Deadlock Detection Algorithm

	Allocation			Request			Available		
	A	B	C	A	B	C	A	B	C
P_0	0	1	0	0	0	0	0	0	0
P_1	2	0	0	2	0	2			
P_2	3	0	3	0	0	0			
P_3	2	1	1	1	0	0			
P_4	0	0	2	0	0	2			

Total
(A B C)
= (7 2 6)

1. work = (0 0 0)

   Finish[i] = false, i = 0, 1, 2, 3, 4

2. Finish[0] = false; $Request_0 = (0\ 0\ 0) \leq (0\ 0\ 0)$ ← work

   Finish[0] = true;

   ∴ work = work + $allocation_0$

   $= (0\ 0\ 0) + (0\ 1\ 0)$

   $= (0\ 1\ 0)$

3. Finish[2] = false; $Request_2 \leq work$

   Finish[2] = true; $(0\ 0\ 0) \leq (0\ 1\ 0)$

   ∴ work = work + $allocation_2$

   $= (0\ 1\ 0) + (3\ 0\ 3) = (3\ 1\ 3)$

4. Finish[3] = false; $Request_3 \leq work$

   Finish[3] = true; $(1\ 0\ 0) \leq (3\ 1\ 3)$

   ∴ work = work + $allocation_3$

   $= (3\ 1\ 3) + (2\ 1\ 1)$

   $= (5\ 2\ 4)$

Scanned with CamScanner

5. Finish[1] = false; $Request_1 \leq work$

$(2\ 0\ 2) \leq (5\ 2\ 4)$

Finish[1] = true;

$\therefore$ work = work + $allocation_1$

$= (5\ 2\ 4) + (2\ 0\ 0)$

$= (7\ 2\ 4)$

6. Finish[4] = false; $Request_4 \leq work$

$(0\ 0\ 2) \leq (7\ 2\ 4)$

Finish[4] = true;

$\therefore$ work = work + $allocation_4$

$= (7\ 2\ 4) + (0\ 0\ 2)$

$= (7\ 2\ 6)$

Hence, Finish[i] = true; for all i,

with the sequence, $\langle P_0, P_2, P_3, P_1, P_4 \rangle$

IV Suppose P_2 has an additional request for C.

$\rightarrow (0\ 0\ 1)$

$\therefore$

	Allocation			Request			Available		
	A	B	C	A	B	C	A	B	C
P_0	0	1	0	0	0	0	0	0	0
P_1	2	0	0	2	0	2			
P_2	3	0	3	0	0	1			
P_3	2	1	1	1	0	0			
P_4	0	0	2	0	0	2			

Total (A B C) = (7 2 6)

1. work = (0 0 0)
2. $Request_0 \leq work$; work = work + $allocation_0$ = (0 1 0)
3. Now, none of $\boxed{Request_i \leq work}$

$\therefore$ Deadlock of Processes, P_1, P_2, P_3 and P_4.

Scanned with CamScanner