THE UNIVERSITY OF QUEENSLAND
AUSTRALIA

This exam paper must not be removed from the venue

Venue	____________________
Seat Number	__________
Student Number	|__|__|__|__|__|__|__|__|__|
Family Name	____________________
First Name	____________________

School of Electrical Engineering & Computer Science

Semester Two Examinations, 2024

COMP3301 Operating Systems Architecture

This paper is for St Lucia Campus students.

Examination Duration: 120 minutes

Planning Time: 10 minutes

Exam Conditions:

- No written or printed material permitted
- Casio FX82 series or UQ approved and labelled calculator only
- During Planning Time - Students are encouraged to review and plan responses to the exam questions

Materials Permitted in the Exam Venue:
(No electronic aids are permitted e.g. laptops, phones)

None

Materials to be supplied to Students:
Additional exam materials (e.g. answer booklets, rough paper) will be provided upon request.

None

Instructions to Students:
If you believe there is missing or incorrect information impacting your ability to answer any question, please state this when writing your answer.

Answer all questions. Answer questions in the spaces provided on the exam paper.
Total 120 Marks

For Examiner Use Only

Question	Mark
1	
2	
3	
4	
5	
6	

Total _________

Question 1. (20 marks)

Calculate performance metrics for various process schedulers.

All of the schedulers are pre-emptive – a process is pre-empted as soon as a higher priority process appears. The larger the priority number, the higher the priority level.

In the following tables, times are all in milliseconds.
Assume processes do not stall for I/O. Ignore scheduler delays. Throughput is measured as processes per second.
The CPU time of aborted processes is included in CPU utilisation.

(a) Complete the following tables for a **Priority** Scheduler. You can assume the kernel is priority pre-emptive. The smaller the priority number, the higher the priority.

Process Number	Arrival Time	Execution Time	Priority	Completion Time	Turnaround Time	Waiting Time
P1	0	4	2			
P2	3	5	4			
P3	6	1	1			
P4	10	3	3			
P5	15	6	1			
P6	20	2	2			

Average Waiting Time	
Average Turnaround Time	
Actual CPU Utilisation	
Average Throughput	

Process Timeline Table. Mark where a process Starts (S), Runs (R), Pauses (P), Ends (E). You can add more columns if required.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38
P1																																							
P2																																							
P3																																							
P4																																							
P5																																							
P6																																							

/10

../Question 1 continued over the page

Question 1 (continued)

(b) Repeat part a) but now assume that two CPU cores (core 1 and core 2) can be used to run processes, concurrently. Complete the following tables for a **Priority** Scheduler. You must consider the core affinity (core 1, core 2 or X – any core). You can assume that a process with no affinity (X) can run on any core.

Process Number	Arrival Time	Execution Time	Priority	Core Affinity	Completion Time	Turnaround Time	Waiting Time
P1	0	4	2	1			
P2	3	5	4	X			
P3	6	1	1	2			
P4	10	3	3	2			
P5	15	6	1	X			
P6	20	2	2	1			

	Core 1	Core 2
Average Waiting Time		
Average Turnaround Time		
Actual CPU Utilisation		
Average Throughput		

Process Timeline Table (Mark where a process Starts (S), Runs (R), Pauses (P), Ends (E). You can add more columns if required.

Core 1

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38
P1																																							
P2																																							
P5																																							
P6																																							

Core 2

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38
P2																																							
P4																																							
P3																																							
P5																																							

/10

Total ____/ 20

Question 2. (20 marks)

Consider a file system where disk block addresses are represented by a **16-bit** integer. A disk block is **512** bytes long.

Assume that a single command (read or write) to the disk controller consists of a starting disk block address, the number of contiguous blocks to read/write, and the location of the memory buffer for the data. Assume that commands can be sent continuously to the controller (you don't need to wait until the previous one has been serviced unless the next read needs the contents of the previous block). Assume that disk requests are serviced in the order received and with a latency of **1ms** between each request (from the end of the previous data transfer or from the time the command was received, whichever is longer), and a streaming transfer rate of one block per **25µs**.

Calculate the requested information for each particular file system in question 2 **a)** and **b)**. Assume that CPU operation times can be neglected. Assume that the whole file can fit into memory if needed.

../Question 2 continued over the page

Question 2 (continued)

(a) Contiguous allocation File System
Assumptions – the file control block is in memory, the file index allocation table is in memory, any new data that needs to be written to the file is in memory. Assume that all the disk blocks in the file can fit in memory.

Calculate the number of disk controller commands, the number of disk block I/O operations (i.e. total number of disk block reads and writes), and the total time to complete each of the following operations.

Starting Disk Block: 31
File length: 9,728 bytes

	Number of disk controller commands	Number of disk block I/O operations	Total Time
(i) One block is added to the end of the file.			
(ii) One block is added to the beginning of the file.			
(iii) Data in location 2074 (bytes, from top of file) is altered.			
(iv) One block is removed from the end of the file.			
(v) Block 39 is removed from the file.			
(vi) The file is deleted (data does not need to be overwritten).			

Working

/6

../Question 2 continued over the page

Question 2 (continued)

(b) Linked File System

Assumptions – the file control block is in memory, each file block contains a link to the next file block, any new data that needs to be written to the file is in memory. Assume the disk blocks in the file are not contiguous.

Calculate the number of disk controller commands, the number of disk block I/O operations (i.e. total number of disk block reads and writes), and the total time to complete each of the following operations.

Use the same Disk Blocks used for the file (without modifications) in part **a)**

	Number of disk controller commands	Number of disk block I/O operations	Total Time
(i) One block is added to the end of the file.			
(ii) One block is added to the beginning of the file.			
(iii) Data in location 2074 (bytes, from top of file) is altered.			
(iv) One block is removed from the end of the file.			
(v) Block 39 is removed from the file.			
(vi) The file is deleted (data does not need to be overwritten).			

Working

/6

../Question 2 continued over the page

Question 2 (continued)

(c) Free Space Map

You can assume the last disk block in the file is the last disk block of the disk volume.

Use the same Disk Blocks used for the file (without modifications) in part **a)**

(i) Calculate the memory (in bytes) required to store the bitmap of free space blocks of the disk volume? You must use the given disk block address range (given in Question 2) to calculate the total number of blocks.

__ /2

(ii) Show a subset of the bits of the free space bitmap of the file, from bit index 24 to 39. Use the table below. Use a 1 for block free and 0 for block occupied.

24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39

__ /2

(iii) Which file allocation system type can be easily retrieved using a free space bitmap?

__ /2

(iv) What is a disadvantage of using a linked file allocation to track free space?

__ /2

Total ____/ 20

Question 3. (20 marks)

A process accesses the following data memory pages, as seen in the table below. During each iteration, the corresponding pages are accessed, e.g. pages 0 and 1 are accessed in iteration 0.

Table showing Memory Pages Numbers accessed

Iteration 0	0	2	1	2	3
Iteration 1	1	1	2	1	5
Iteration 2	2	3	6	3	2
Iteration 3	3	1	1	1	9

(a) At the end of iteration 1, calculate the total number of page faults generated if the First In First Out (FIFO) page replacement algorithm is used, with **i)** 2 page buffer and **ii)** 4 page buffer. Also state what the minimum buffer size is to ensure that no page faults occur.

i)

ii)

iii) minimum buffer size = ______

/5

(b) At the end of iteration 3, calculate the total number of page faults generated if the Least Recently Used (LRU) page replacement algorithm is used, with **i)** 2 page buffer and **ii)** 4 page buffer. Also state what the minimum buffer size is to ensure that no page faults occur.

i)

ii)

iii) minimum buffer size = _____

/5

../Question 3 continued over the page

Question 3 (continued)

(c) What are three hardware support features needed to support demand paging?

i)

ii)

iii)

__/3

(d) Copy-on- Write (COW)

i) What is Copy-on-Write (COW)?

ii) Why does it allow for more efficient process creation?

__/2

(e) Free Pages

i) What is anonymous memory?

ii) What must a free page be filled with before allocation?

__/2

(f) What are three ways to optimise demand paging?

i)

ii)

iii)

__/3

Total ____/ 20

Question 4. (20 marks)

(a) List and describe two types of semaphores?

i)

ii)

__/4

(b) What are two atomic operations that are used to control a semaphore?

i)

ii)

__/2

(c) If a semaphore is implemented with a waiting queue, describe what happens with the queue when the semaphore performs the following operations.

Block)

Wakeup)

__/2

(d) What is the key difference between a mutex lock and a semaphore?

__/2

../Question 4 continued over the page

Question 4 (continued)

(e) What type of processor architecture is suitable for a mutex lock?

__/2

(f) What happens if a process is never removed from a semaphore's wait queue?

__/2

(g) Consider a data set that is shared among a number of concurrent processes (Readers and Writers). Describe how the following semaphores and count is used. Note: Readers can only read the data set and they do not perform any updates. Writers can both read and write.

Semaphore rw_mutex)

Semaphore mutex)

Integer read_count)

__/6

Total ____/ 20

Question 5 (20 marks)

(a) Describe the following Virtual Machine Managers (VMM) or hypervisors.

Type 0 Hypervisor)

Type 1 Hypervisor)

Type 2 Hypervisor)

__/6

(b) Describe the following variants of VMMs.

Paravirtualisation)

Emulation)

Container)

__/6

../Question 5 continued over the page

Question 5 (continued)

(c) A VMM is assigned a fixed amount of memory when it boots up. If a guest OS requires more memory than the VMM can provide, describe two ways in which a VMM can mitigate this problem with double paging and using a 'memory balloon' driver.

i)

ii)

__/4

(d) List four ways in which a VMM can provide IO access to a guest OS.

i)

ii)

iii)

iv)

__/4

Total ____/ 20

Question 6. (20 marks)

You have been asked to design an operating system for an online-based ticket booking portal used to book tickets for the singer T-swizzler's concert tour. The ticket booking portal is accessed by users via a web page and mobile app interfaces, from which a user orders a ticket from. Once a ticket has been purchased, it must be printed by an administrator on a restricted printer with special paper before being posted to the user. Due to T-swizzler's large fan base, the portal must be able to service millions of users, every hour. The portal will also be a target for malicious users and therefore needs to ensure that only authorised users and administrators can access the portal. All ticket holder records must be stored. To save cost, as T-swizzler needs a VIP helicopter to travel to the concert venue, only inexpensive magnetic hard-drive storage (prone to high rates of failure with read/write head breakages) with RAID (0+1) can be used. The operating system should use multiple CPU cores for parallelism.

You can also assume the following:

- The operating systems uses multiple CPU cores for redundancy.
- Magnetic disk mass storage is used. Rapid changes in the read/write head direction will cause a breakage/fault.
- The restricted printer can only be accessed by an administrator user.

For each of the indicated aspects of the operating system:

(i) Describe your suggested design alternative,

(ii) Give one other less favourable choice,

(iii) Explain two advantages of the suggested alternative compared to the less favourable choice

(a) What type of Multithreading Model (one-to-one, one-to-many, many-to-many) should be used? (i) Suggested solution (ii) Less favoured alternative (iii) Two advantages of suggested solution
/5

../Question 6 continued over the page

Question 6 (continued)

(b) What type of CPU scheduler should be used?
(i) Suggested solution (ii) Less favoured alternative
(iii) Two advantages of suggested solution

__/5

../Question 6 continued over the page

Question 6 (continued)

(c) What access control or type of list should be used to ensure that only authorised users and administrators can have access to the restricted printer? (i) Suggested solution (ii) Less favoured alternative (iii) Two advantages of suggested solution
/5

../Question 6 continued over the page

Question 6 (continued)

(d) What disk scheduling should be used? (i) Suggested solution (ii) Less favoured alternative (iii) Two advantages of suggested solution
__/5

Total ____/ 20

BLANK PAGE FOR WORKING

END OF EXAMINATION