UNIVERSITY OF HERTFORDSHIRE

Session 1999/2000

Semester A/B

Faculty of Engineering and Information Sciences
Department of Computer Science

CS2/CSP3

Module Code: 2COM0003

Systems and Networks 2

DURATION OF EXAM: 3 hours

THE FOLLOWING IS PROVIDED FOR THIS EXAMINATION:

ONE answer book

INSTRUCTIONS TO CANDIDATES:

Answer FOUR questions

All questions carry equal marks

This paper consists of SIX questions on FIVE pages.

1. Throughout this question you should state carefully any assumptions upon which your answers rely.

   (a) State and prove a relationship between the average values of the following quantities: the number N of interactive system users, the interactive system throughput X, the interactive response time R of the system, and the think time Z of an interactive user.

   (5 marks)

   (b) A certain server acts as a query engine for both development (prototype) databases and production (fielded) databases.

   Each development query requires on average 18 seconds of cpu time and 6 seconds of access to the I/O channel. For production queries the corresponding figures are 30 seconds and 42 seconds respectively. Both the cpu and the I/O channel show an average utilization of 80%.

   (i) Show that the average query completion rate is the same for both types of query.
   (ii) What is the throughput for each type of query?

   (4 marks)

   (c) The mean server response time for a production query (time from submission to completion) is five minutes. During a typical shift, there were 70 developers building prototypes. On average, 15 queries (development plus production) were resident (queued or being processed) in the server at any time.

   (i) What was the average server response time for a development query during the shift?

   (ii) How long was the mean think time of a developer? (The think time Z is the time between receiving the results of a query and submitting the next query).

   (4 marks)

   (d) Following defragmentation of the disk, the I/O channel utilization rises to 81.5%, although the throughput of production queries is unchanged.

   Explain carefully what you believe is happening, and give your reasons. Calculate the likely effect on query response time observed by developers. What was constraining the system?

   (6 marks)

   (e) Following this, it is desired to increase the throughput of production queries to 72 queries per hour. Is an additional I/O channel required? What other recommendations would you make?

   (6 marks)

2. (a) Suppose $G = < N, T, S, P >$ to be a context-free grammar.

(i) Explain carefully the meaning of the term 'derivation tree'.

(3 marks)

(ii) Define the language $L = L(G)$ generated by the grammar.

(3 marks)

(iii) What does it mean to say that G is ambiguous?

(2 marks)

(b) Suppose it is desired to add arithmetic expressions of the conditional form

$$< ae > \longrightarrow \texttt{if} < be > \texttt{then} < ae > \texttt{else} < ae > \texttt{endif}$$

to a programming language generated by a grammar which already has non-terminal symbols $< ae >$ and $< be >$ denoting arithmetic and boolean expressions respectively.

(i) Using as example a statement such as

```
x := if x>3 then y else x+1 endif;
```

explain carefully how you would go about modifying the parser so as to produce derivation trees for programs which used the new construct. (You can make any reasonable assumptions you like about how the parser you are modifying works at the moment, but you must state explicitly what assumptions you are making.)

(8 marks)

(ii) Could adding the new construct make the grammar ambiguous if it wasn't before? Give your reasons.

(3 marks)

(iii) What additional semantic information would need to be specified for the new construct in order to generate code unambiguously? Justify your answer.

(2 marks)

(c) Would adding to the original grammar the construct

$$< ae > \longrightarrow \texttt{if} < be > \texttt{then} < ae > \texttt{else} < ae >$$

be equivalent, or might omitting the '`endif`' cause problems? Explain your reasoning.

(4 marks)

3. (a) Explain what it means to say that programs show *locality* of memory reference.

(4 marks)

(b) Assume a system with a virtual memory of 16 pages and a physical memory of 8 page frames; the system uses *demand* paging. The page replacement algorithm attempts to keep one page frame free and will initiate paging out after the free frame is filled. The page that is evicted from memory will be the one with the lowest numeric usage count, if more then one have the same usage count it will the first (lowest numbered) of them. The leftmost usage bit is set whenever the page is referenced (and also when a page is loaded), at fixed intervals all usage bits are right shifted one bit position and a zero bit is introduced at the leftmost position.

Given the following state of the page table, usage bits and free page frame:

page no.	frame no.	usage bits			
0	3	0	0	0	1
1	4	0	1	1	1
2	-				
3	-				
4	1	1	0	1	0
5	0	0	1	1	1
6	-				
7	-				
8	2	0	0	1	1
9	6	1	0	1	0
10	-				
11	-				
12	7	0	0	0	0
13	-				
14	-				
15	-				

and the free page frame = 5

And the following sequence of page references (page numbers) generated by a program:

0, 2, 5, 6,

Show the state of the page table and the free page frame at the end of the page reference sequence. More marks will be given for the correct pages in memory and the free frame than for the usage bits.

(9 marks)

(c) What is the usual name of the paging out algorithm in the previous part of the question (b), and why does it normally remove pages that are least likely to be needed.

(4 marks)

(d) In paging systems, what is the *working set*? How might the working set be used in scheduling? And what might happen if it is not used?

(8 marks)

4. (a) Define what a *process* is, and define what a *thread* is on a Unix system. What is the relationship between them?

(5 marks)

(b) What system calls does Unix provide to support the creation of processes and execution by them of new programs? Include in your answer (i) a description of the system calls, and (ii) how they are used to run a new program, as a new process.

(7 marks)

(c) Use a simple example to help explain how two concurrent threads, a *producer* and a *consumer* using shared memory to exchange composite data values could fail because the consumer will get an inconsistent composite value.

(5 marks)

(d) What is a *semaphore*? Explain how it can be used to overcome the difficulty of shared data access described in the previous part of this question.

(8 marks)

5. (a) Describe the BSD socket interface that the TCP/IP transport service (that's the part that uses TCP) presents to application programs. Include in your answer what the calls are, their parameters and how should they be used by the client and the server.

(8 marks)

(b) What service can the TCP transport layer expect from the IP network layer?

(3 marks)

(c) Describe how the TCP transport layer provides:

(i) process to process connections, (2 marks)

(ii) streams of characters, (2 marks)

(iii) reliable transmission, first define what *reliable* means and then say how it is achieved, (9 marks)

(iv) some form of flow-control for the receiver. (1 marks)

6. (a) Describe the HTTP/1.0 protocol used to communicated between WWW clients and servers.

(11 marks)

(b) If an HTTP Web server receives and identifies a request to run a CGI program it will send part of the HTTP protocol response sequence of lines to the client and leave the CGI program to send part of the sequence. Which lines are sent by the server and what is done by the CGI program? And why is this necessary?

(4 marks)

(c) How are input arguments sent to a server with a `GET` request? And how are they made available to the CGI program?

(4 marks)

(d) Briefly summarise how the DNS translates an internet domain name into an IP address.

(6 marks)