wait & signal.

2) It can be defined as a situation where processes wait indefinitely within the semaphore.

3) RAG – Resource Allocation Graph.

Deadlocks can be described more precisely in terms of directed graph called RAG.

4) - Mutual Exclusion
- Hold & Wait
- No-~~premeation~~ preemption.
- Circular wait

5) Circular wait → A set $\{P_0, P_1, \ldots P_n\}$ of waiting processes must exist such that P_0 is waiting for resource held at P_1 & P_1 for P_2 so on ... P_{n-1} for P_n & P_n waiting for resource held at P_0.

6) Hold & Wait → A process must hold atleast one resource & should wait to acquire ~~ma~~ additional resources that are currently being held by other processes

7) No-preemption → Resources can't be preempted, i.e resource can be released only voluntarily by the process holding it, after that process has completed its task.

8) Request edge → In RAG, a directed edge from process P_i to resource R_j denoted as $P_i \to R_j$ is called as Request edge.

$P_i \to R_j$ signifies that P_i requested an instance of resource R_j & currently waiting for that resource.

9) A directed edge $R_j \to P_i$ is assignment edge signifies that an instance of resource type R_j has been allocated to P_i.

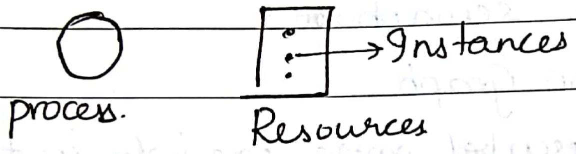
Page

(A) Deadlock:- Two or more processes are waiting indefinetly for an event that can be caused only by one of its waiting process which such state is reached these processes are said to be in deadlock.

Scanned by CamScanner

10A) A claim edge $P_i \rightarrow R_j$ indicates that process P_i may request resource R_j which is represented by dashed line.
A claim edge can be converted to request edge when a process requests a resource.
When resource is released by a process, the assignment reconverts to claim edge.

11A)



12A) with dashed lines - - - ->

13A) A state is safe if the system can allocate resources to each process in some order & still avoid a deadlock.
- A system is in safe state only if there exists safe sequence.

14A) Sequence that is followed by the system to be in safe state is called safe sequence.

15A) A sequence followed by the system to be in unsafe state then it is unsafe sequence, then deadlock occurs.

19 A) – Resource-Allocation Graph Algo
– Bankers Algo
– Safety Algo
– Resource-Request Algo

20 A) Available, Max, Allocation, Need.

21 A) Available → m – no. of available resources of each type.
if available[j] = k → k instances of resource R_j available

22 A) Max → n×m matrix → defines maximum demand of each process
max[i, j] = k, P_i may request k instances of R_j

23 A) Allocation → n×m → no. of resources allocated currently to each process.
allocation[i, j] = k P_i is currently allocated k instances of R_j

24 A) Need → n×m → remaining resources need of each process
need[i, j] = k P_i may need k instances of R_j to complete task.

need = max - allocation

25 A) Need[i, j] = Max[i, j] – Allocation[i, j]

26 A) If all resources have only single instance, then we can define a deadlock detection algorithm that uses a variant of ~~reso~~ RAG called wait-for graph.

27 A) If there is a resource in b/w 2 process which is requested by one process & is assigned to other process then that resource can be removed and the wait for graph can be obtained.

28A) 1) Process Termination.

-Abort all deadlock processes

-Abort 1 process at a time until deadlock cycle is eliminated.

2) Resourse preemption

– Selecting a victim – Roll back – Starvation.

29A) – Selective a victim

– Rollback

– Starvation

30A) It is a operation which returns back the databases to previous states.