TE WHARE WĀNANGA O TE ŪPOKO O TE IKA A MĀUI
VICTORIA
UNIVERSITY OF WELLINGTON

EXAMINATIONS — 2005

MID-YEAR

COMP 305

Operating Systems

Time Allowed: 3 Hours

Instructions: Answer question one.

Answer **five** of questions two through seven.

Each question is worth 25 marks and should take you about 25 minutes.

Paper foreign language dictionaries are permitted.

Non-programmable calculators without full alphabetic keys are permitted.

Electronic dictionaries and programmable calculators are not permitted.

Question 1. Warm Up [25 marks]

A few short questions to get the mind working. Remember to allow approximately one minute for each mark.

(a) [4 marks] List four states that a process may be in.

(1) Ready (2) Running (3) Waiting for an event (4) Sleeping

OR new, ready, running, blocked, terminated

(b) [5 marks] Contrast **process context switching** with **thread context switching**.

Unlike a process, a thread shares code, data and files only need to switch registers and stack

(c) [3 marks] Why is a cycle in a resource allocation graph insufficient to show that a deadlock exists?

Can be deadlock free and a cycle if there are multiple instances of resources.

(d) [3 marks] List three approaches to structuring large page tables.

Hierarchial, hashed and inverted.

(e) [5 marks] Briefly describe three goals of directory systems.

Efficiency, locate a named file quickly. Provide convenient naming to users. Logically group files together

(f) [5 marks] Briefly discuss three potential drawbacks to biometric authentication.

Possible loss of attribute, can't change attribute easily if compromised, slow

Question 2. Synchronization and Deadlock [25 marks]

Operating systems have many concurrent activities that may interfere with each other.

(a) [3 marks] Explain why the non-synchronised version of the Nacho's Producer-Consumer program could result in the consumer thread attempting to remove an item from an empty buffer.

> *There is no synchronisation so consumer can remove an item from the buffer before producer adds an item to the buffer. Consumer can run first because of random yielding or IO activity.*

(b) [6 marks] Identify the three main sections of a protocol for solving the critical section problem and explain the purpose of each section.

> *Three marks given for identifying entry section, critical section and exit section. Entry section – control access to the critical section. Critical section - where access to shared data takes place. Exit section - signals exit to any waiting processes.*

(c) [4 marks] The **testAndSet** instruction can be thought of as a function implemented atomically in hardware. Here is a C++ style pseudocode definition of the function:

```
bool testAndSet(bool &lock) {
  bool tmp = lock;
  lock = true;
  return tmp;
}
```

Does the following use of **testAndSet** successfully implement a critical section? Justify your answer.

```
lock = false;  // global variable lock = true if held
...
while(testAndSet(lock)) yield();
// the critical section
lock = false;
```

> *Mutual exclusion and progress satisfied but not bounded waiting – 3 marks and 1 mark for explanation.*

(d) Consider the problem of deadlock when writing to a file.

(i) [3 marks] State the conditions that must hold for deadlock to occur.

(ii) [9 marks] Could deadlock be prevented by ensuring that one of these conditions does not occur? Briefly explain for each condition why it could or could not be eliminated.

Mutual exclusion - cannot allow multiple writes or reads and writes, will lead to an inconsistent state.

Hold and wait - could prevent process from trying to access any other files at the same time.

No preemption - no, if a resource can't be preempted if there is too much state, e.g. writing a file.

Circular wait - could use a method such as ordered requests for locks to avoid.

Question 3. CPU Scheduling [25 marks]

The processor is a key resource that all operating systems must manage. Without access to the processor a program cannot execute.

(a) [5 marks] Consider a process's life cycle. State four circumstances that might cause CPU scheduling decisions to take place.

1. *When a process switches from the running state to the waiting state (I/O request or invocation of wait/yield)*
2. *When a process switches from the running state to the ready state (for example, when an interrupt occurs)*
3. *When process switches from running state to ready state (for example, completion of IO)*
4. *When process terminates (exit called).*

Extra mark given for full answer. Two marks for identifying one reason for each switch.

(b) Consider a system with one CPU and four jobs. Each job has an arrival time and burst time as given in the table below.

Job	Arrival Time	Burst Time
1	0	5
2	4	10
3	6	1
4	9	9

(i) [5 marks] Draw a Gantt chart and calculate the average **turnaround time** for the round robin scheduling algorithm with quanta of five.

Diagram (1 mark - its a Gantt chart; 2 mark - right except for got two jobs in wrong order; 3 marks - correct)

Calculation (1 mark - calculated waiting time instead or similar mistake, 2 marks - correct). Unless trivial diagram its possible to have 2 marks with incorrect Gantt chart.

Diagram: 0 J1 5 J2 10 J3 11 J4 16 J2 21 J4 24

Turnaround = 5 + 21-4 + 11-6 + 21-9 divided by 4

(ii) [5 marks] Draw a Gantt chart and calculate the average **waiting time** for the preemptive SJF scheduling algorithm.

0 J1 runs to completion
5 J2 runs for 1 unit (9 remaining)
6 J3 runs to completion
7 J2 runs for 2 units (7 remaining)
9 J2 remaining less than J4 so run to completion

16 J4 runs to completion
24 J4 finishes

waiting time = 0 + (5-4) + (6-6) + (16-10) = 11/4

Diagram (1 mark - its a Gannt chart; 2 mark - right except got order of two jobs wrong; 3 marks - correct)

Calculation (1 mark - calculated waiting time instead or similar mistake, 2 marks - correct).

(iii) [10 marks] Kurt wants to implement a preemptive SJF scheduling algorithm for the Nachos operating system. Briefly discuss three problems Kurt would have to solve to implement a fairer scheduling algorithm in Nachos.

He would have to solve three main problems:

- *How to predict the length of the CPU burst. He could use exponential averaging where CPU burst is based upon previous behaviour. For new proceses this requires estimate for new processes. (3 marks)*
- *How to yield control. He would have to work out how to modify the thread admission so that a new scheduling decision could be made when a thread entered the ready queue. (3 marks)*
- *How to prevent starvation. He could use a solution based upon aging. (3 marks)*

Question 4. Memory Management [25 marks]

(a) [4 marks] Define the terms **internal fragmentation** and **external fragmentation**.

External fragmentation – there is enough available memory but it is not contigious. (2 mark) Internal fragmentation – where memory is allocated in fixed size blocks the memory allocated to the process might be larger than how much is actually occupied because can only be allocated in fixed-size blocks. (2 marks)

(b) [5 marks] Sketch a diagram to show the process of mapping a logical address into a physical address where the translation lookaside buffer has cached the mapping between the logical page and physical page frame.

Must indicate components of logical address (page and offset). Must show translation lookaside buffer (or at least the use of it). Must show the physical memory being accessed with the offset applied.

(c) Consider a logical address space of eight pages of 1024 words each, mapped onto a physical memory of 32 frames. The machine architecture reads one word at a time from memory.

(i) [2 marks] How many bits are there in the logical address?

(ii) [2 marks] How many bits are there in the physical address?

Logical address: 13 bits (eight pages - 3 bits, 1024 - ten bits)
Physical address: 15 bits (32 pages - 5 bites, 1024 - ten bits)

(d) [8 marks] Consider a basic demand paging scheme. Outline the steps involved in handling a page fault.

- *Find the location of the desired page on the disk.*
- *Find a free frame: a) if there is a free frame, use it. b) if there is no free frame, use a page-replacement algorithm to select a vicitim frame. c) write the victim frame to the disk; change th page and frame tables accordingly.*
- *Read the desired page into the (newly) free frame; change the page and frame tables.*
- *Restart the user process.*

(e) [4 marks] Discuss how an operating system could detect and eliminate thrashing.

Detect thrashing by evaluating the level of CPU utilisation as compared to the level of multiprogramming. If the CPU utilisation is low but multiprogramming is high then probably thrashing. Can eliminate by reducing the level of multiprogramming.

Question 5. File Systems [25 marks]

(a) [4 marks] List four different file access methods.

- *Sequential access*
- *Direct access*
- *Indexed access*
- *Memory mapped access*

(b) [5 marks] Draw a figure that illustrates the main features of a graph structured directory; in particular identify the possible relationships between directory entries.

Must show: multiple directories, file within directories, cross linked directories and cross linked files, explicitly shows directory entries

(c) Consider the **contiguous**, **linked** and **indexed (single-level)** allocation strategies. Assume the following conditions hold for each problem posed below.

- The file consists of ten data blocks numbered zero to nine.
- Both the file control block and the block information to be stored are already in memory.
- In the case of indexed allocation, the index block is already in memory.
- In the contiguous-allocation case, assume that there is empty space available at the end of the file.

(i) [3 marks] For each strategy, calculate the number of disk I/O operations required to add a block to the end of the file.

Contiguous: 1, Linked: 3 (write new block, read last block, rewrite last block to point there), Indexed: write block

(ii) [3 marks] For each strategy, calculate the number of disk I/O operations required to remove block nine from the file.

Contigious: change length of file 0, Linked: find 8th block (8 reads), update 8th block to point to 10th block = 9, Indexed: 0 update in memory

(d) Consider system-wide open file tables.

(i) [3 marks] Identify three typical types of information contained in a system-wide open file table.

The system-wide open file table would contain file control blocks, accounting information, buffers, a count of the number of processes with this file currently open and locking information (3 marks - must list three of these)

(ii) [3 marks] Imagine implementing a system-wide open file table in Nachos. Constrast this with the implementation of the per-process open file table used in Nachos project two and three.

The Nachos open file table was a per-process file table. It contained a FCB for each open file. If a system-wide file table was being used the per-process file table would only contain pointers to the system-wide fie table. (3 marks)

(iii) [4 marks] Would an operating system using a system-wide open file table perform better when opening files than an operating system only using per-process open file tables? Justify your answer.

With a per-process file table a new FCB must be created every time a file is opened. With a system-wide file table, the system-wide file table can be searched to see if it is already open. If it is already open then nothing more than creating an entry in the per-process file table is required. In the case of multiple processes accessing the existing open file this will lead to better performance.

Question 6. I/O Subsystems [25 marks]

(a) [3 marks] List three dimensions (or characteristics) in which device hardware may vary from each other.

Three of:

- *character-stream versus block*
- *sequential or random access*
- *synchronous or asynchronous*
- *shareable or dedicated*
- *speed of operation*
- *read-write, read only or write only*

(b) [5 marks] Why is using a direct-memory access controller more efficient for transfering large amounts of data than a polling or interrupt mechanism?

DMA hands off processing to a special-purpose processor. CPU just tells it where to read from and where in memory to write to. DMA operates independently writing data directly to memory. With the other approaches the CPU is either busy waiting or interrupted whenever a byte is transferred.

(c) [5 marks] Jordan is implementing a disk scheduling algorithm. There are many factors that can impact upon performance of disk scheduling algorithms. Discuss two main factors that might influence her choice of algorithm.

- *Number of requests. If queue is usually empty then all scheduling algorithms behave the same (FCFS).*
- *Locality of files. If close together will result in limited head movement. Whereas indexed leads to greater movement.*
- *Position of directories.*

(d) Assume a disk drive with 3,000 cylinders numbered 0 to 2,999. The driver is currently serving a request at cylinder 25, and the previous request was at 100. The queue of pending requests, in FIFO order, is:

```
3, 28, 2, 1000, 1500, 1200, 2, 2998, 5, 10
```

For each disk scheduling algorithm specified below, compute the seek schedule, calculate the total head movement, and identify the last cylinder accessed. **Always start from the current head position**.

(i) [4 marks] FCFS algorithm, and

2 marks for the schedule (straightforward), 1 mark for the last position and 1 mark for the total.

The schedule is : 25, 3, 28, 2, 1000, 1500, 1200, 2, 2998, 5, 10.

(ii) [8 marks] LOOK algorithm.

6 marks for the schedule, 1 mark for last position and 1 mark for the total. More marks for this question because it does require a little more calculation.

The schedule is: 25, 28, 10, 5 ,3, 2, 2, 1000, 1200, 1500, 2998.

Question 7. Security [25 marks]

(a) [4 marks] Define the terms **access right** and **protection domain**.

Access right is object-id and rights that may be exercised upon the object, protection domain is set of access rights

(b) Consider the following access control matrix.

	file A	file B	file C
D_1		rw	
D_2	rw		r
D_3		r*	

Domain membership is as follows: Ann is a member of domain D_1; Bob is a member of domain D_2; and, Carol is a member of domain D_3. The access rights are: r is read, w is write, $*$ indicates that the right may be copied from one row to another but must stay in the same column.

(i) [2 marks] Who may write to file A? Justify your answer.

only Bob, only one with an entry r in that column.

(ii) [2 marks] Under what conditions would Bob be able to read file B? Justify your answer.

Yes, Carol could grant him the right, she has a right in her column*

(iii) [4 marks] Specify the access control lists necessary to implement this access control matrix.

file A: (D_2, rw); file B: (D_1, rw)(D_1,r); file C: (D_2,r)*

(iv) [4 marks] Specify the capabilities necessary to implement this access control matrix. What domains would possess these capabilities?

D_1: (file B, rw); D_2: (file A, rw)(file C,r); D_3: (file B, r)*

(c) [9 marks] You are given an operating system that does not authenticate users. You have been asked to design a password-based login system. Describe the login-process and explain the reason behind each step of the process. Note that biometric authentication is not being considered for this system.

1. *Accept username and password. Need username to lookup password. (2 marks)*

2. *Encrypt password. Lookup user entry. Store password in encrypted form to prevent attacker reading the password file. Also use file protection if available. Compare encrypted password. Note that a salt should be used to protect against password collisions.(5 marks)*
3. *Incorrect? Don't tell them which is wrong, makes it harder for the attacker. (1 mark)*
4. *Only allow three logins, increase work factor for the attacker. (1 mark)*
5. *Slow down the process to increase work factor. (1 mark)*

****