# CS & IT ENGINEERING

## Operating System

### Process Synchronization

**Lecture No. 9**

**By- Dr. Khaleel Khan Sir**

## TOPICS TO BE COVERED

- Problem Solving with Semaphore
- Classical IPC Problems
- Producer-Consumer Problem

Both Binary & Counting Semaphore Can be used to Solve the Problem of C.S

RQ

~~$P_3$~~ ~~$P_2$~~

BSEM S = ~~1~~ 0

$P_1$ $P_2$ $P_3$

Entry: P(S);

<CS> ~~$P_2$~~

Exit: V(S); $P_2$

L <FIFO>

~~$P_2$~~ ~~$P_3$~~ $P_1$

L

~~$P_2$~~ ~~$P_3$~~

CSEM S = ~~1~~ ;

~~0~~ ~~-1~~ $\boxed{\cancel{-2}}$ ~~-1~~ ~~0~~ 1

{

CS [ ]

}

1) M.E : ✓

2) Progress : ✓

3) Bound. Wait : ✓

Q

Bsem S = 1 ,T= 1

```
Pr (i)
{
  while (1)
  {
    P(S);
    P(T);
    <CS>
    V(T);
    V(S);
  }
}
```

```
Pr (j)
{
  while (1)
  {
    P(T);
    P(S);
    <CS>
    V(S);
    V(T);
  }
}
```

UM

Pre

Blocked

$P_i$

Blocked

Condition for deadlock

PreEmption b/w the Two Down opn

Guarantee M/E but is NOT deadlock free

Does it guarantee ME? How about Deadlock?

## Q

```
BSem S = 1, T = 0;
Pr (i)
{
  while (1)
  {
    P(T);
    Print ('1');
    Print ('1');
    V(S);
  }
}

Pr (j)
{
  while (1)
  {
    P(S);
    Print ('0');
    Print ('0');
    V(T);
  }
}
```

CS

<NCS>

CS

What is the Regular Expression that gets generated?

$(0011)^+$

001100110011 . . . . . .

1) O/P : as a Series

2) M/E : ✓

3) Progress : ✗

4) Bound wait : ✓

Impl. of STRICT ALTERNATION

Using SEM

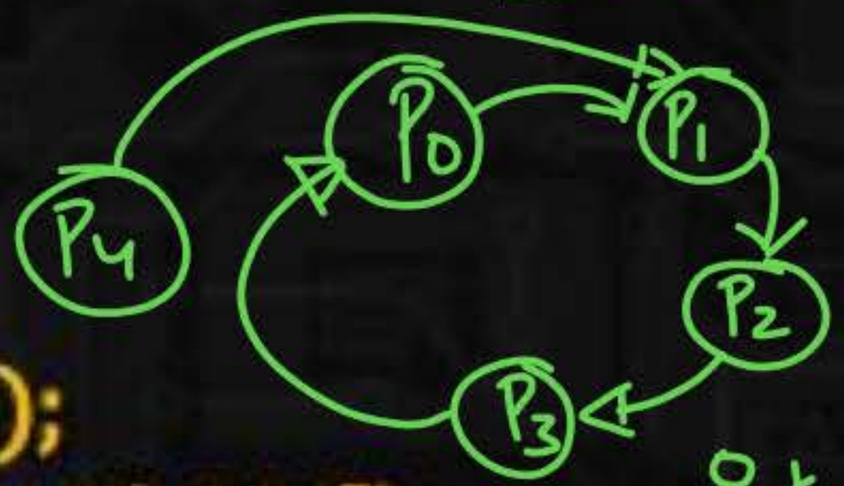
Q

```
Bsem m [0....4] = {1};
Pr (i) i = 0, 4
{
    while (1)
    {
        P(m[i]);
        P(m[(i+1) %4]);
        <CS>
        V(m[i]);
        V(m[(i+1) %4]);
    }
}
```

array

P4Q

deadlock?




$0\ \cancel{1} = m[0] = \cancel{1}\ 0$

$0\ \cancel{1} = m[1] = \cancel{1}\ 0$

$0\ \cancel{1} = m[2] = \cancel{1}\ 0$

$0\ \cancel{1} = m[3] = \cancel{1}\ 0$

$0\ \cancel{1} = m[4] = \cancel{1}\ 0$

(i) Guarantee M/E : ?

- NO -

(ii) If M/E is NOT Guaranteed then Max # of Processes that can be in CS ? : 2

$P_1$ : X ; $P_3$ : X

$P_4$ : X

$\boxed{P_0\ P_2}$ CS

What is the maximum no. of processes that can be in <CS>.

**Q.** BSem S = 1, T = 0, Z = 0

```
Pr (i)
{
  While (1)
  {
    P(S);
    Print (*);
    V(T);
    V(Z);
  }
}

Pr (j)
{ While(1){
    P(T);
    V(S);
  }
}

Pr (k)
{
    P(Z);
    V(S);
}
```

What is the minimum and maximum no. of "*" that get printed.

Pri ; Prj ; PrK ; Pri ← Blked

I : * * : Min

Pri ; Prj ; Pri ; PrK ; Pri

II : * * *

∞

$(*)^+$

BSEM S=1, T=1, Z=1

Ans: 2

```
Pri
{
  while(1)
  {
    P(S);
    print('*');
    V(T);
    V(Z);
  }
}
```

```
Prj
{
  P(T);
  V(S);
}
```

```
Prk
{
  P(Z);
  V(S);
}
```

Min=1 <Prj; Prk; Pri; Pri> *

Max=3; <Pri; Prj; Pri; Prk; Pri; Pri>

Q

2022

Consider the following ~~threads~~ Process, T1, T2 and T3 executing on a single processor, synchronized using three binary semaphore variables, S1, S2, and S3, operated upon using standard wait() and signal(). The ~~threads~~ Processes can be context switched in any order and at any time.

Which initialization of the semaphores would print the Sequence BCABCABCA....?

| $T_1$ | $T_2$ | $T_3$ |
|---|---|---|
| While(true){ | While(true){ | While(true){ |
| Wait($S_3$); 0 | Wait($S_1$) 1 | Wait($S_2$) 0 |
| Print("C"); | Print("B"); | Print("A"); |
| Signal($S_2$);} | Signal($S_3$);} | Signal($S_1$);} |

A. S1 = 1; S2 = 1; S3 = 1

B. S1 = 1; S2 = 1;S3 = 0

✓ C. S1 = 1; S2 = 0; S3 = 0

D. S1 = 0; S2 = 1; S3 = 1

$S_1 = 1$
$S_2 = 0$
$S_3 = 0$

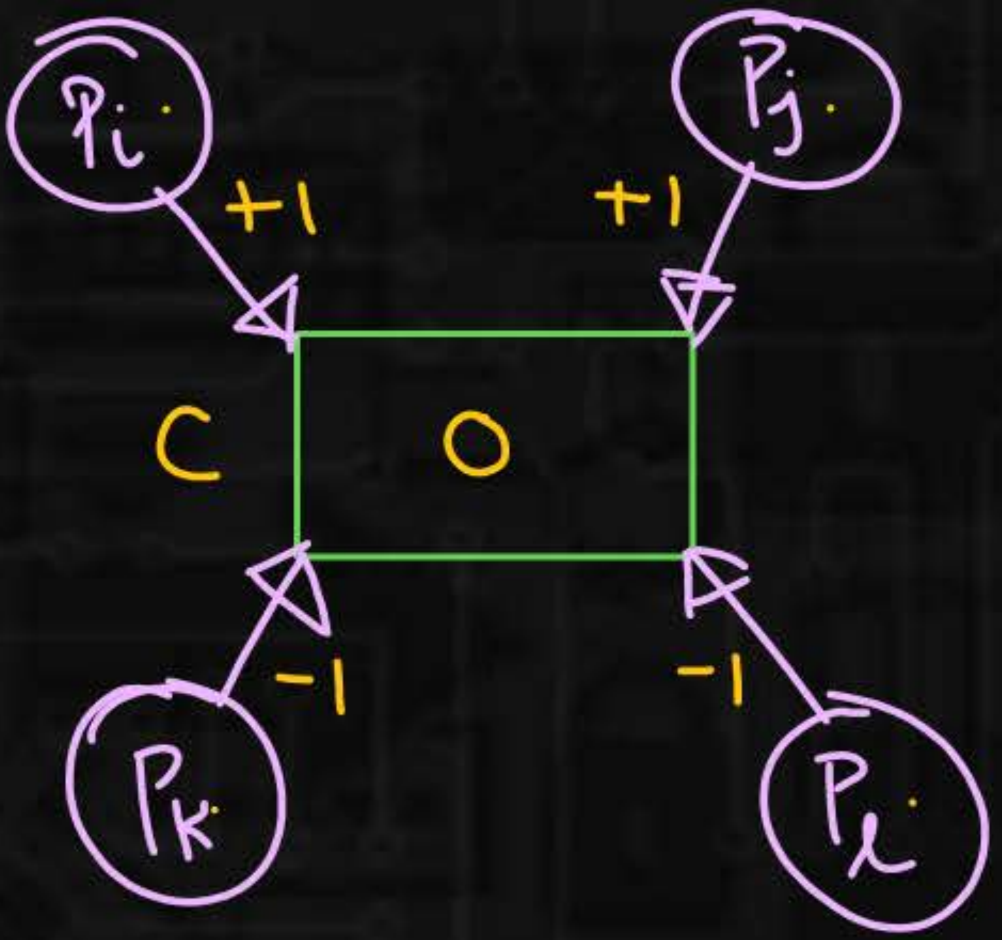
$T_2 \rightarrow T_1 \rightarrow T_3$

**Q**

H/W

Consider a counting semaphore initialized to 2, there are 4 concurrent processes $P_i$, $P_j$, $P_k$ & $P_L$. The Processes $P_i$ & $P_j$ desire to increment the current value of variable C by 1 whereas $P_K$ & $P_L$ desire to decrement the current value of C by 1. All Processes perform their update on C under semaphore control. What can be the minimum and maximum value of C after all Processes finish their update.

CSEM $\underline{S=2}$ ;




Assuming initial value of C is ∅ ;

```
Process(x)        x = i, j, k, l
{                     +1     -1
   P(S);
   <C = C ± 1>    Load
   V(S);          Inc/Dcr
}                 Store
```

**Q**

Consider Three Processes using four Binary Semaphores a, b, c, d in the order shown below.

$a = \cancel{1}\ 0;\ b = \cancel{1}\ 0;\ c = \cancel{1}\ 0;\ d = 1$

OS Concepts
9th/10th Ed

Which Sequence is a Deadlock Free sequence?

Deadlock

(I) X: P(a); P(b); P(c);
Y: P(b); P(c); P(d);
Z: P(c); P(d); P(a);

✓

(II) X: P(b); P(a); P(c);
Y: P(b); P(c); P(d);
Z: P(a); P(c); P(d);

$a = \cancel{1}\ 0;\ b = \cancel{1}\ 0;\ c = \cancel{1}\ 0;\ d = \cancel{1}\ 0$

(III) X: P(b); P(a); P(c);
Y: P(c); P(b); P(d);
Z: P(a); P(c); P(d);

(IV) X: P(a); P(b); P(c);
Y: P(c); P(b); P(d);
Z: P(c); P(d); P(a);

$a = \cancel{1}\ 0;\ b = \cancel{1}\ 0;\ c = \cancel{1}\ 0;\ d = 1$

$a = \cancel{1}\ 0;\ b = \cancel{1}\ 0;\ c = \cancel{1}\ 0;\ d = 1$

Deadlock

**Q**

Each of a set of n processes executes the following code using two semaphores a and b initialized to 1 and 0, respectively. Assume that count is a shared variable initialized to 0 and not used in CODE SECTION P.

```
CODE SECTION P                      NCS
wait(a); count=count+1;
if (count==n) signal (b);
signal (a): wait (b) ; signal (b);
CODE SECTION Q                      P3 P2 P1
```

$a = \cancel{1}$ ; $b = \cancel{0}\ 1$ $\quad$ $\boxed{n = 3}$

P1 P2 P3 $\quad$ count $\boxed{\cancel{2}\ \cancel{0}\ \cancel{1}}\ 3$

What does the code achieve ?

A. It ensures that no process executes CODE SECTION Q before every process has finished CODE SECTION P.

B. X It ensures that two processes are in CODE SECTION Q at any time.

C. X It ensures that all processes execute CODE SECTION P mutually exclusively.

D. It ensures that at most n−1 processes are in CODE SECTION P at any time.

Consider the two functions Incr( ) and Decr( ) G-2023

```
Incr( ){
    wait(s);
    x = x + 1;
    signal(s);
}

Decr( ) {
    wait(s);
    x= x - 1;
    signal(s);
}
```

5

3

L R, x
In/Dc R
Store x, R

x = 10

I1: BSEM S = 1

(10 + 5 − 3)

= 12

Five Processes invoke Incr( ) and Three Processes invoke Decr( )

X is a shared variable initialized to 10.

$I_1$ : s value is 1 (Binary semaphore) : Max|Min = 12

$I_2$ : s value is 2 (Counting semaphore)

Let $V_1$ and $V_2$ be the minimum possible values of the implementation of I1 and I2, then choose the values of x for $V_1$ and $V_2$.

CSEM S=2

(A) 12, 7 ✓

(B) 7, 7 ×

(C) 15, 7 ×

(D) 12, 8

# Application of SEMAPHORE

## II. Classical IPC Problems.

- Producer-Consumer
- Reader-Writer Problem
- Dining-Philosopher


THANK YOU
GW
SOLDIERS !