Lec -22

* Deadlock avoidance :-

① Current state → ① no. of processes
② max need of R. of each proc.
③ Currently allocated amount of R. to each Proc.
④ Max. amount of each resource.

schedule R(s) R(s)

↳ schedule.
Process & Resources.

→ System safe state

safe state

P_1 P_2 P_3 P_4 →

$P_1 \rightarrow P_3 \rightarrow P_4 \rightarrow P_2$

P	allocated A	allocated B	allocated C	max. need A	max. need B	max. need C	available A	available B	available C	Remaining need A	Remaining need B	Remaining need C	
P_1	0	1	0	7	5	3	3	3	2	7	4	3	P_1
P_2	2	0	0	3	2	2	5	3	2	1	2	2	P_2
P_3	3	0	2	9	0	2	7	4	3	6	0	0	P_3
P_4	2	1	1	4	2	2	7	4	5	2	1	1	P_4
P_5	0	0	2	5	3	3	7	5	5	5	3	1	P_5
Total already:	7	2	5				10	5	7				

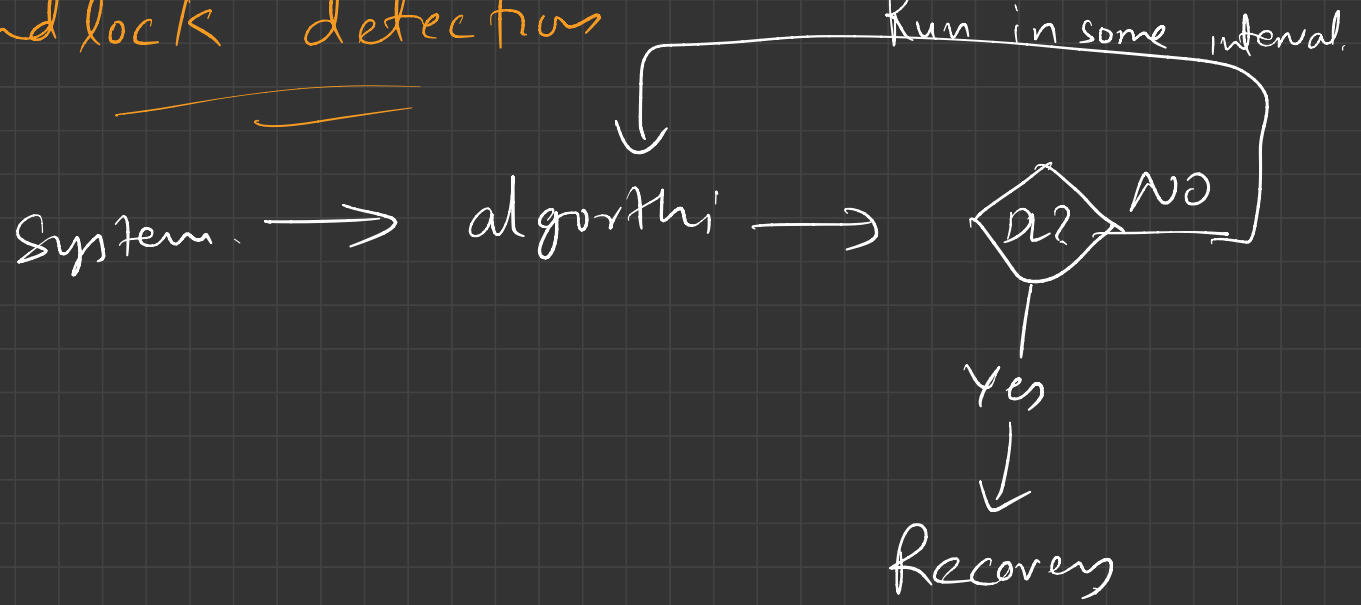
Total ⇒ A = 10, B = 5, C = 7

$P_2 \rightarrow P_4 \rightarrow P_5 \rightarrow P_1 \rightarrow P_3$

↳ safe state

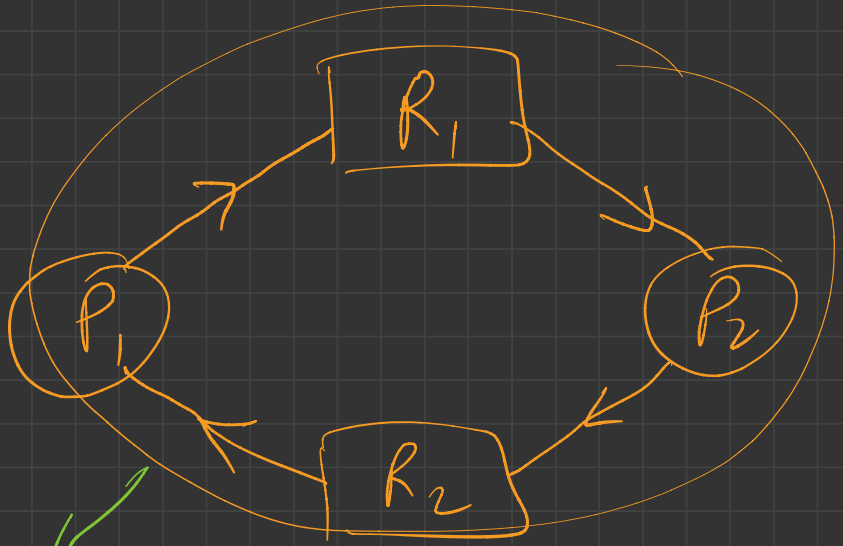
unsafe state

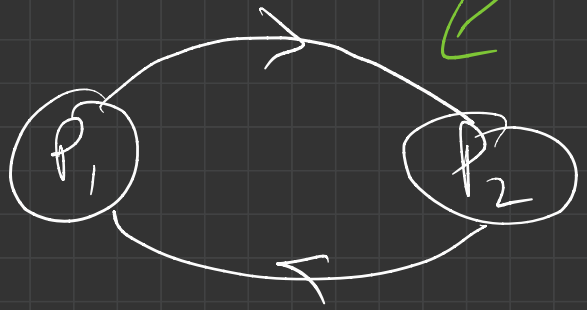
* Deadlock detection



① Single Instance of each resource

* wait-for graph





Cycle ⇒ DL ✓

No cycle ⇒ DL X.

② Multiple Instances.

— Banker's algorithm.

→ Safe sequence available → No DL.

→ No safe seq. available → DL detected.