DEADLOCK AND ITS CONDITIONS

Deadlock:

Resources

Resources are the passive entities needed by processes to do their work. A resource can be a hardware device (eg. a disk space) or a piece of information (a locked record in the database). Example of resources include CPU time, disk space, memory etc. There are two types of resources:
Preemptable – A Preemptable resources is one that can be taken away from the process owing it with no ill effect. Memory is an example of preemptable resources.

Non-preemptable – A non-preemptable resources in contrast is one that cannot be taken away from its current owner without causing the computation to fail. Examples are CD-recorder and Printers. If a process has begun to burn a CD-ROM, suddenly taking the CD recorder away from it and giving it to another process will result in a garbled CD. CD recorders are not preemptable at any arbitrary moment.

Read-only files are typically sharable. Printers are not sharable during time of printing .One of the major tasks of an operating system is to manage resources.
Deadlocks may occur when processes have been granted exclusive access to Resources. A resource may be a hardware device (eg. A tape drive) file or a piece of information (a locked record in a database). In general Deadlocks involves non preemptable resources. The sequence of events required to use a resource is:

1. Request the resource
2. Use the resource
3. Release the resource

Conditions for Deadlock:

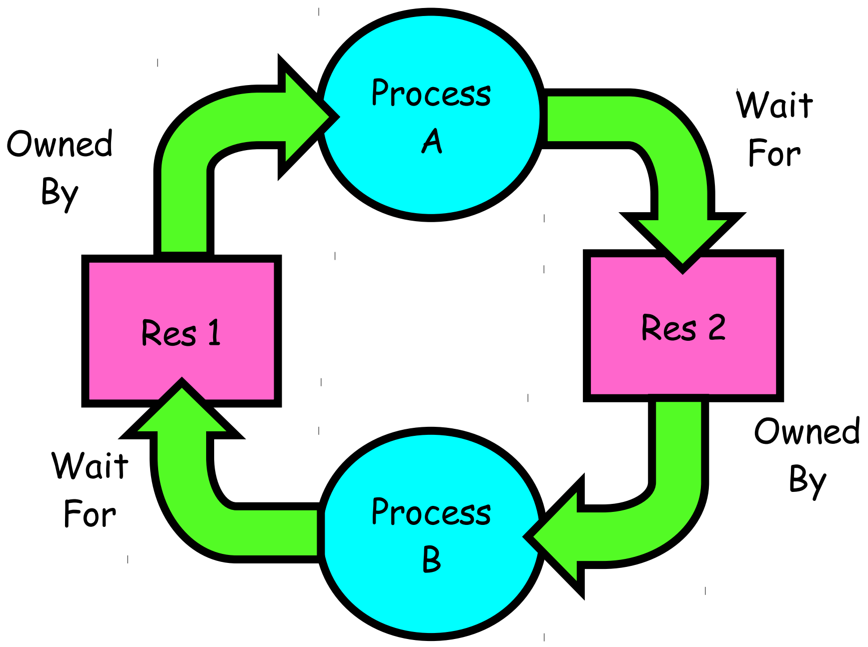
Coffman et al. (1971) showed that four conditions must hold for there to be a deadlock:

1. Mutual exclusion
   Only one process at a time can use a resource.
2. Hold and wait
   Process holding at least one resource is waiting to acquire additional resources held by other processes.
3. No preemption
   Resources are released only voluntarily by the process holding the resource, after the process is finished with it
4. Circular wait
   There exists a set {P1 , …, Pn } of waiting processes.
   P1 is waiting for a resource that is held by P2
   P2 is waiting for a resource that is held by P3
   …
   Pn is waiting for a resource that is held by P1

All of these four conditions must be present for a deadlock to occur. If one or more of these conditions is absent, no Deadlock is possible.

What is Deadlock?

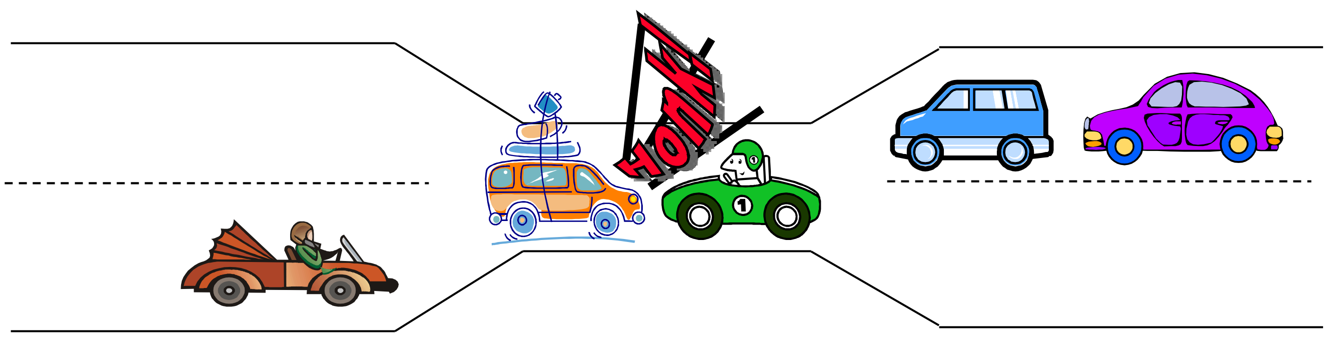
In Computer Science a set of process is said to be in deadlock if each process in the set is waiting for an event that only another process in the set can cause. Since all the processes are waiting, none of them will ever cause any of the events that would wake up any of the other members of the set & all the processes continue to wait forever.



Example 1:

- Two process A and B each want to recored a scanned document on a CD.
- A requests permission to use Scanner and is granted.
- B is programmed differently and requests the CD recorder first and is also granted.
- Now, A ask for the CD recorder, but the request is denied until B releases it. Unfortunately, instead of releasing the CD recorder B asks for Scanner. At this point both processes are blocked and will remain so forever. This situation is called Deadlock.

Example:2
Bridge Crossing Example:



Each segment of road can be viewed as a resource

- Car must own the segment under them
- Must acquire segment that they are moving into

For bridge: must acquire both halves

- Traffic only in one direction at a time
- Problem occurs when two cars in opposite directions on bridge: each acquires one segment and needs next

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

Starvation is possible

- East-going traffic really fast ==>no one goes west

Starvation vs. Deadlock

Starvation: thread waits indefinitely

Example, low-priority thread waiting for resources constantly in use by high-priority threads

Deadlock: circular waiting for resources

Thread A owns Res 1 and is waiting for Res 2 Thread B owns Res 2 and is waiting for Res 1

Deadlock ==> Starvation but not vice versa

Starvation can end (but doesn't have to)

Deadlock can't end without external intervention

Source : http://dayaramb.files.wordpress.com/2012/02/operating-system-pu.pdf