MIDTERM TEST -- CS 177

University of California, Riverside.

Spring 1995

All aids allowed.

Notes to students:

a. State all assumptions where appropriate.
b. In general, show your intermediate work.

1. **[10 marks]**

The *poker test* is a procedure (in conjunction with the chi-square test) for testing independence of random numbers, based on the frequency with which certain digits are repeated in a series of numbers. Consider a series of 3-digit U(0,1) random numbers (e.g., 0.375, 0.688, ...). In this case, there are 3 possibilities for the way digits can repeat within each random number (see below for an example) and there are theoretical probabilities associated with each possibility.

a) What is the theoretical probability that the three-digit ‘‘poker hand’’ contains three different digits, exactly one pair of like digits, and three like digits? (Beware that these digits are independently sampled, unlike a real poker hand drawn from a deck of playing cards!)

b) Suppose we generate a sequence of 1000 3-digit U(0,1) random numbers from the target random number generator, and find that 680 have three different digits, 289 contain exactly one pair of like digits, and 31 contain three like digits. Let $\alpha = 0.05$ for the chi-square test. Based on the poker test, are these numbers independent?

2. **[10 marks]**

Let X be a random variable with distribution function $F_X(x)$ and let Y be an independent random variable with distribution function $F_Y(y)$.

a) Briefly explain the meaning of the distribution function $F_X(x)$, and its relation to the density function $f_X(x)$.

b) What is the distribution function of $min\,(X, Y)$?

c) What is the distribution function of $max\,(X, Y)$?

3. **[10 marks]**

In the *theory()* routine of the CSIM version of the $M/M/1$ queueing system, shown below, 4 quantities are calculated as follows:

throughput = 1 / (mean interarrival time)
utilization = throughput * (mean service time)
mean number in system = utilization / (1 - utilization)
mean response time = (mean service time) / (1 - utilization)

Note that both mean interarrival time and mean service time are available as defined constants. Not surprisingly, Little’s Law is at work here, in several ways.

By assuming that 2 out of 4 of the left-hand-side performance measures are available (you choose the 2) as well as the defined constants, show how Little’s Law can be used to calculate the other 2 left-hand-side measures.

4. **[10 marks]**

Consider the CSIM program for modelling an $M/M/1$ queue shown below. Modify the program to handle a system with *two* servers, one after the other, instead of just one. As before, new customers enter the system from a Poisson source and then wait in line for the first server. However when they have finished their first service they go on to the second server instead of leaving the system. Service times at the second server are exponential with the same mean as the first server, but they are independently resampled.

We will consider two cases. In case I, we assume that there is separate waiting area for customers waiting for the second server so there is no connection between the two queues except the flow of customers from queue one into queue two. In case II, we assume that there is no waiting area for the second server. Thus, if a customer completes service at the first server, but the second server is still busy with the previous customer, the customer is *blocked* at the first server, unable to leave but no longer receiving any service.

Briefly indicate what needs to be changed in the CSIM program below in order to model these two cases. Do not worry about performance measures. Worry about new data declarations and new control structures, in CSIM. Answers with "Replace line ... with", "Delete line ...", and "Insert after line ..." are fine from a style viewpoint.

5. **[10 marks]**

a) Briefly explain what the likelihood function is, and how it is used to determine the maximum likelihood estimate (MLE) for some parameter of a random variable.

b) On page 349 of the book, it says that the MLE for the Poisson distribution with parameter λ is:

$$\hat{\lambda} = \bar{X}(n)$$

Prove it.

```
/* simulate an M/M/1 queue
  (an open queue with exponential service times and interarrival intervals)
*/

#include "lib/csim.h"

#define   SVTM1.0/*mean of service time distribution */
#define   IATM2.0/*mean of inter-arrival time distribution */
#define   NARS5000/*number of arrivals to be simulated*/

FACILITY f;/*pointer for facility */
EVENT done;/*pointer for counter */
TABLE tbl;/*pointer for table */
QTABLE qtbl;/*pointer for qhistogram */
int cnt;   /*number of active tasks*/

sim()     /*1st process - named sim */
{
          int i;

          set_model_name("M/M/1 Queue");
          create("sim");/*required create statement*/
```

```
	f = facility("facility");/*declare facility*/
	done = event("done");/*declare event*/
	tbl = table("resp tms");/*declare table */
	qtbl = qhistogram("num in sys", 10);/*declare qhistogram*/

	cnt = NARS;/*initialize cnt*/
	for(i = 1; i <= NARS; i++) {
	hold(expntl(IATM));/* hold interarrival*/
	cust();/*initiate process cust*/
	}
	wait(done);/*wait until all done*/
	report();/*print report*/
	theory();/*print theoretical res*/
	mdlstat();
}

cust()	/*process customer*/
{
	float t1;

	create("cust");/*required create statement*/

	t1 = clock;/*time of request */
	note_entry(qtbl);/*note arrival */
	reserve(f);/*reserve facility f*/
	hold(expntl(SVTM));/*hold service time*/
	release(f);/*release facility f*/
	record(clock-t1, tbl);/*record response time*/
	note_exit(qtbl);/*note departure */
	cnt--;/*decrement cnt*/
	if(cnt == 0)
	set(done);/*if last arrival, signal*/
}

theory() /*print theoretical results*/
{
	float rho, nbar, rtime, tput;

	printf("\n\n\n\t\t\tM/M/1 Theoretical Results\n");

	tput = 1.0/IATM;
	rho = tput*SVTM;
	nbar = rho/(1.0 - rho);
	rtime = SVTM/(1.0 - rho);

	printf("\n\n");
	printf("\t\tInter-arrival time = %10.3f\n",IATM);
	printf("\t\tService time       = %10.3f\n",SVTM);
	printf("\t\tUtilization        = %10.3f\n",rho);
```

```
    printf("\t\tThroughput rate    = %10.3f\n",tput);
    printf("\t\tMn nbr at queue    = %10.3f\n",nbar);
    printf("\t\tMn queue length    = %10.3f\n",nbar-rho);
    printf("\t\tResponse time      = %10.3f\n",rtime);
    printf("\t\tTime in queue      = %10.3f\n",rtime - SVTM);
}
```