CS & IT ENGINEERING

Operating System

Process Synchronization

Lecture - 02

Recap of Previous Lecture

- **Topic** System Call: Fork()
- **Topic** Synchronization
- **Topic** Race Condition
- **Topic** Critical Section

Topics to be Covered

- Topic: Synchronization
- Topic: Race Condition
- Topic: Mutual Exclusion

[NAT]

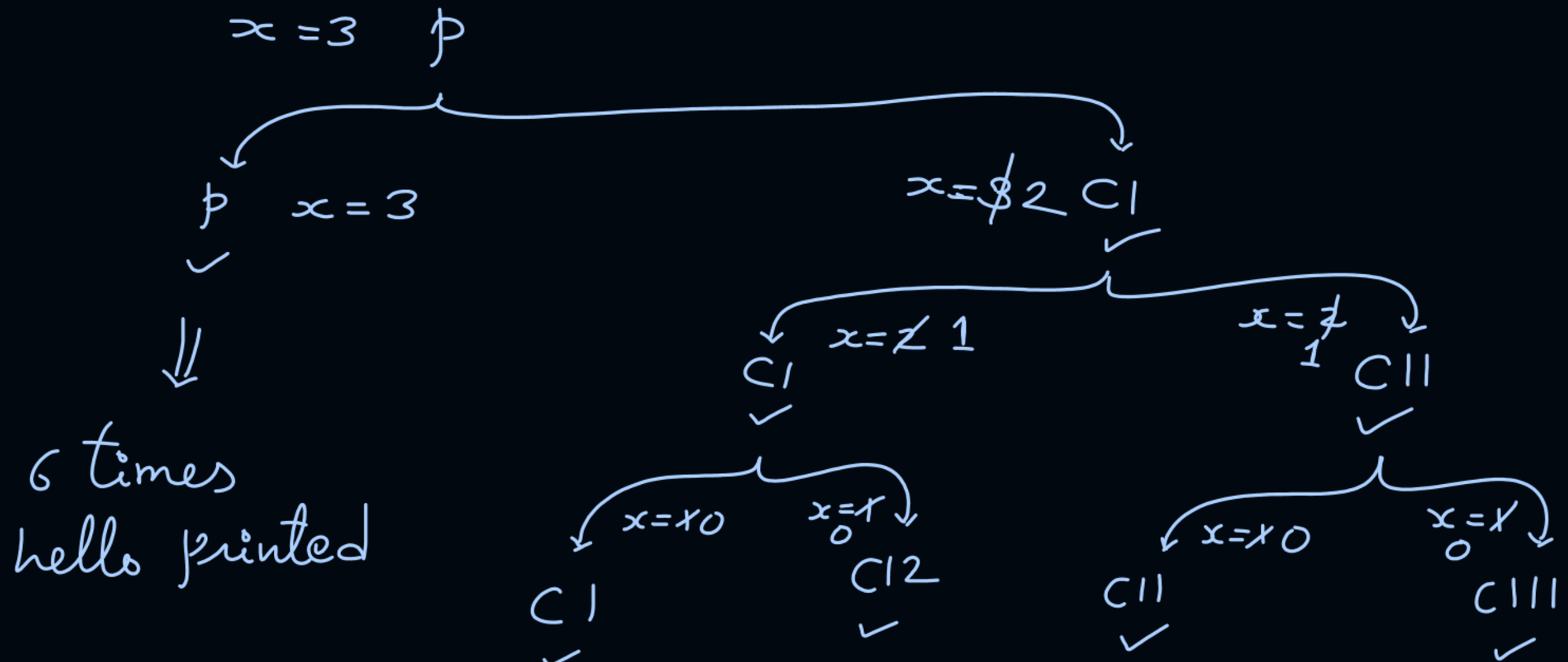
Ans = 14

#Q. Consider the following code snippet using the fork() and wait() system calls. Assume that the code compiles and runs correctly, and that the system calls run successfully without any errors.

```
int x = 3;
while(x > 0) {
fork();
printf("hello");
wait(NULL);
x--;
}
```

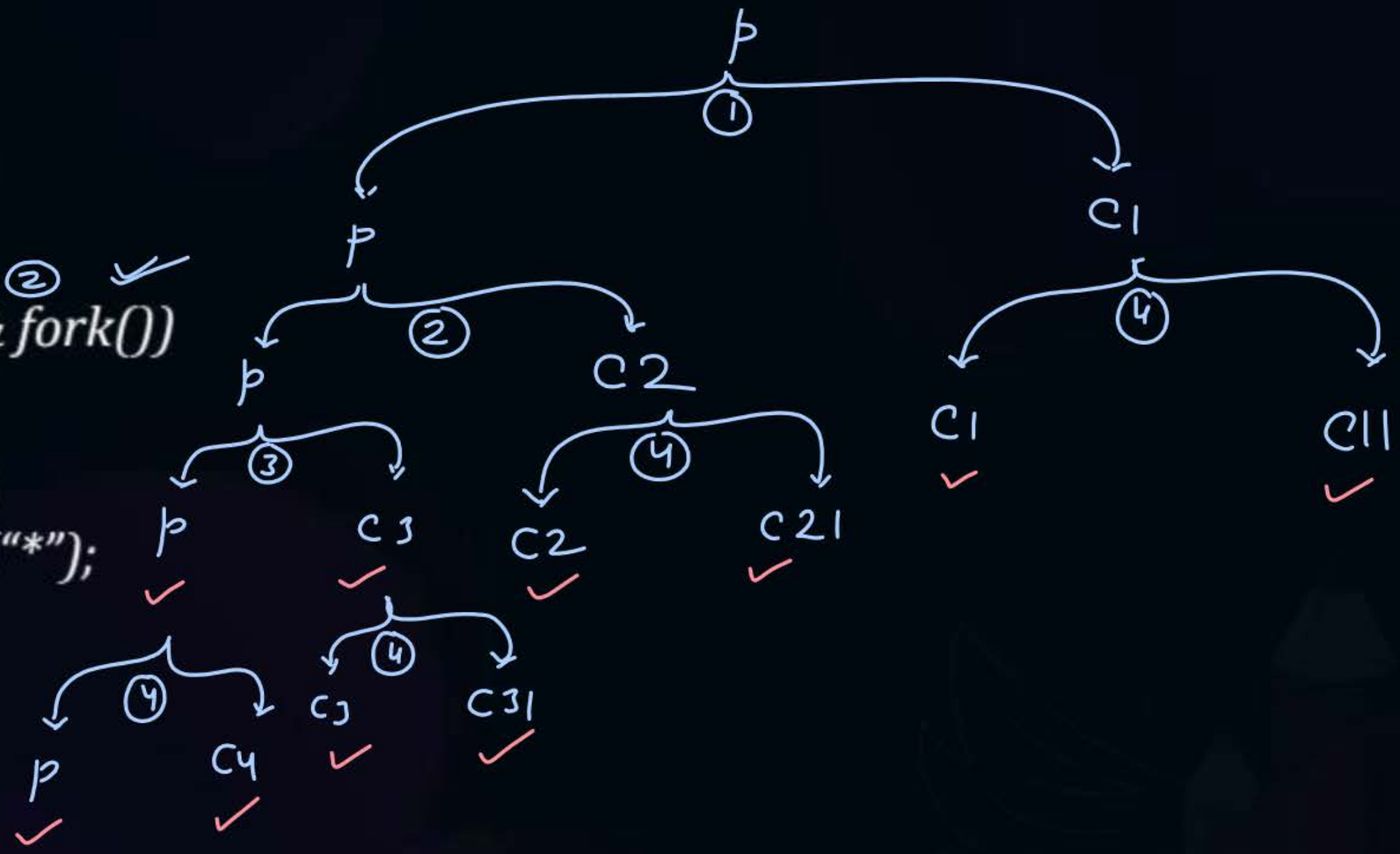
The total number of times the printf statement is executed is _____?

GATE-2024


x = 3 p
p x = 3
x = 2 C1
x = 1 C1
x = 1 C11
x = 0 x = 1
C1 C12
x = 0 x = 1
C11 C111
6 times
hello printed

[NAT]

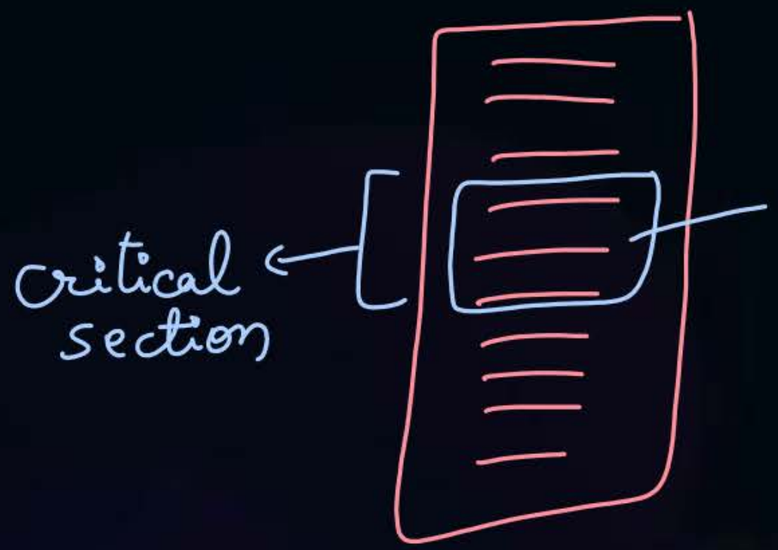
```
void main()
{
 if (fork() && fork())
  {
      fork();
      printf("*");
  }
 fork();
printf("*");
}
```



Ans = 10

Topic : Critical Section

The critical section is a code segment where the shared variables can be accessed.



Topic : Race Condition

A race condition is an undesirable situation, it occurs when the final result of concurrent processes depends on the sequence in which the processes complete their execution.

ex:- int x = ~~7~~ 8;

Processes

P1	P2
R1 = x	R2 = x
R1 = R1 + 2	R2 = R2 + 3
x = R1	x = R2

P1	P2	P1	P2	P1	P2
R1 = x	R2 = x	R1 = R1+2	R2 = R2+3	x = R1	x = R2

P1

R1 = ~~8~~ 7

P2

R2 = ~~5~~ 8

x = ~~5~~ ~~8~~ 7

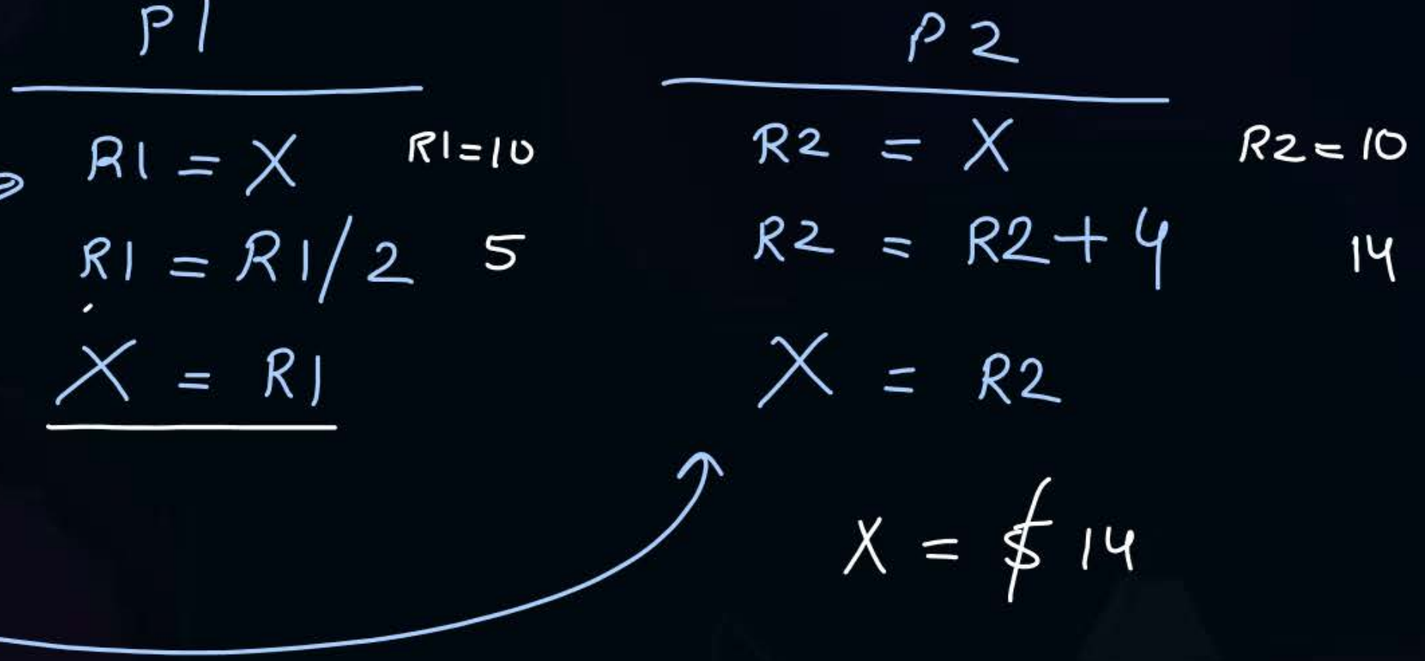
	P2	P1
	x = R2	x = R1

[NAT]

#Q. X = 10

P1
X = X /2

P2
X = X+4



How many different values of X are possible after both processes finish executing?

Ans = 4 (5, 7, 9, 14)

Case 1:-

execute P1 Completely and then P2.

$X = 10 \not\rightarrow 9$

Case 2:-

execute P2 Completely and then P1.

$X = 10 \not\rightarrow 7$

Case 3:-

Both P1 and P2 reads X = 10 Concurrently and P2 finished last.

$X = 10 \not\rightarrow 14$

Case 4:-

Both p1 & p2 reads $x = 10$ Concurrently and p1 finishes last

$x = \cancel{10}\ \cancel{14}\ 5$

[NAT]

#Q. The following pair of processes share a common variable X.

Process A	Process B
int Y;	int Z;
8 Y = X*2;	5 Z = X+1;
X = Y;	X = Z;

① ⇒ x = 4 8 9

② ⇒ x = 4 5 10

③ ⇒ x = 4 8 5

④ ⇒ x = 4 5 8

X is set to 4 before either process begins execution. As usual, statements within a process are executed sequentially, but statements in process A may execute in any order with respect to statements in process B.

How many different values of X are possible after both processes finish executing?

5, 8, 9, 10

Ans = 4

#Q $x = 8$

P1

$x = x - 2$

8

6

P2

$x = x + 3$

8

11

max possible value of x = 11 ?

min 11 of x = 6 ?

#Q $x = 8$

P1	P2
$x = x + 2$	$x = x + 3$
8 10	8 11

max possible value of x = 13 ?

min —— 11 —— x = 10 ?

Run P1 and P2 one after another

$x = 13$

#Q $x = 10$

P1

$x = x - 2$

~~10~~ 8

P2

$x = x - 3$

~~10~~ 7

P3

$x = x + 4$

~~10~~ 14

$x = ~~14~~$ 8

max possible value of x = 14 ?

min ___ 11 ___ = 5 ?

Distinct possible values of x = 7 ?

$x - 2 - 3 + 4$

---

final result of any 1 process only. 8, 7, 14

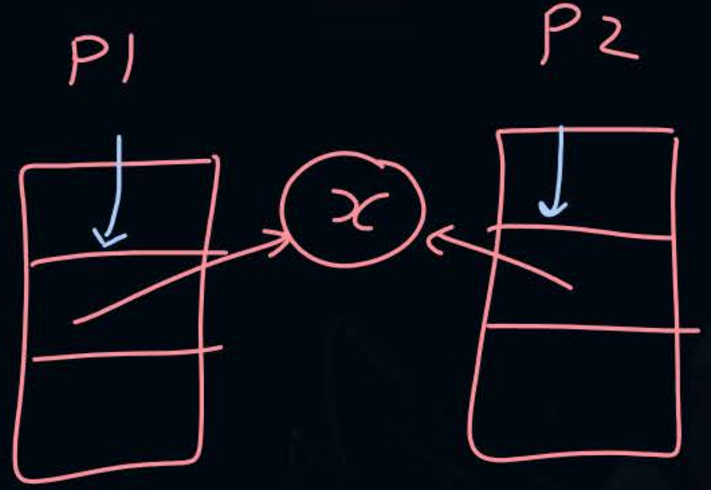
— 11 — any 2 processes ⇒ 5, 12, 11

— 11 — all 3 processes ⇒ 9

Topic : Solution of Critical Section Problem

Requirements of Critical Section problem solution:

1. Mutual Exclusion
2. Progress
3. Bounded Waiting



2 mins Summary

Topic Synchronization

Topic Race Condition

Topic Mutual Exclusion

/vishvadeepsir

Happy Learning

THANK - YOU