- If there is cycle in the wait-for-graph, then the system is in deadlock.

Deadlock Recovery

- PPT

Process Termination : PPT

- Factors on which the term 'minimum cost' depends — PPT

Deadlock Avoidance

Example 1

Max resource : 12 tape drives

	Max needs	Current Allocation
P_0	10	5
P_1	4	2
P_2	9	2

- There are 3 free tape drives.
- The OS is in a safe state, since $\langle P_1, P_0, P_2 \rangle$ is a safe sequence.

- Banker's Algorithm (Used for deadlock avoidance)

PPT

1.

	Allocation	Max.	Available	Need
	A B C	A B C	A B C	A B C
P_0	0 1 0	7 5 3	3 3 2	7 4 3
P_1	2 0 0	3 2 2		1 2 2
P_2	3 0 2	9 0 2		6 0 0
P_3	2 1 1	2 2 2		0 1 1
P_4	0 0 2	4 3 3		4 3 1

For P_0

request $\rightarrow (7, 4, 3) > (3, 3, 2) \rightarrow$ available

For P_1

$(1, 2, 2) \leq (3, 3, 2)$

$\therefore$ OS grants P_1's request

$\therefore$ Available $\rightarrow (2, 1, 0)$

$\therefore$ after P_1 releases its resources, available $\rightarrow (2,1,0) + (3,2,2) = (5,3,2)$

For P_2

$(6,0,0) > (5, 3, 2)$

For P_3

$(0,1,1) \leq (5,3,2)$

$\therefore$ OS grants P_3's request.

$\therefore$ available $\rightarrow (5,2,1)$

$\therefore$ after P_3 releases its resources, available $\rightarrow (5,2,1) + (2,2,2) = (7,4,3)$

For P_4

$$(4,3,1) \leq (7,4,3)$$

$\therefore$ available $\rightarrow (3,1,2)$

$\therefore$ after P_4 releases its resources,

available $\rightarrow (3,1,2) + (4,3,3)$

$= (7,4,5)$

$\therefore$ For P_0

$$(7,4,3) \leq (7,4,5)$$

$\therefore$ available $\rightarrow (0,0,2)$

$\therefore$ after release, available $\rightarrow (0,0,2) + (7,5,$

$= (7,5,5)$

$\therefore$ ~~For P_2~~

$\cancel{(7,5,5)}$

$\therefore \langle P_1, P_3, P_4, P_0, P_2 \rangle \rightarrow$ safe state

2.

	Allocation	Max	Available	Need
	A B C	A B C	A B C	A B C
P_0	~~0 1 0~~ 1 1 2	4 3 3	2 1 0	~~4 0 3~~ 3 2 1
P_1	~~2 0 0~~ 2 1 2	3 2 2		~~0 2 0~~ 1 1 0
P_2	~~3 0 2~~ 4 0 1	9 0 2		~~6 0 0~~ 5 0 1
P_3	~~2 1 1~~ 0 2 0	7 5 3		~~5 4 2~~ 7 4 3 3
P_4	~~0 0 2~~ 1 1 2	11 2 3		~~11 0 1~~ 10 1 1

$\therefore$ For P_0

$\cancel{(4,2,3) > (2,1,0)}$ $(3,2,1) > (2,1,0)$

For P_1

$\cancel{(4,2,0) > 2,1,0}$

~~1,1~~

$(1,1,0) \leq (2,1,0)$

$\therefore$ available $\rightarrow (1,0,0)$

$\therefore$ after release $\rightarrow (1,0,0) + (3,2,2)$

$= (4,2,2)$

For P_2

$(5,0,1) > (4,2,2)$

For P_3

X

For P_4

X

$P_1, P_0, P_2, P_3,$
P_4
$\downarrow$
unsafe

For P_0

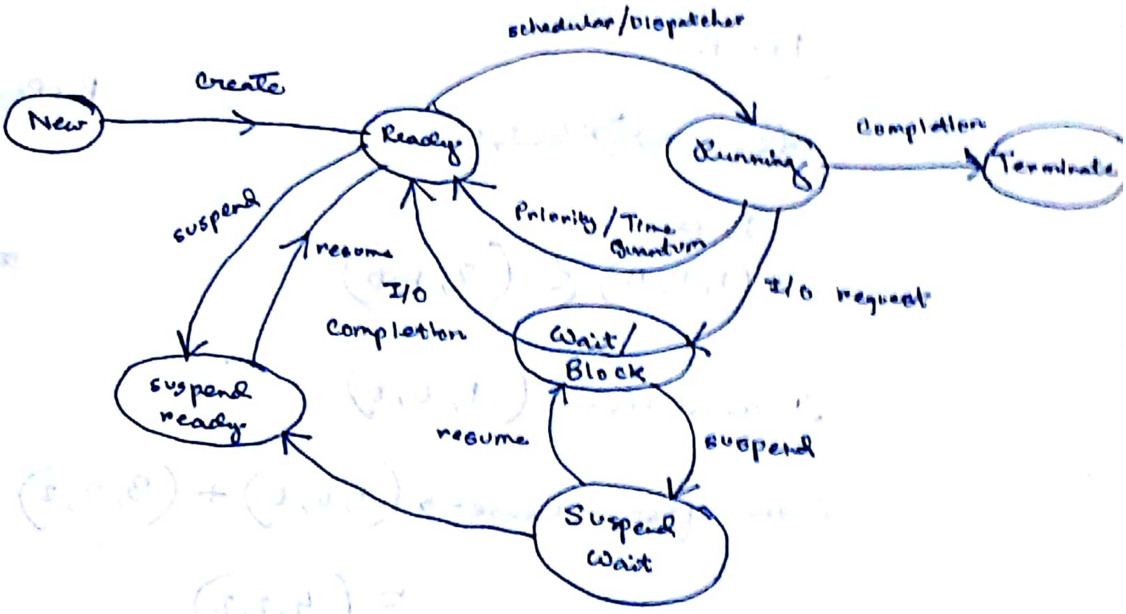
$(3,2,1) \leq (4,2,2)$

$\therefore$ available $\rightarrow (1,0,1)$

$\therefore$ after release $\rightarrow (1,0,1) + (4,3,3)$

$= (5,3,4)$

Process State Diagram



Roles of Semaphores—

1) Process Synchronization
2) ~~Maintaining~~ Controlling order of Execution of Process
3) Managing Resources

2) P1 P2 P3

S1 = 0, S2 = 0

$P2 \rightarrow P1 \rightarrow P3$

R A C

Initialize both semaphores to 0.

P1	P2	P3
wait (S1)	- - -	-wait(S2).
- - -	- - -	- - -
- - -	- - -	- - -
signal (S2)	signal(S1)	wait(S2)

3) For managing resources

P_i

{

wait(S)

C.S.

signal(S)

R.S.

}

- signal – V()
- wait – P()

- 2 process P, Q

2 binary semaphores S, T

Process P	Process Q
while(1)	while(1)
{	{
P(S)	P(T)
print '0'	print '1'
print '0'	print '1'
V(S)	V(T)
}	}

O/P: 001100110011

	W	X	Y	Z
a)	P(S)	V(S)	P(T)	V(T)
b)	P(S)	V(T)	P(T)	V(S)
c)	P(S)	V(T)	P(T)	V(S)
d)	P(S)	V(S)	P(T)	V(T)

S, T = 1

S = 1, T = 0

S, T = ~~0~~ 1

S = ~~1~~, T = ~~1~~ 0

~~0~~ 1