unacademy

Control flow statements - Part III

Comprehensive Course on C- Programming

Pankaj Sharma • Lesson 10 • May 13, 2024

unacademy

CS & IT Engineering

C Programming
Control Flow statements-III

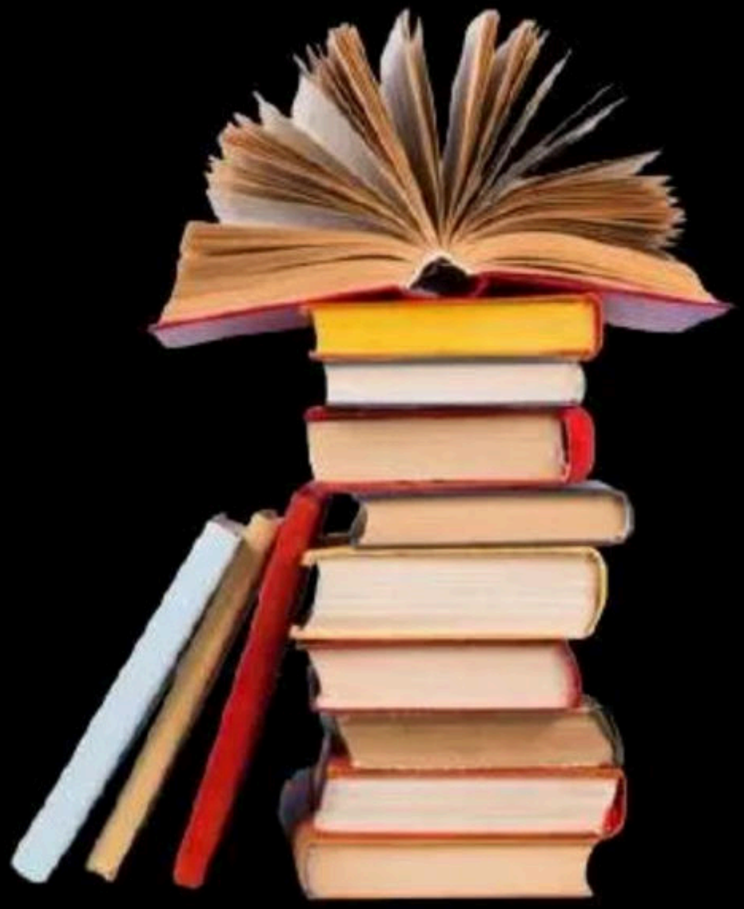
Lecture - 10

By- Pankaj Sir

unacademy

Topics

to be covered



1. Iterative statements part -II

1.
```
for(i=1; i<=10; i++)
    printf("Pankaj");
```
→ 10 ✓

2.
```
for(i=1; i<=n; i++)
    printf("Pankaj");
```
n>=0

→ n ✓

3. `for(i=1; i<=10; i=i+2)`
   `printf("Pankaj");`

$i = 1, 3, 5, 7, 9, 11$ ✗

5 times

4. `for(i=1; i<=n; i=i+2)`
   `printf("Pankaj");`

odd

$n \geq 0$

$n \rightarrow$ even (10)

$\frac{10}{2} = 5$ ✓

$n \rightarrow$ odd (11)

$\frac{11}{2} = 5$

$\left\lceil \frac{n}{2} \right\rceil$ ✓

unacademy

5.
```
for( i=1; i<=n; i = i*2)
    printf("Pankaj");
```

n = 100

i = 1, 2, 4, 8, 16, 32, 64, X 28

i ki last value

n
100 → 7 times

n = 64

i = 1, 2, 4, 8, 16, 32, 64

i ki last value

i = 2^0, 2^1, 2^2, 2^3, ... 2^k

How many counting

last value of i

ft → (k+1) times

⇒ What is k ?

unacademy

$$2^k <= n$$

$$\log(2^k) <= \log n$$

$$k \log 2 <= \log n$$

$$k <= \frac{\log n}{\log 2}$$

$$\Rightarrow k <= \log_2 n$$

$$k = \lfloor \log_2 n \rfloor$$

unacademy

$1 < < = \log_2 n$

$\Downarrow$

3

$n = 8$

$\log_2 10$

$n = 10$

$\Rightarrow \lfloor 3.x \rfloor$

$= 3$

1.
```
for(i=1; i<=3; i++)
  {
   Code
  }
```
How many times code will execute:

$i = 1,2,3 \Rightarrow$ 3 times

$i=1$ → code ✓

$i=2$ → code ✓

2.
```
for(i=1; i<=10; i++)
  {
   Code
  }
```
$i = 1,2,3,\ldots 10 \Rightarrow$ 10 times

3.)
```
for(i=1; i<=n; i++)
  {
   Code
  }
```
n times

unacademy

1.

```
for(j=1;j<=4;j++)
  {
  printf("Pankaj");
  }
```

↳ Pankaj Pankaj Pankaj Pankaj

5.

```
for(i=1;i<=3;i++)
  {
  Code
  }
```

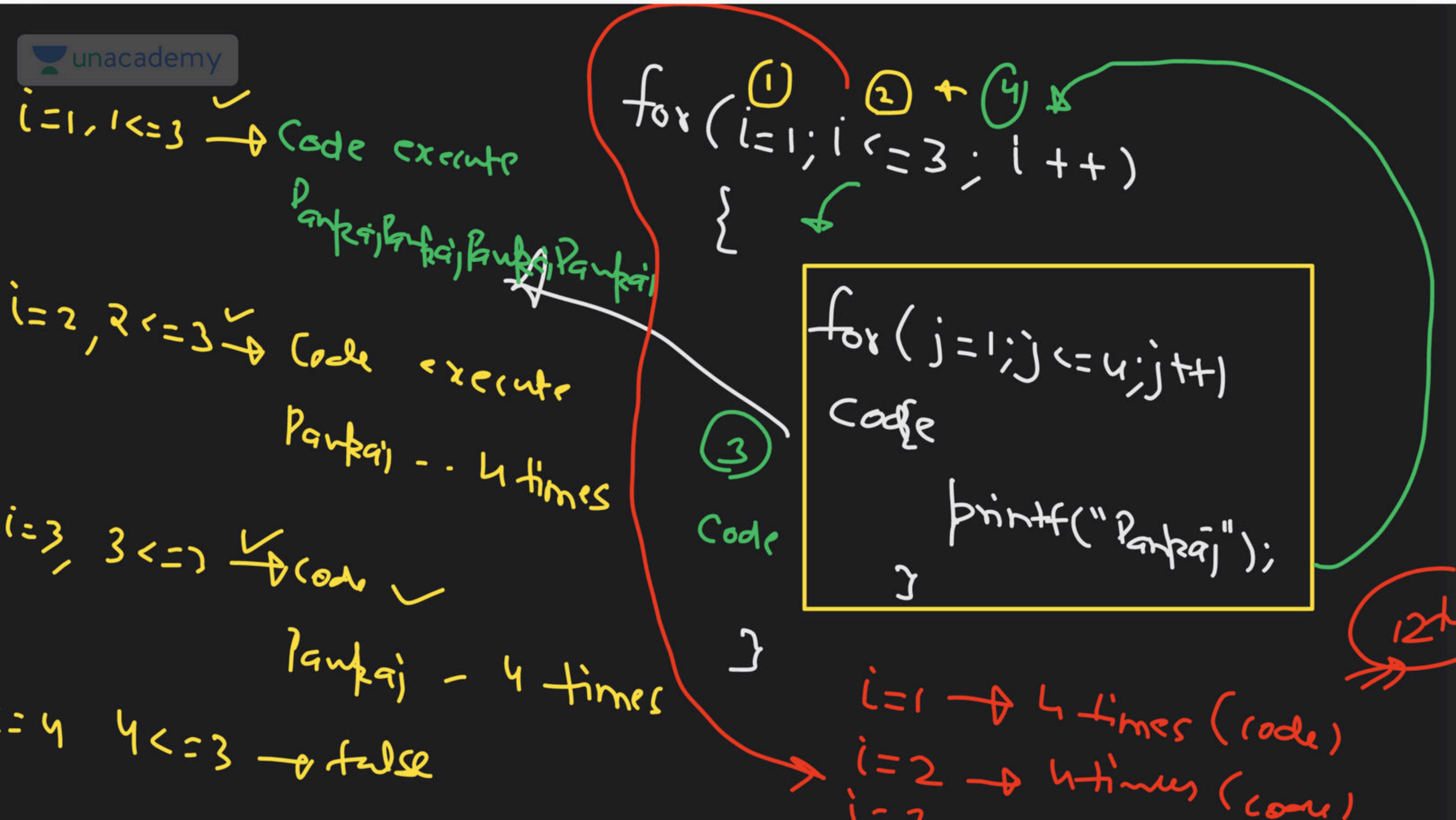
① ② ④ ③

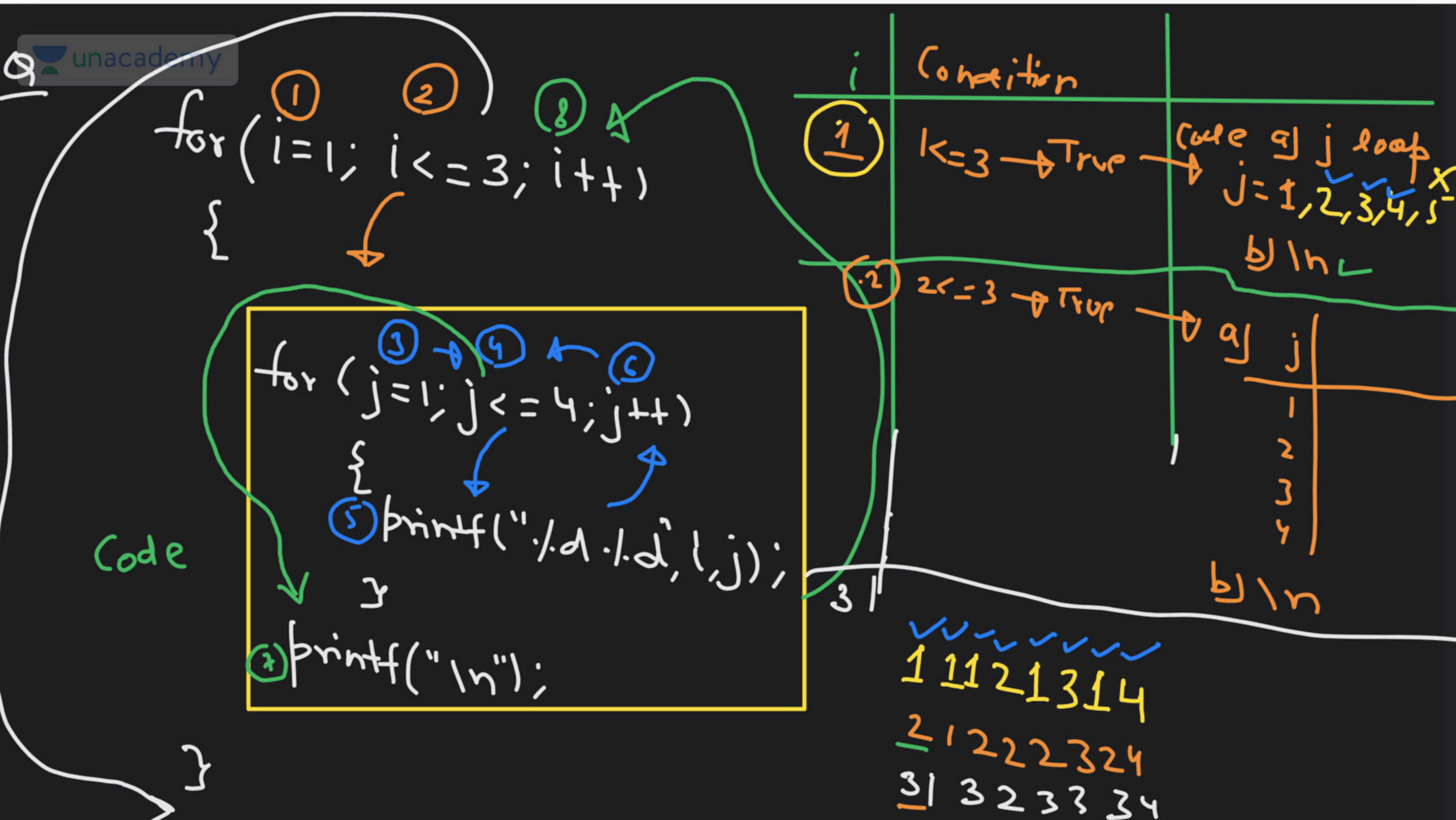
What can be Code?

i=1, 1<=3 ✓ → Code ✓

i=2, 2<=3 ✓ → Code

i=3 3<=3 ✓ → Code


unacademy
i=1, 1<=3 → Code execute
Pankaj Pankaj Pankaj Pankaj
i=2, 2<=3 → Code execute
Pankaj - - 4 times
i=3, 3<=3 → Code
Pankaj - 4 times
=4 4<=3 → false
for (i=1; i<=3; i++)
{
Code
}
for (j=1; j<=4; j++)
Code
printf("Pankaj");
}
1
2
4
3
i=1 → 4 times (code)
i=2 → 4 times (code)
12th


unacademy
for(i=1; i<=3; i++)
{
for(j=1; j<=4; j++)
{
printf("%d %d", i, j);
}
printf("\n");
}
Code
i
Condition
1
1<=3 → True
Code of j loop
j = 1, 2, 3, 4, 5
o/p \n
2
2<=3 → True
o/p j
1
2
3
4
o/p \n
3
1 1 1 2 1 3 1 4
2 1 2 2 2 3 2 4
3 1 3 2 3 3 3 4

Q

```
for(i=1; i<=3; i++)
{
    for(j=1; j<=4; j++)
    {
        printf("Pankaj");
    }
}
```

Code

i	
1	j = 1,2,3,4 → 4 times
2	j = 1,2,3,4 → 4 times
3	j = 1,2,3,4 → 4 times

4 → 4 <= 3 → ×

i = 1 → (4)

i = 2 → (4)

i = 3 → (4)

4 + 4 + 4 (3 times)

= 3 × 4

for every value of i

$\Rightarrow$ inner loop $\Rightarrow$ 4 times ff.

3×4

$= 12$ times

unacademy

Q1

```
for(i=1; i<=3; i++)
{
    for (j=1; j<=10; j++)
    {
        printf("Pankaj");
    }
}
```

3 times

10 times

i=1 → 10

i=2 → 16

i=3 → 16

10 + 10 + 10

= 3 × 10 = 30 times

Q

```
for(i=1 ; i<=3; i++)
{
    for( j=1 ; j<=n; j++)
    {
        printf("Pankaj");
    }
}
```

3

n

$i=1 \rightarrow n$

$i=2 \rightarrow n$

$i=3 \rightarrow n$

$n+n+n$

$= 3 \times n$

$= 3n$ times

unacademy

Q)

```
for(i=1; i<=n; i++)
{
    for(j=1; j<=3; j++)
    {
        printf("Pankaj");
    }
}
```

i = 1 → 3
i = 2 → 3
:
i = n → 3

3 + 3 + 3 + 1 ... n times

= 3n

unacademy

Q

```
for(i=1; i<=n; i++)          → n
  {
    for(j=1; j<=n; j++)      → n
      {
        printf("Pankaj");
      }
  }
```

$n \times n$

$= n^2$ times

Q

```
for(i=1;i<=n;i++)
{
    for(j=1;j<=n;j=j*2)
    {
        printf("Pankaj");
    }
}
```

n times

$(1+\lfloor \log_2 n \rfloor)$

$n(1+\lfloor \log_2 n \rfloor)$

Sahib

Q

```
for(j=1; j<=10; j=j/2)
{
    printf("Pankaj");
}
```

0 <= 10

3 → —

j=1 1<=10 → ✓

j=1/2

=0

j=-0/2

=0

ϕ — f_0

log → =

$i = 1, \frac{1}{2}$ ↓ 0

ggar → { bhalbasi }

unacademy

```
for( j=128; j>=1; j=j/2)
{
    pf("Pankaj");
}
```

j = 128
j = 64
j = 32
16
8
4
2
1

$\log_2^{128}$

$= 7 + 1$

$\Rightarrow 8$ times

unacademy

```
for(i=1; i<=128; i=i*2)
{

}
```

i = 1, 2, 4, 8, 16, 32, 64, 128

```
for(j=128; j>=1; j=j/2)
{

}
```

j = 128, 64, 32, 16, 8, 4, 2, 1

Q

```
for (i=1; i<=3; i++)
{
    for (j=1; j<=i; j++)
    {
        printf("Pankaj");
    }
}
```

i	j
1	1 → (1)
2	1,2 → (2)
3	1,2,3 → (3)

1 + 2 + 3 ⇒ 6 times

⇓

dependent loop

loop unfolding

1. $1+2+3+\cdots+K = K(K+1)/2$

2. $1^2+2^2+3^2+\cdots+K^2 = K(K+1)(2K+1)/6$

3. $1^3+2^3+3^3+\cdots+K^3 = \left[\frac{K(K+1)}{2}\right]^2$

unacademy

A.P

$a, a+d, a+2d, \ldots a+(n-1)d$

$T_1 \quad T_2 \quad T_3 \quad \ldots T_n$

$T_2 - T_1 = (a+d) - a = d$

$T_3 - T_2 = (a+2d) - (a+d) = d$

$$S_n = \frac{n}{2}\left[a_1 + l\right] = \frac{n}{2}\left[a + (a+(n-1)d)\right]$$

$$= \frac{n}{2}\left[2a + (n-1)d\right]$$

unacademy

G.P

$T_1 = a$

$T_2 = a \cdot r$

$T_3 = ar^2$

$T_4 = ar^3$

$\vdots$

$T_n = ar^{n-1}$

$\frac{T_2}{T_1} = \frac{ar}{a} = r$

$\frac{T_3}{T_2} = \frac{ar^2}{ar} = r$

$S_n = \frac{a(r^n - 1)}{r - 1}$, $S_n = \frac{a(1 - r^n)}{1 - r}$

$S_\infty = \frac{a}{1 - r}$

2-Pts

```
for(i=1; i<=n; i++)
{
    for(j=i; j<=2*i; j++)
    {
        pf("Pankaj");
    }
}
```

i = 1	i = 2	i = 3	...	i = n
j = 1 to 2	j = 2 to 4	j = 3 to 6		j = n to 2n
②	4-2+1 ③	6-3+1 ④		2n-n+1 (n+1)

$S = 2 + 3 + 4 + \dots + (n+1)$

$$S = 2 + 3 + 4 + \cdots + (n+1)$$

$$\frac{n}{2}\left[2 + (n+1)\right] = \frac{n}{2}(n+3)$$

Q1

```
for(i=1; i<=n; i++)
{
  for(j=i; j<=3*i; j++)
  {
    printf("Pankaj");
  }
}
```

i=1	i=2	i=3
j = 1 to 3	j = 2 to 6	j = 3 to 9
3	6-2+1	9-3+1
	5	7

-- i = n

j = n to 3n

3n-n+1

2n+1

$S = 3 + 5 + 7 + \cdots + (2n+1)$

unacademy

$$S = 3 + 5 + 7 + \cdots + (2n+1)$$

$$= \frac{n}{2}\left[3 + (2n+1)\right]$$

$$= \frac{n}{2}\left[2n+4\right]$$

$$= \frac{n}{2}\left[2(n+2)\right]$$

$$= n(n+2)$$

H.W

```
for( i=1; i<=n; i=i+3)
{
    for (j=i; j<=n; j++)
    {
    } printf("Pankaj");
}
```

$$(\lfloor \log_2 n \rfloor + 1) \text{ times}$$

unacademy

THANK YOU!

Here’s to a cracking journey ahead!