NATIONAL EXAMINATIONS December 2015

98-COMP A-5 OPERATING SYSTEMS

3 Hour Duration

NOTES:

1. If doubts exist as to the interpretation of any question, the candidate is urged to submit with the answer paper a clear statement of any assumption made.

2. Provide justifications for your answers. Show all your work.

3. CLOSED BOOK. Candidates may use one of the two pocket calculators, the Casio approved model or Sharp approved model. No other aids.

4. The candidate has to answer **any five questions** (each question has multiple parts).

5. Total Marks = 100.

6. This exam has got 6 pages (including this page).

98- COMP-A5 OPERATING SYSTEMS

1 [20 marks].

Consider the following arrivals on a system. Each process has a single CPU burst and does not perform any I/O.

Process	Arrival Time (seconds)	Execution Time(seconds)
Proc1	0	16
Proc2	2	6
Proc3	3	12
Proc4	6	3

(a) What is the minimum mean process turnaround time that can be achieved by any non-preemptive CPU scheduling strategy?

(b) Find the mean process turnaround time when the Round Robin (RR) policy (with a time slice of 2 seconds) is used for CPU scheduling.

(c) Find the mean process turnaround time when a variant of the Round Robin policy is used for CPU scheduling. In this strategy the time slice allocated to a process is based on its execution time. For processes with execution times that are below 7 seconds a time slice of 2 seconds is used. For processes with execution times equal to or greater than 7 seconds but below 13 seconds a time slice of 3 seconds is used. For all other processes a time slice of 5 seconds is used.

2. [20 marks].

(a) Briefly explain what is meant by a demand-paged virtual memory system.
Briefly discuss the advantages of a demand-paged virtual memory systems.

(b) Briefly discuss the program characteristics that are responsible for making virtual memory work.

(c) Consider a demand paged virtual memory system in which a single program is currently running. The page map table is held in associative registers (associative memory).

It takes 20 milliseconds to service a page fault if an empty frame is available or the replaced page is not modified, and 50 milliseconds if the replaced page is modified. Memory access time is 100 nanoseconds. Assume that for 60% of the page faults a page replacement is necessary and the page to be replaced is modified.

What is the maximum acceptable page fault rate such that the effective memory access time for the program is not greater than 200 nanoseconds?

3 [20 marks].

(a) Consider the following sequence of memory addresses generated during a program execution on a demand paged virtual memory system:

102, 1001, 1249, 8800, 9991, 2780, 2701

(i) Derive the page reference string for a system with a page size of 100.
(ii) What is the minimum number of page faults for this page reference string?

(b) Consider a demand paged virtual memory system and the following page reference string:

101, 112, 113, 114, 115, 113, 114, 111, 116, 117, 118, 117, 118, 118, 117.

What is the number of page faults generated when 5 frames are allocated to a program for:
(i) the LRU page replacement policy (ii) the optimal page replacement policy?

(c) Briefly differentiate between compile time and execution time address binding for programs.

4 [20 marks]

(a) Different methods exist for storing information on the disk. Consider a file currently consisting of 150 blocks (numbered 1 - 150). Assume that the directory is available in main memory.

For each of the following cases (A-D) compute the minimum number of disk operations that are required when linked allocation is used.

(A) The contents of block 100, 98 and 102 are read.
(B) The contents of block 110 are exchanged with the contents of block 95.
(C) Two new blocks are inserted after block 80. The contents of both these new block are the same as that of block 100.
(D) Blocks 75 and 150 are deleted.

Consider each case (A-D) separately. Note that each disk operation corresponds to the reading of a block from the disk or the writing of a block to the disk. While computing the number of disk operations, ignore the disk operations that may be required for the location and maintenance of free space. Since the directory is in main memory any operation on the directory is not counted as a disk operation.

ASSUME: The length of the file is known to the system.

(b) Consider an operating system which uses a Process Table for short term scheduling (CPU scheduling). The Process Table contains a set of records each of which represents a Process Control Block (PCB). The PCB contains the current state of the process (ready, blocked or running). Another field in the PCB is ltime that is incremented every T units of time by the operating system. Every 100T units of time the operating system also multiplies the ltime for each existing process by a factor M (M >1). The initial value of ltime is set to 0 when the process first enters the multiprogramming mix.

The short term scheduler used by this operating system is fairly simple: it performs a new scheduling decision only when the running process blocks for I/O or it completes. The short term scheduler scans the Process Table starting from the top and examines each PCB. The ready process that has the smallest ltime is selected to run on the CPU.

Identify as many defects as you can in the design of the short term scheduler. Will using a value of M<1 improve the situation?

5 [20 marks]

(a) Briefly discuss the role of multiple CPUs and multiple disks in improving system reliability.

(b) Consider a system comprising R resources of the same type on which P processes are running. For each of the following cases determine whether the system is deadlock free and a deadlock can never occur on the system. Clearly justify your answer by making reference to the four necessary conditions for the occurrence of a deadlock. Provide clear descriptions of system states in explaining your answer.

(i) Resources are non-sharable. Once a process has acquired a resource it must give up the resource voluntarily before another process can use it. R= 5; P =4. Each of the processes can hold up to a maximum of 2 resources at a time

(ii) Resources are non-sharable. Once a process has acquired a resource it must give up the resource voluntarily before another process can use it. R= 10; P =8. Each of the processes can hold up to a maximum of 3 resources at a time.

(iii) R = 2 and P = 10. All the resources in the system are read-only files. Any process can access at most 2 resources at a time.

(iv) Resources are non-sharable. R= 10; P =8. Each of the processes can hold up to a maximum of 3 resources at a time. Two of the processes are high priority processes. Only these high priority processes can preemptively capture a resource from a low priority process if necessary.

6 [20 marks]

Consider a system in which multiple processes execute concurrently. The system contains a shared data record, which may be concurrently accessed by these processes. A **monitor** is used to control access to the shared data. Multiple processes can share the data concurrently as long as the number of processes sharing the data is less than H. A requesting process is blocked if allowing the process to access the data will violate this condition. A blocked process is allowed to proceed when one of the processes exits the monitor. For example, consider a system in which H=2. If Pi and Pj are currently accessing the shared data and Pk wants to concurrently access the data, the access is not allowed because allowing Pk to share the data concurrently will make the number of processes sharing the data to be greater than H. On the other hand if Pj was the only process using the shared data when Pk requests access it is allowed to proceed.

The monitor contains two procedures: get_access and done_access. Whenever a process wants to access the record it calls the procedure get_access. If the calling process satisfies the condition described earlier it is allowed to proceed; it is **blocked** otherwise.

Whenever a process finishes its operation on the data record it calls the procedure done_access. If one or more processes is/are blocked when done_access is called, one of the blocked processes is allowed to access the record after the completion of the done_access procedure.

The typical operations performed by a process Pj are given by the following algorithm.

Process Pj

repeat

1. Perform computation.
2. Call procedure get_access in the monitor
   {If the process is not blocked inside the monitor it means that the desired operation on the shared record can be performed by the process.}
3. Perform the desired operation on the data record.
4. Call procedure done_access in the monitor.

until false

(a) Write the algorithm (pseudo-code) for the monitor that will control access to the data record. The monitor must contain the two procedures get_access and done_access (described above) that are called by the processes. You may incorporate other procedures/functions inside the monitor if necessary.

Explain your solution clearly.
Declare the variables required by your solution in the variable declaration part of the monitor. <u>However this variable declaration CAN NOT include a variable of type semaphore.</u>

The generic structure of a monitor is included below for your assistance.

```
monitor monitor-name
        {
                variable declarations
                procedure P1 (...) {
                        . . .
                }
                procedure P2 (...) {
                        . . .
                }
                procedure Pn (...) {
                        . . .
                }
                {
                        initialization code
                }
        }
```

(b) Describe the three requirements associated with the solution to the critical section problem. Discuss your solution in the context of these three requirements.

7 [20 marks]

(a) Consider a webserver and an alarm system. On the webserver responses to requests need to be performed within 5 seconds for at least 90% of the requests received. In the alarm system the alarm bell must be switched on within 900 ms of the occurrence of the trigger condition for the alarm. For each system discuss whether it needs to be considered as a hard real time or a soft real time system.

(b) Briefly explain how the following disk scheduling algorithms work: SSTF and FCFS.
Include in your discussion their expected performance in terms of minimizing the movement of the disk head and the scheduling overheads accrued during their operation.

(c) File protection is an important goal for an operating system in a multiuser computing system. Discuss with the help of examples any one method of controlling file access used for achieving protection in a file system. Include the overhead associated with the method described.

(d) What is an inode? Briefly discuss its characteristics and use in the Unix operating system.

Approximate Marking Scheme

1(a) 6 marks
1(b) 7 marks
1(c) 7 marks

2(a) 6 marks
2(b) 4 marks
2(c) 10 marks

3(a) 6 marks
3(b) 12 marks
3(c) 2 marks

4(a) 12 marks
4(b) 8 marks

5(a) 4 marks
5(b) 16 marks

6(a) 15 marks
6(b) 5 marks

7(a) 4 marks
7(b) 6 marks
7(c) 5 marks
7(d) 5 marks