Roll no :- T21287

Page No.
Date

Assignment No : 5

AIM : Implement the C program for Deadlock Avoidance Algorithm Bankers Algorithm

OBJECTIVES : To study

- Deadlock & starvation
* Condition for deadlock & it's handling
* Banker Algorithm

THEORY :

What is Deadlock ?

A set of processes is deadlock if each process in the set is waiting for an event that only onther process in the set can cause. Because all the processer are waiting none of them will ever cause any of the events that could wake up any of the other members of the sets & all the processes continue to wait forever

Conditions for Deadlock

1. Mutual exclusion Condition Each resources is either currently assigned to exactly one process or is available

2. Hold & wait condition process currently holding resources granted earline can request new resources.

3. No preemption condition Because. Resources

previously granted eannot be forcibly taken away from a process. They must be explicity relesed by the process holding them

4. Circular wait condition

Consider a computer system that runs 5000 jobs per month with no deadlock-prevention or deadlock occurs about twice per month, & the operator must terminate & return about 10 jobs per deadlock. Each job is worth about $2 (in cpu time) & the jobs terminated tend to the about half-done when they are aborted

A systems programmer has estimated that a deadlock-avoidance algorithm could be installed in the system with an increase in the average execution time per job of about 10 percent. Since the machine currently has 30 percent idle time, all 5000 jobs per month could still be run, although turnaround time would increase by about 20 percent on avereage

Conclusion :

Thus we have implemented dining philosophers problem using 'C' in Linux.