System Programming Lab

2021315385

이건 / Gun Daniel Lee

Report 14

This report summarizes the definition, condition, and prevention of a deadlock.

Definition:

A deadlock problem is caused when a set of processes or threads is blocked caused by processes holding a resource while waiting for another resource which is being held by other processes.

Conditions:

A deadlock can occur only if all four Coffman conditions hold true simultaneously. These four conditions are mutual exclusion, hold and wait, no preemption, and circular wait.

1. Mutual exclusion: Only one process at a time can access a resource.
2. Hold and wait: A process holding at least one resource is waiting to acquire additional resources held by other processes.
3. No preemption: Resource can be released only voluntarily by the process holding it, after that process has completed its task.
4. Circular wait: There must exist a set of waiting processes such that a process is waiting for a resource held by the next process in the set, creating a circular waiting cycle.

Prevention:

There are several preventive measures to prevent each of the four Coffman conditions from occurring.

1. Mutual exclusion prevention: A prevention is not required for sharable resources, but it must hold for non-sharable resources.
2. Hold and wait prevention: The program must guarantee that whenever a process requests resources, it is not holding any other resources.
3. No preemption prevention: If a process must wait for another resource, all resources currently held are implicitly preempted. Additionally, if requesting resources are allocated to some other process that is waiting for additional resources, then the program should preempt the desired resources from the waiting process.
4. Circular wait: The program must impose total ordering of all resource types. The

program must also require that each process requests resources in an increasing order of enumeration.

In summary, when the four Coffman conditions, mutual exclusion, hold and wait, no preemption, and circular wait, are held true simultaneously, this causes several processes to freeze, unable to move on. Therefore, a program must follow preventive measures for each condition to avoid deadlocks from occurring.