OPERATING SYSTEM

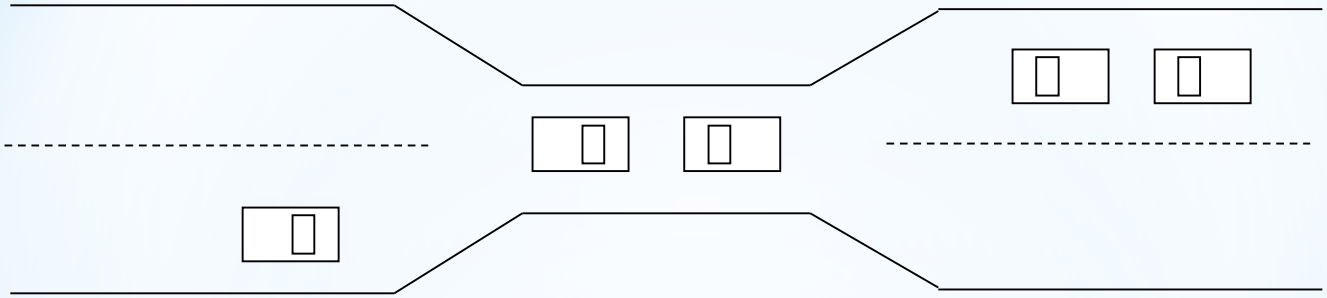
DEAD LOCKS

Operating system

THE DEADLOCK PROBLEM

In a computer system deadlocks arise when members of a group of processes which hold resources are blocked indefinitely from access to resources held by other processes within the group.

BRIDGE CROSSING EXAMPLE



* Traffic only in one direction.
* Each section of a bridge can be viewed as a resource.
* If a deadlock occurs, it can be resolved if one car backs up (preempt resources and rollback).
* Several cars may have to be backed up if a deadlock occurs.
* Starvation is possible.

THE DEADLOCK CHARACTERIZATION

1.Necessary and sufficient conditions

2.Resource allocation graph

CONDITIONS FOR DEADLOCKS

1.Mutual exclusion. No resource can be shared by more than one process at a time.

2.Hold and wait. There must exist a process that is holding at least one resource and is waiting to acquire additional resources that are currently being held by other processes.

3.No preemption. A resource cannot be preempted.

4.Circular wait. There is a cycle in the wait-for graph.

RESOURCE ALLOCATION GRAPH

A set of vertices V and a set of edges E.

V is partitioned into two types:

$P = \{P_1, P_2, \ldots, P_n\}$, the set consisting of all the processes in the system.

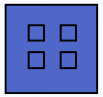
$R = \{R_1, R_2, \ldots, R_m\}$, the set consisting of all resource types in the system.

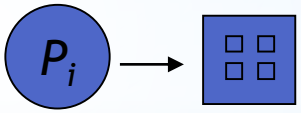
request edge – directed edge $P_1 \rightarrow R_j$

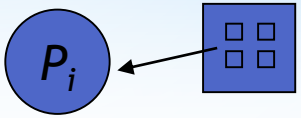
assignment edge – directed edge $R_j \rightarrow P_i$

RESOURCE-ALLOCATION GRAPH (CONT.)

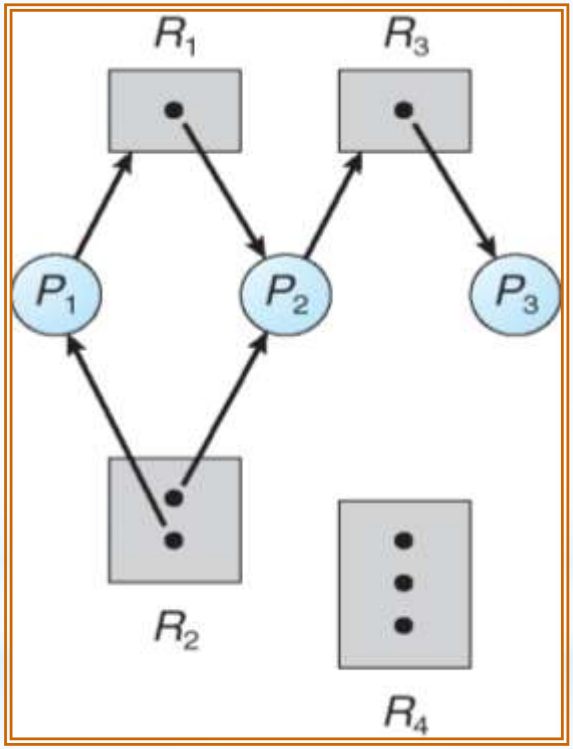
Process 

Resource Type with 4 instances 

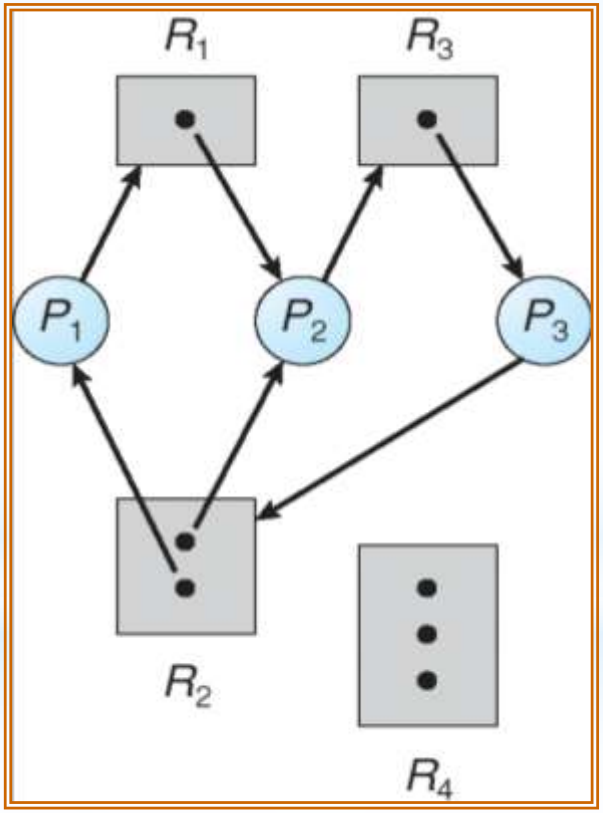
P_i requests instance of R_j 

P_i is holding an instance of R_j 

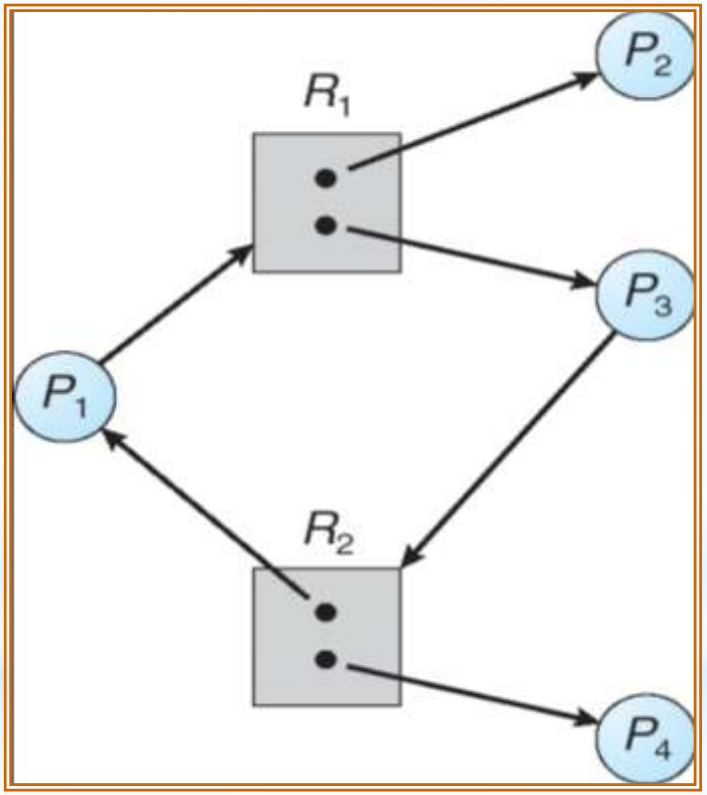
EXAMPLE OF A RESOURCE ALLOCATION GRAPH



RESOURCE ALLOCATION GRAPH WITH A DEADLOCK



GRAPH WITH A CYCLE BUT NO DEADLOCK



Chaitanya
Women's College
(DEGREE & P.G)
Affiliated to Andhra University

THANK YOU