|  |  |  |
| --- | --- | --- |
| Features | Deadlock | Starvation |
| Definition | Deadlock happens when every process holds a resource and waits for another process to hold another resource. | Starvation happens when a low priority program requests a system resource but cannot run because a higher priority program has been employing that resource for a long time. |
| Basic | A deadlock occurs when no process can proceed and becomes blocked. | Starvation occurs when low priority procedures are blocked while high priority operations proceed. |
| Other names | Deadlock is also known as circular wait. | Starvation is known as a Lived lock. |
| Resources | Other processes block requested resources while a process is deadlocked. | High-priority processes continue to use the requested resources. |
| Arising Condition | Mutual exclusion's occurrence, Hold and wait, No preemption, and Circular wait all happen simultaneously. | Uncontrolled resource management, enforcement of priorities. |
| Prevention | It can be prevented by avoiding the situations that lead to deadlock. | Aging may prevent it. |

Conclusion

Multiple processes are executing in an operating system. Both starvation and deadlock are two methods that may happen when there are processes. Deadlock happens when every process holds a resource and waits to obtain a resource held by another process. In contrast, starvation happens when a process waits indefinitely for a required resource. Deadlock may cause processes to starvation, and on the other side, starvation can break the deadlock.