TE WHARE WĀNANGA O TE ŪPOKO O TE IKA A MĀUI

VICTORIA
UNIVERSITY OF WELLINGTON

EXAMINATIONS — 2009
MID-YEAR

COMP305

OPERATING SYSTEMS

Time allowed: THREE HOURS

Instructions: The examination contains 5 questions, you must answer all questions

Each question is worth 20 marks.

The exam consists of 100 marks in total.

Paper foreign to English language dictionaries are allowed.

Electronic dictionaries and programmable calculators are not allowed.

Question 1 Processes and Threads **[20 marks]**

(a) What is meant by the terms:

i. [1 Mark] batch processing,
Programs with the same runtime context are run sequentially

ii. [1 Mark] multi-programmed, and
Programs with the same runtime context are in memory simultaneously.

iii. [1 Mark] time sharing.
CPU is sliced, giving each program effectively a 1/n sized computer.

(b) [2 Marks] Outline the kernal state typically associated with a process.
PCB: FT, stacks, registers, pagetable.

(c) [2 Marks] Outline the kernal state typically associated with a kernal thread.
Kernal stack and registers.

(d) [4 Marks] Draw a diagram outling the lifecycle of a process.
Lecture 3, slide 22.

(e) [3 Marks] How can processes on the same machine communicate? Why is this hard?
Need some form of IPC supported by the OS. Shared memory is a possiblity, as is a file based scheme, message passing, or RPC. It is hard because memory protection is required to stop processes on a time sharing machine from interfereing with each other – due to bugs or malicisouness (memory protection schemes).

(f) [6 Marks] Consider the problem of writing a server application. The application has many simultaneous requests, and each request involves a single file access. The requests can be considered independent and the server will run on a dedicated machine. The system on which you are developing this application does not support kernal threads, you will only be able to use user-level threads and processes. Outline a brief design of your application, indicating how you will architect your sever to enable high service throughput.
They will have to use processes, the file system access will break user-level threads (unless they have a tricky and inventive solution).

Question 2 Scheduling **[20 marks]**

(a) [1 Mark] What is the convoy effect, and in which algorithm does it arise?
Small jobs stuck behind a long job – this is a FCFS problem.

(b) [2 Marks] Which type of system would suffer most from the convoy effect?
Interactive – convoy effect has a big impact on response time.

(c) When choosing a quantum for RR scheduling:

i. [1 Mark] Why is context switch time important when selecting the quanta?
Too long a quanta = FCFS, too short =overhead.

ii. [2 Marks] What effect can increasing the quanta have on the average turnaround time?
Average turnaround does not always improve with the length of the quanta.

PROCESS	ARRIVAL TIME	CPU BURST
1	0	1
2	3	2
3	3	6
4	5	1

Table 1: Process arrival and burst times.

(d) [3 Marks] For the processes in Table 1, draw a Gantt chart and compute the average turnaround time for SJF.
2 marks for Gantt chart, 1 for calculation.

(e) SJF is optimal, but we don't know the length of the next CPU burst. One way to 'estimate' the next burst is to use prior bursts to compute the next expected burst length. A common way to do this is to use the decaying weighted average.

i. [2 Marks] What is the formula for the decaying weighted average?
$\tau_{n+1} = \alpha t_n + (1 - \alpha)\tau_n$

ii. [2 Marks] Compute the burst 'estimates for process $P_i = \{4, 7, 1, 5\}$, $\alpha = 0.6$ and $\tau_0 = 5$
5.00, 4.35, 6.07, 2.78

(f) In scheduling terms, what is meant by:

i. [2 Marks] Hard affinity, and
A thread may only run on a specific (or set) of processors. This is to optimize some hardware or meet some other constraint.

ii. [2 Marks] Soft affinity?
To make use of previously cached information, if possible, a thread will be rescheduled to the same processor on which it last ran.

(g) [3 Marks] Explain the term 'Priority Inversion'.
A higher priority task is waiting while a lower priority task is running, due to some other blocking effect. For example, slide 27, lecture 11.

Question 3 Memory Management **[20 marks]**

(a) [1 Mark] Explain the difference between internal and external memory fragmentation.
Internal is within the unit of allocation (left-over space on a page), whereas external fragmentation is left-over unallocated space outside the unit of allocation. (0.5 each)

(b) [3 Marks] State and characterise the three allocation methods used when trying to fit a memory request to a set of memory fragments in a free-list.
Best-fit (lots of small useless fragments), first-fit (high fragmentation at start of list, more useable left-overs), worstfit(leaves the biggest leftovers which are easier to allocate).

(c) [1 Mark] What must happen to a TLB on a context switch, and why?
Must be flushed, as its translations will not be valid for the new process (0.5 action, 0.5 reason).

(d) [4 Marks] What is virtual memory? Give 3 reasons why virtual memory is a good idea.
Virtual memory, logical space > physical space:
1) *Only part of the program needs to be in memory (error handling etc not usually needed)*
2) *Allows address spaces to be shared.*
3) *Allows for more efficient process creation.*

(e) [5 Marks] Show, with the aid of an annotated diagram, a demand paging system resolving a pagefault.
Lecture 14, slide 12. Must have valid bits (and maybe dirty).

(f) [6 Marks] Explain the 'second-chance' page-replacement algorithm.
Lecture 14, slide 22.

Question 4 File Systems **[20 marks]**

(a) [2 Marks] What is meant by an **Indexed** file allocation scheme?
Essentially inodes. Non contig.

(b) [2 Marks] What are two responsibilities of a filesystem?
Directory Service, Storage Service

(c) [2 Marks] System wide and per-process open file tables have different roles in the system. Show in a diagram how they coordinate to support file sharing.
Lecture 17, slide 8.

(d) [5 Marks] With the aid of a diagram, illustrate how indexed files are updated reliably – with the changes to the on-disk structure being committed in a single block write.
Lecture 17, slide 20.

(e) [1 Mark] What is disk scheduling seeking to minimise in order to maximise disk performance?
Seek time.

(f) Imagine you have a disk with 600 tracks with the head currently at position 231. For the following track sequence {517, 305, 43, 212, 113, 443, 390, 44, 464, 552}, compute the average head movement for:

i. [4 Marks] FCFS
ii. [4 Marks] C-Look

Question 5 Security **[20 marks]**

(a) [3 Marks] What is the domain model.

Lecture 20, page 6.

Access right is ability to execute a given operation on an object <object, access-set>

Domain contains a set of access rights relating to different objects.

(b) Authentication:

i. [2 Marks] What is the problem with an OS storing a file of passwords, when they are only readable by privileged users and OS utilities?

Either by accident or malicious intent the file may become available to ordinary users. Once this has happened, ALL passwords need to be reset.

ii. [2 Marks] The solution used in UNIX is to encrypt the passwords, and store the results in a publicly readable file. Explain how this is used and how it solves the problems noted in question (b).i.

The /bin/login program accesses the file and retrieves your encrpted password. The password you enter is then encrypted using the same process, crypt(), and the two encrypted values are compared. NOTE: Your encrypted password is NOT decrypted – think trapdoor.

(c) [5 Marks] Digital signatures and message digests are used to Identify the creator of a document and to ensure the accuracy of the document. This is commonly used in OS to verify that a patch, driver or program is indeed proper and correct before being applied to the OS. Explain how a message digest is used to verify a document.

Lecture 20, slides 18 and 19.

(d) Consider the following access control matrix. The matrix describes the access rights for a simple system containing three domains (Administrators, Programmers, and Guests), three files and an executable program.

	File A	*File B*	*Program P*
D1 (Admin)	*RW*	*R**	*Ex*
D2 (Programmer)	*RW*		*Ex*
D3 (Guest)	*R*	*W*	

i. [1 Mark] Which entities can *Read* File A?

Anyone

ii. [2 Marks] How could the programmer gain *Read* privilege for File B?

If given permission by Admin, i.e the transfer of the R right

iii. [2 marks] Specify the access control lists to implement this access control matrix.

File A - (D1,RW)(D2, RW)(D3,R)
File B – (D1, R)(D3,W)*
Program P – (D1, Ex)(D2, Ex)

(e) [3 Marks] List 3 of the 4 protection mechanisms from Java.

3 of {Class Loader, Verifier, JVM runtime tests, or Security manager}

* * * * * * * * * * * * * * * * * * * *