

CS 201 Operating Systems Test 2

Instructor: Dr. Alan Ling

Date: November 11, 2003

Time: 3:20pm to 5:00pm

NAME:

This test has 100 marks to be earned, and 100 minutes in which to earn them. Budget your time accordingly! The examination paper has six questions on five pieces of paper. Attempt all questions. Write all answers on this test paper in the test provided, using backs of sheets for rough work or extra space. Texts, notes, and other aids are not permitted.

Problem 1: (10 points) What is the critical-section problem? What are the three requirements that a solution to the critical-section problem must satisfy?

Problem 2: (20 points) Give a solution to the critical-section problem for the multiple process without using semaphore. Include a prove of validity of your algorithm. Indicate clearly the entry section and exit section of your code.

Problem 3: (20 points) Consider a government with three *party* processes (Progressive Conservative, Liberals and New Democratic Party) and one *agent* process (USA). Each party continuously put a policy onto the floor and then votes on it. But to put a policy onto the floor and then vote it, the parties need three ingredients: public support, financial support and media support. One party (Progressive Conservative) has financial support; one party (Liberals) has public support; one party (New Democratic Party) has media support. The agent (USA) can generate infinite supply of all three type of supports. The agent (USA) places two of the supports out. The party who has the remaining supports then put a policy onto the floor and votes on it, signaling the agent on completion. The agent then puts out another two

of the three supports, and the cycle repeats. Write a program to synchronize the agent and the parties.

Problem 4: (10 points) Describe the necessary conditions for the existence of a deadlock situation.

Problem 5: (20 points) State and define the banker's algorithm (including the data structures, safety algorithm, and resource-request algorithm).

Problem 6: (20 points) Consider a system consisting of m resources of the same type, being shared by n processes. Resources can be requested and released by processes only one at a time. Show that the system is deadlock-free if the following two conditions hold:

- a. The maximum need of each process is between 1 and m resources.
- b. The sum of all maximum needs is less than $m + n$.