UNIVERSITY OF LONDON
IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 1996

BEng Honours Degree in Computing Part II
MEng Honours Degrees in Computing Part II
for Internal Students of the Imperial College of Science, Technology and Medicine

This paper is also taken for the relevant examinations for the Associateship of the City and Guilds of London Institute

PAPER 2.9

SIMULATION AND MODELLING
Thursday, May 16th 1996, 10.00 - 11.30

Answer THREE questions

For admin. only: paper contains
4 questions
4 pages (excluding cover page)

1 In a *priority* queueing system customers are allocated a priority on arrival, which is a number between 1 and n inclusive, the highest priority being n. There is a single server and after a service completion the customer with the highest priority is allocated the server next. If more than one such customer exists the one who arrived first is selected. If the server is idle on arrival the arriving customer goes straight into service. A customer in service is *not* interrupted by the arrival of a higher priority customer.

For the system in question the arrival process is Poisson with rate r and the service times are exponentially distributed with mean $S(p)$ for priority p, $1 \le p \le n$.

A simulator for this system models the stream of arriving customers by the following event procedure:

```
PROCEDURE Arrival() ;
  ProcessArrival( GetPri() ) ;
  Schedule( Arrival, Now() + ExpSample( 1/r ) )
END Arrival ;
```

where `GetPri()` returns a sample priority and `ExpSample( x )` returns a sample from an exponential distribution with mean `x`.

a Identify the state variables required by the simulator and write down pseudocode for the procedure `ProcessArrival` used above and the event procedure `Departure` for modelling service completion events. You may assume the existence of an object `Server` with access procedures `ClaimServer` and `ReleaseServer`.

b The probability that an arriving customer is of priority i is given by $p(i)$, $1 \le i \le n$, so that

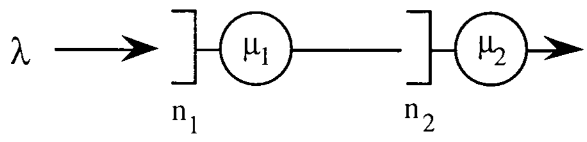
$$\sum_{i=1}^{n} p(i) = 1$$

Using this probability distribution define the procedure `GetPri` used above for allocating customer priorities.

You may use any program notation you wish so long as your solution is clear and readable. You need not define any support procedures you use provided you use suitably obvious names. Avoid all unnecessary detail.

Note: 70% of the marks are allocated to part a.

2 Consider the following tandem queueing network:



The arrival process is Poisson with rate λ and the service time distributions are both exponential with respective rates μ_1 and μ_2.

Let the state of the system be the pair (n_1, n_2) where n_1 is the population of the first queue and n_2 the population of the second, and let P_{n_1,n_2} be the probability that the system is in the state (n_1, n_2), $n_1,n_2 \geq 0$. For convenience, write $\rho_1=\lambda/\mu_1$ and $\rho_2=\lambda/\mu_2$.

2a Draw the Markov diagram showing the states (i,j), $i+j \leq 2$, and the associated transition rates between them.

2b Using the diagram, or any other means, write down the equilibrium balance equations for the system.

2c Show that the general solution $P_{n_1,n_2} = \rho_1^{n_1}(1-\rho_1)\, \rho_2^{n_2}(1-\rho_2)$, $n_1,n_2 \geq 0$, is a solution to your balance equations.

2d From the general solution given in part 2c show that equilibrium probability distribution for the number of customers at the second queue, Q_n $n\geq 0$ say, is given by

$$Q_n = \rho_2^{n}(1-\rho_2)$$

Turn over

3a A random number generator has been written and has been called *n* times to produce the samples $X_1, ..., X_n$, $0 \le X_i \le 1$, $1 \le i \le n$, $n>0$. To assess the quality of the generator, it is necessary to test for independence of the generated samples.

The *serial test* is a technique for testing the independence of pairs of samples in the sequence; it works as follows. Adjacent samples are paired up so that sample X_i is paired with sample X_{i+1}, $1 \le i \le n-1$. The resulting pairs are then used to update a *two-dimensional* histogram consisting of b^2 buckets, $B_{u,v}$, $1 \le u,v \le b$. For the pair (X_i, X_{i+1}) the index u is given by $\lfloor b X_i \rfloor + 1$ and v by $\lfloor b X_{i+1} \rfloor + 1$; the contents of bucket $B_{u,v}$ is then incremented. The notation $\lfloor r \rfloor$ denotes the largest integer less than or equal to r.

If the samples are independent the expected contents of $B_{u,v}$ after all pairs have been processed is $(n-1)/b^2$ for all $1 \le u,v \le b$. The χ^2 test can therefore be used to compare the observed and expected bucket contents.

Apply the serial test to the following data at the 10% significance level, assuming b=2. What conclusions can you draw from the test?

```
.560 .113 .889 .053 .573 .708 .340 .421 .713 .236 .247 .108
.683 .865 .298 .755 .086 .474 .313 .370 .644 .279 .137 .058
.353 .344 .136 .472 .860 .665 .499 .753 .282 .656 .644 .711
.726 .703 .926 .977 .775 .610 .940 .896 .158 .635 .648 .911
.892 .893 .577 .177 .118 .863 .359 .146
```

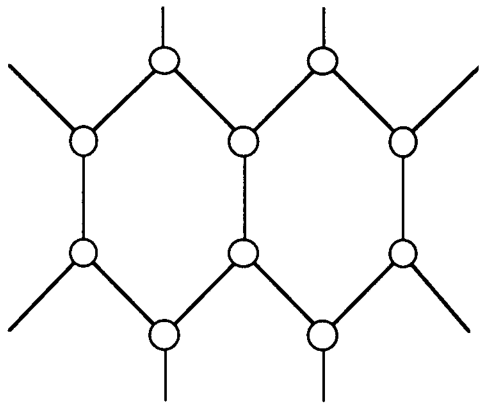
3b The *Weibull* distribution is often used to model "time to failure" for machinery or electronic components. Its cumulative distribution function (cdf) is given by

$$F(x) = 1 - \exp(-(x/\alpha)^{\beta})$$

Show how the *transform* method can be used to generate samples from the Weibull distribution with parameters $\alpha = 1$ and $\beta = 2$. As part of your answer you should state any mathematical properties which the method assumes, although you are not required to prove them.

Note: 60% of the marks are allocated to part a.

4 A new packet-switched telecommunications network has been proposed which consists of a set of transmitter/receiver nodes, each with three I/O ports, interconnected in a mesh:



At certain times, a node (the *transmitter)* will initiate a communication with another node (the *receiver*). This takes the form of a variable-sized packet which is routed to the receiving node in accordance with a routing algorithm. The routing algorithm tells the transmitter on which of its three output links it should send the packet first. When a packet arrives on an input link the algorithm determines whether the packet is destined for that node (i.e. if it is the receiver of the packet) and, if it is not, which output link to forward it to in order to reach the destination. Each output link has a FIFO (first-in-first-out) buffer containing any outstanding packets waiting to use the link.

A stochastic discrete-event simulator has been written to model this system for experimental purposes, but no measurement code has yet been incorporated. You are asked to advise the simulation designer on various aspects of the measurement process.

a Explain *briefly* how the designer might incorporate code to estimate the following quantities (do *not* attempt to detail the required code):

i. The mean total packet transmission time from transmitter to receiver

ii. The mean number of packets queued at a given output buffer

iii. The distribution of the number of packets received at a given node per unit time

b Having incorporated the measurement code the simulator is executed eight times, with each run being based on a different random number sequence. The mean packet transmission times in milliseconds recorded on each run are as follows:

26.2 29.4 25.5 27.9 24.8 26.1 28.0 25.1

Compute the 90% confidence interval for the mean transmission time.

Note: 70% of the marks are allocated to part a

End of paper

The student's *t* distribution

The values t_υ in this table are such that $F(t_\upsilon) = P$ where F is the student's t cumulative distribution. Thus the P values represent fractile points for the student's t distribution having υ degrees of freedom. For two-sided confidence interval calculations, with confidence level $1 - \alpha$ and sample size n, $P = 1 - \alpha/2$ and $\upsilon = n - 1$. For a one-sided confidence interval calculation, with confidence level $1 - \alpha$ and sample size n, $P = 1 - \alpha$ and $\upsilon = n - 1$.

	P				
υ	0.75	.90	.95	.975	.99
3	.765	1.638	2.353	3.182	4.541
4	.741	1.533	2.132	2.776	3.747
5	.727	1.476	2.015	2.571	3.365
6	.718	1.440	1.943	2.447	3.143
7	.711	1.415	1.895	2.365	2.998
8	.706	1.397	1.860	2.306	2.896
9	.703	1.383	1.833	2.262	2.821
10	.700	1.372	1.812	2.228	2.764
15	.691	1.341	1.753	2.131	2.602
20	.687	1.325	1.725	2.086	2.528
30	.683	1.310	1.697	2.042	2.457
60	.679	1.296	1.671	2.000	2.390
∞	.674	1.282	1.645	1.960	2.326

The chi-square distribution

The values u in this table are such that $F(u) = P$ where F is the chi-square cumulative distribution with k degrees of freedom. Thus, for example, a random variable having the chi-square distribution with 5 degrees of freedom has a 10 percent probability of being greater than 9.24, a 10 percent probability of being less than 1.61, and an 80 percent probability of having a value in the range from 1.61 to 9.24.

	P					
k	.10	.25	.50	.75	.90	.99
3	.584	1.21	2.37	4.11	6.25	11.3
4	1.06	1.92	3.36	5.39	7.78	13.3
5	1.61	2.67	4.35	6.63	9.24	15.1
6	2.20	3.45	5.35	7.84	10.6	16.8
7	2.83	4.25	6.35	9.04	12.0	18.5
8	3.49	5.07	7.34	10.2	13.4	20.1
9	4.17	5.90	8.34	11.4	14.7	21.7
10	4.87	6.74	9.34	12.5	16.0	23.2
12	6.30	8.44	11.3	14.8	18.5	26.2
14	7.79	10.2	13.3	17.1	21.1	29.1
16	9.31	11.9	15.3	19.4	23.5	32.0
18	10.9	13.7	17.3	21.6	26.0	34.8
20	12.4	15.5	19.3	23.8	28.4	37.6
30	20.6	24.5	29.3	34.8	40.3	50.9
40	28.9	33.7	39.3	45.6	51.8	63.7
50	37.6	43.0	49.3	56.3	63.2	76.2