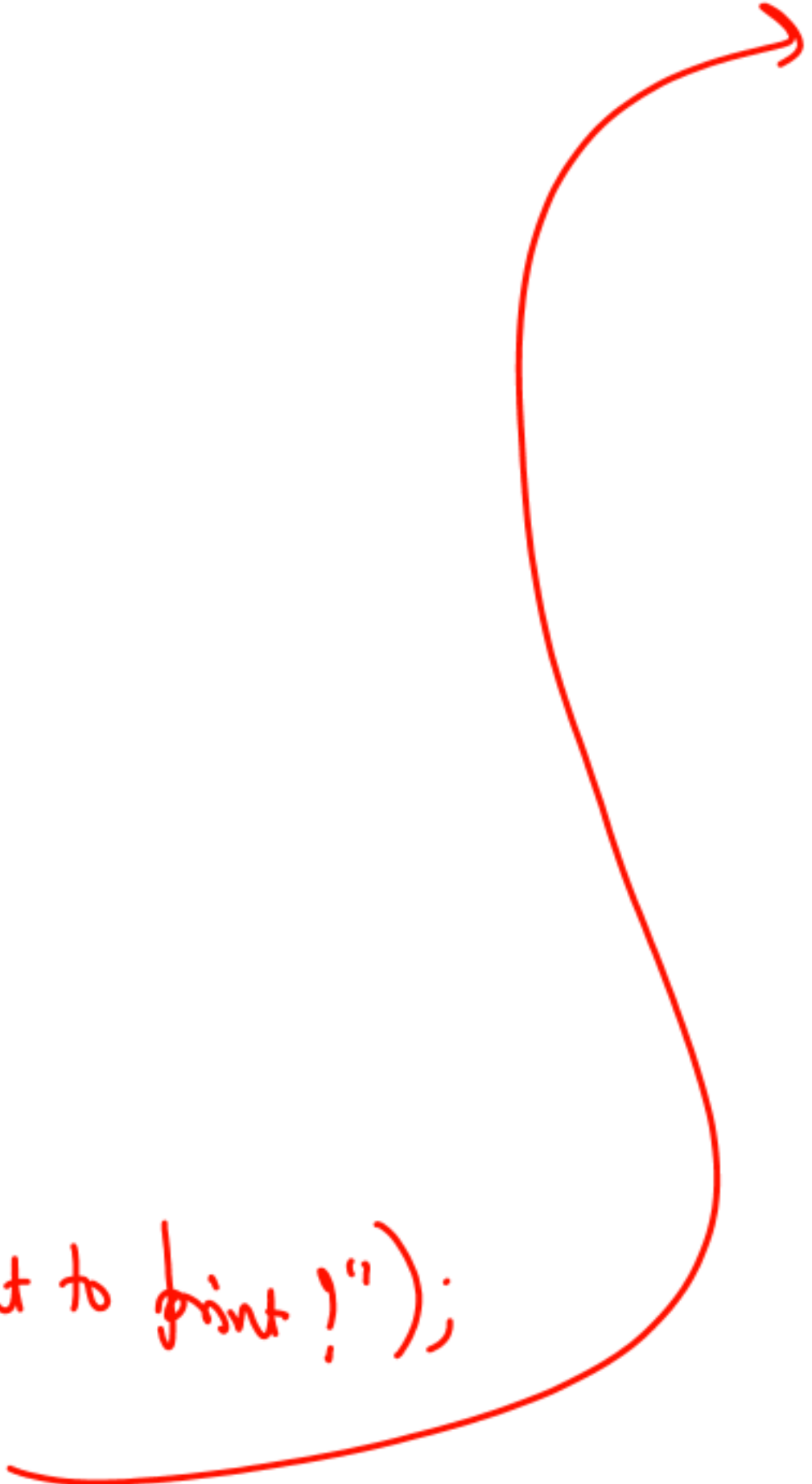
```
#include <stdio.h>
```

```
int main () {
```

```
    int n;
```

```
    printf("How many rows you want to print?");
```

```
    scanf("%d", &n);
```



```
for (int i=1; i<=n; i++) {
    for (int j=0; j<i; j++) {
        printf("*");
    }
    printf("\n");
}
```

```

1  #include<stdio.h>
2
3  int main(){
4      int n;
5      printf("Enter the number of rows you want to print? \n");
6      scanf("%d", &n); // 5 -> i = 1,2,3,4,5
7      for (int i = 1; i<=n; i++){
8          for (int j = 0; j<i; j++){
9              printf("*");
10         }
11         printf("\n"); ←
12     }
13     return 0;
14 }

```

Enter the number of rows you want to print?

```

5
*
**
***
****
*****

```

$n = 5$

① $i = 1, j \leq 5$ (True) Output

① $j = 0, j < i$ (True), $j++$

$j = 1$ → *

② $j < i \Rightarrow 1 < 1$ (False) —

$i++ \rightarrow 2$

② $i = 2, 2 \leq 5$ (True)

↳ for ① $j = 0, j < i$ (True)

$j++ \rightarrow j = 1$ → * *

② $j = 1, j < i$ (True) → *

$j++ \rightarrow j = 2$

③ $j = 2, j < i$ ($2 < 2$) False

③ $i = 3, 3 \leq 5$ (True)

for ① $j = 0, j < i$ (True) $j++$ → * * *

for $j = 1, j < i$ (True) $j++$ →

for $j = 2, j < i$ (True) →

for (int i=1; i<=2; i++){

for (int j=0; j<i; j++){

for (int k=0; k<j+1; k++){

printf("%d", k);

};

printf("%d", j);

printf("%d", i);

}

① Dry Run

① A) i=1, i<=2 (True)

↳ B) j=0; j<i (True)

↳ C) k=0, k<j+1 (True)

↳ Print → k, k++
k=1 k<j+1 = false

→ j=1, 1<i (False)

i++ ⇒ i=2

② A) i=2, i<=2 (True)

↳ B) j=0, j<i (True)

↳ C) k=0, k<j+1 (True)
k++, k=1, k<j+1

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

↳ C) k=0, k<j+1 (True) k++
k=1 k<j+1 (True) k++
k=2 k<j+1 (True)

→ j++ → ②, j<i (2<2) false

i++ → i=3

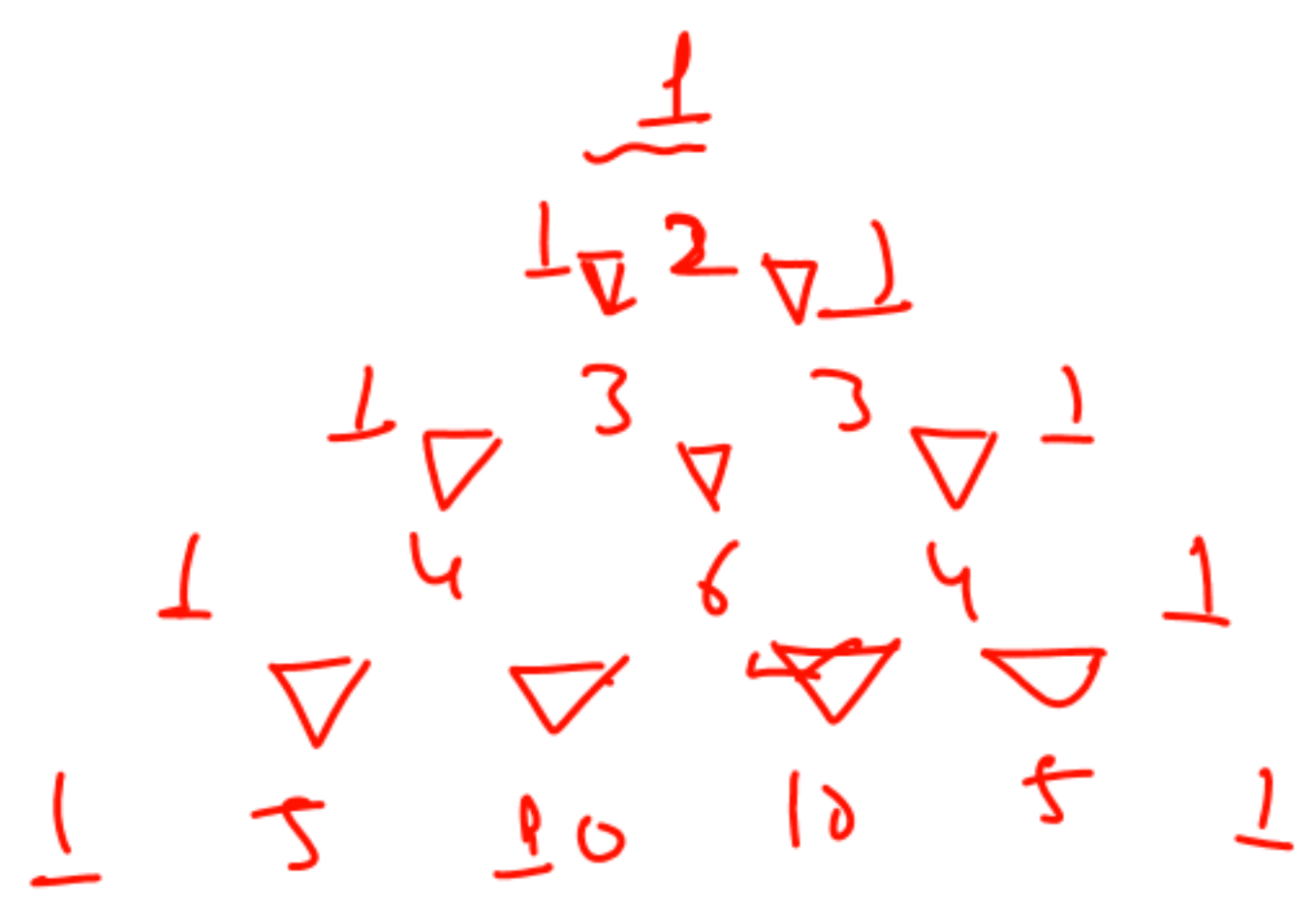
③ A) i=3, i<=2 (False)

Output

0	0	1
0	0	0
1	1	2


```
1 #include<stdio.h>
2 int main(){
3     for (int i = 1; i<=2; i++){
4         for (int j = 0; j<i; j++){
5             for (int k = 0; k<j+1; k++){
6                 printf("%d ", k);
7             }
8             printf("%d ", j);
9         }
10        printf("%d ", i);
11    }
12    return 0;
13 }
```

0 0 1 0 0 0 1 1 2
PS C:\Users\sagar\OneDri
hand\Programming in C\Co



5 4 3 2 1
4 3 2 1
3 2 1
2 1
1

1
1 2
1 2 3
1 2 3 4
1 2 3 4 5