

### Conditions for Deadlock to occur -

Deadlock can arise if following four conditions hold simultaneously (Necessary Conditions)

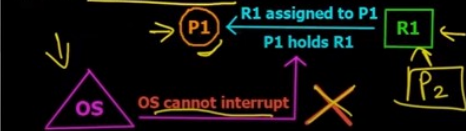
#### 1) Mutual Exclusion - ✓

One or more than one resource are non-sharable (Only one process can use at a time)



#### 3) No Preemption - ✓

A resource cannot be taken from a process unless the process releases the resource.



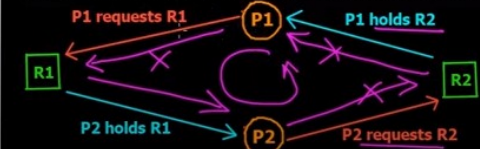
#### 2) Hold and Wait -

A process is holding at least one resource and waiting for other resources.



#### 4) Circular Wait -

A set of processes are waiting for each other in circular form.



### Methods for handling deadlock -

There are three ways to handle deadlock :

#### 1) Deadlock prevention or avoidance -

>> The idea is to not let the system into deadlock state.

#### 2) Deadlock detection and recovery -

>> Let deadlock occur, then do preemption to handle it once occurred.

#### 3) Ignore the problem all together -

>> If deadlock is very rare, then let it happen and reboot the system.

>> Ignore the problem and pretend that deadlocks never occur in the system

3 Strategies to handle deadlocks :

#### 1) Preemption -

>> We can take a resource from one process and give it to other.

>> This will resolve the deadlock situation, but sometimes it does causes problems.

#### 2) Rollback -

>> In situations where deadlock is a real possibility, the system can periodically make a record of the state of each process and when deadlock occurs, roll everything back to the last checkpoint, and restart, but allocating resources differently so that deadlock does not occur.

#### 3) Kill one or more processes -

>> This is the simplest way, but it works.