24. CONTINUE- LOOP CONTROL STATEMENT

EXAMPLE 1

```
for(int i = 1; i ≤ n; i++){
        if(i = 5){
            continue;
            }
      print(i);
}
```

Solution:

if i $= 5$, ***the loop will skip that part and loop will run*** $(continue)$***upto*** n.

That is $1, 2, 3, 4, 6, 7, \ldots\ldots n$ ***and i will be printed***

$(1 + 1 + 1 + \cdots n - 1$ ***times***$) = n - 1$ ***and***

the time complexity we get is $O(n)$.

EXAMPLE 2

```
for(int i = 1; i ≤ n; i++){
    for(int j = 1; j ≤ n; j++){
        if(j = 5){
            continue;
            }
        print("Hello");
 }
}
```

Solution:

j will run n − 1 times n i.e $n^2 - n$ that the innermost statement Hello ***will be executed.***

Time complexity $= O(n^2 - n) = O(n^2)$.

EXAMPLE 3

```
for(int i = 1; i ≤ n; i++){
    for(int j = 1; j ≤ n; j++){
        if (j = i/2) {
            continue;
        }
      print("Hello");
 }
}
```

Solution:

$when\ i = 1\ , j = \frac{1}{2} = 0.5\ or\ 0\ doesn't\ exist\ , hence\ j\ runs\ n\ times.$

$when\ i = 2\ , j = \frac{2}{2}$

$= j\ skips\ 1\ and\ runs\ upto\ n\ times\ hence\ n - 1\ times.$

$when\ i = 3\ , j = \frac{3}{2}$

$= j\ skips\ 1\ and\ runs\ upto\ n\ times\ hence\ n - 1\ times.$

……

$when\ i = n\ , j = \frac{n}{2} = j\ skips \frac{n}{2} and\ runs\ upto\ n - 1\ times.$

$\boldsymbol{Therefore\ ,j\ will\ run\ n-1\ times\ n\ i.e\ n^2-n\ that\ the\ innermost\ statement}$ Hello $\boldsymbol{will\ be\ executed}$.

$\boldsymbol{Time\ complexity = O(n^2 - n) = O(n^2)}$.

.....