CS 162

MIDTERM 1

FALL 1994

Prof. Alan Jay Smith

You have until the end of class for this exam. The exam is closed book. All answers should be written on the exam paper. Anything that we can't read or understand won't get credit. Any question for which you give no answer at all will receive 25% partial credit. Please answer in standard English; illiterate or illegible answers to essay questions will lose credit. Please watch the front board for corrections and other informations. This exam has 6 questions on 7 pages and is in two parts.

Problem #1

In a discrete event simulation (such as you did for assignment #1), explain how you compute (accurately) the mean number of customers in the system. Show a mathematical formula if that is necessary. (10)

Problem #2

Define/explain the "exponential queue" scheduling algorithm. Explain the reasoning behind it - i.e. why should it work well? (12)

Problem #3

Why is minimizing the variance of the flow time (var(f(i)) not the same as minimizing the variance of (f(i)/s(i))? What scheduling algorithm does the former? Which is preferable? Why? (13)

Problem #4

What is the difference between a process and a thread? Define each. (10)

Problem #5

List and explain the four requirements for deadlock. (12)

Problem #6

Show how the SWAP instruction (as defined in class) can be used to implement P() and V(). (This should be at the same level of detail as was used in class.) Why do we permit busy waiting in the implementation? (13)

Problem #7

All the cs162 students are working very hard on their programming projects. Unfortunately, there are only N workstations they can use. When a student arrives at the lab, he/she sits down at an open workstation and starts working. If a student arrives at the lab and all the workstations are occupied, he/she waits, unless there are more than 10 students already waiting, in which case s/he leaves. When a student finishes his/her work, he/she will leave and if one or more students are waiting, one will sit down at the workstation. Write code to synchronize the student use of the workstations. You can use the variables & semaphores given below. Feel free to define and use additional variables and semaphores if you need them. (30)

binary semaphore Mutex = 1
counting semaphore Terminals = N
int NumSleeping = 0