98- COMP-A5 OPERATING SYSTEMS

NATIONAL EXAMINATIONS December 2013

98-COMP A-5 OPERATING SYSTEMS

3 Hours 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 5 pages (including this page).

1 [20 marks].

(a) 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	15
Proc2	2	7
Proc3	4	18
Proc4	6	2

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

(ii) Find the mean process turn around time when the FCFS policy is used for CPU scheduling.

(iii) Find the mean process turn around time when a variant of the Round Robin strategy 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 5 seconds a time slice of 2 seconds is used. For processes with execution times equal to or greater than 5 seconds but below 10 seconds a time slice of 3 seconds is used. For all other processes a time slice of 4 seconds is used.

(b) Explain why CPU scheduling is more difficult to perform on a system comprising multiple processors in comparison to a single CPU-based system.

2 [20 marks].

(a) Given below is a solution to the critical section problem involving two concurrent processes A and B. Identify as many **distinct problems** as you can in the design. If similar problems occur at multiple places identify them each time but explain it only once. Your list of errors should include defects (if any) that may not necessarily give rise to incorrect results but do indicate flaws in design. Justify your answer with the help of examples. Be as specific as you can when you describe the situations in which problems occur.

Algorithm

Process A

```
repeat
        want_access [0] = true;
        if want_access [1] {
                want_access [0] = false;
                while want_access[1]  no-op;
                want_access [0] = true;
        };

        Code for CS

        want_access [0] = false;
until false;
```

Process B

```
repeat
        want_access [1] = true;
        if want_access [0] {
                want_access[1] =false;
                while want_access[0]  no-op;
                want_access [1] = true;
        };

        Code for CS

        want_access [1] = false;
until false;
```

Note: CS: Critical Section. no-op: no operation.

(b) With the help of examples discuss the three requirements associated with the solution to the critical section problem.

(c) Discuss with the help of examples the role of a semaphore in solving the critical section problem.

3 [20 marks].

(a) Consider a multiprogrammed system consisting of 11 resources of the same type. No deadlock handling technique is employed by the system. That is, if a resource is requested by a process and one is available, a resource is allocated to the requesting process; otherwise the requesting process is blocked. P processes are run concurrently on the system. Each process can simultaneously hold up to 3 resources at any given point in time. Once a resource is acquired by a process, it must be released by the process before it can be assigned to another process. Assume that each process requests and releases one resource at a time.

(i) For P = 2, can a deadlock occur on the system?
(ii) For P = 14 can a deadlock occur on the system?
(iii) Determine the maximum value of P such that a deadlock can never occur on the system.

(b). Discuss with the help of examples the necessary conditions for a deadlock to occur on the system.

4 [20 marks].

(a) Distinguish between internal and external fragmentation in the context of memory management on a multiprogrammed computer system. Describe a single technique that can control both types of fragmentation on the system.

(b) Consider a multiprogrammed system that uses multiple partitions (of variable size) for memory management. A linked list of holes called the free list is maintained by the operating system to keep track of the available memory in the system. At a given point in time the free list consists of holes with sizes:

202K, 143K, 305K, 380K, 170K, 191K, 225K, and 250K

The free list is also ordered in the sequence given above: the first hole in the list is of size 202K words which is followed by a hole of size 143K words and so on. Jobs with different memory requirements arrive on the system in the following order:

	Arrival Time	Memory Requirement
Job 1	t1	222K
Job 2	t2	205K
Job 3	t3	303K
Job 4	t4	190K

[Given t1 < t2 < t3 < t4]

Explain how memory allocation would be performed in the given situation for (i) the best fit and (ii) the first fit policy. [For each policy determine which hole is allocated to each job after it arrives on the system].

5 [20 marks].

(a) Discuss the advantages and the overheads associated with the use of (i) multiple CPUs and **(ii)** multiple disks on a computing system

(b) Consider a moving head hard disk which consists of a single platter (surface) with 100 tracks on it. The tracks are numbered 0 to 99. The disk has just completed a request at track 50 and is currently serving a request at track 53. The queue of pending requests in FIFO order is:

38, 66, 92.

(i) What is the total head movement (in number of tracks) needed to satisfy all these requests when the Shortest Seek Time First (SSTF) algorithm is used for disk scheduling?

(ii) What is the total head movement (in number of tracks) needed to satisfy all these requests when the SCAN algorithm is used for disk scheduling?

(iii) In what order should the requests be served to minimize the total head movement?

[Assume that no further requests arrive on the system during the service of the above requests.]

(c) Briefly explain execution time address binding in the context of memory management.

6 [20 marks].

(a) What are links? Discuss their roles in the context of file sharing. Include the problems associated with the deletion of shared files in your discussion.

(b) Discuss why controlling accesses to files are important in the context of multi-user systems. Describe any two methods of maintaining access rights for files on the system and how they are used for access control.

(c) Different methods are available for the management of blocks allocated to files. Consider a file currently consisting of 125 blocks. Assume that the directory is available in main memory). The 53rd block in the file needs to be deleted. For each of the following cases compute the minimum number of disk operations that are required. Assume that there is no room for the file to grow in the beginning but there is room to grow in the end.

(i) contiguous allocation (ii) linked allocation (based on a singly linked list) (iii) indexed allocation (assume that only one index block is used)

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.

7 [20 marks].

(a) Why are real time systems usually pre-emptive in nature? Using examples, discuss how priority inversion can occur on a real time system.

(b) Briefly distinguish among a safe state, an unsafe state and a deadlock state. Using a system running four processes that use at least two different types of resources provide an example of (i) an unsafe state and (ii) a deadlock state.

(c) Using examples explain the optimal strategy for page replacement on a virtual memory system on which a program is allocated a fixed number of frames. Why is it difficult to implement on a real system?

(d) Briefly discuss the roles of the *security* and *protection* mechanisms used on a computer system supporting multiple users.