Tutorial 7, CSN – 232

(**Deadline:** March 21, 2018)

Question 1: Consider a computer system that runs 5,000 jobs per month with no deadlock-prevention or deadlock-avoidance scheme. Deadlocks occur about twice per month, and the operator must terminate and rerun about 10 jobs per deadlock. Each job is worth about $2 (in CPU time), and the jobs terminated tend to be about half-done when they are aborted. A systems programmer has estimated that a deadlock-avoidance algorithm (like the banker's 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 5,000 jobs per month could still be run, although turnaround time would increase by about 20 percent on average.

i. What are the arguments for installing the deadlock-avoidance algorithm?
ii. What are the arguments against installing the deadlock-avoidance algorithm?

Question 2: Suppose that a system is in an unsafe state. Show that it is possible for the processes to complete their execution without entering a deadlock state.

Question 3:

i. Starvation can be defined as the situation whereby a process must wait beyond a reasonable period of time, perhaps indefinitely, before receiving a requested resource. Can a system detect that some of its processes are starving? If you answer "yes," explain how it can. If you answer "no," explain how the system can deal with the starvation problem.

ii. Is it possible to have a deadlock involving only one single-threaded process? Explain your answer.