NATIONAL EXAMINATIONS MAY 2005

98-COMP A-5/SOFT 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 five questions (each question has multiple parts).

Answer ANY ONE of questions 1 and 2, ANY ONE of questions 3 and 4, ANY THREE of questions 5, 6, 7, and 8.

5. Total Marks = 100.

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

Answer question 1 OR question 2

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)
P1	0	20
P2	2	10
P3	4	20
P4	6	1
P5	8	5

Find the mean process turn around time in each of the following cases.

(i) When the FCFS policy is used for CPU scheduling

(ii) When the optimal non-preemptive CPU scheduling policy is used for CPU scheduling

(iii) When a multi-level feedback queue based strategy is used. There are three FIFO queues in the system. Processes in level i queue have a preemptive priority of over processes in level j queue (j > i). Processes upon arrival join the tail of the level 0 queue. A time slice of 5 seconds is associated with the processes in level 0 queue. When a process from level 0 queue runs and does not complete before the expiry of its time slice it is preempted and added to the tail of the level 1 queue. A time slice of 10 seconds is associated with level 1 queue. When a process from level 1 queue runs and does not complete by the expiry of its time slice it is added to the tail of the level 2 queue. Processes in level 2 queue are run to completion (i.e. an FCFS policy is used).

(b). What is a real time system? Using examples, discuss the differences among a hard real time, a soft real time and a general purpose time sharing system. [Be brief].

2 [20 marks].

(a) Consider an operating system which uses a <u>Process Table</u> for short term scheduling (CPU scheduling). The Process Table contains a set of records each of which represents a <u>Process Control Block (PCB)</u>. The PCB contains the current state of the process (ready, blocked or running). Another field in the PCB is <u>pval</u> that is incremented every T units of time by the operating system. Every 25T units of time the operating system also multiplies the pval for each existing process by a factor K (K >1). The initial value of pval 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 pval is selected to run on the CPU.

Identify as many defects as you can in the design of the short term scheduler.

(b). Three processes P1, P2, and P3 arrive at times 0.0, 0.5, and 1.0 seconds respectively on the system. Each of these processes has a single CPU burst. They do not perform any I/O. The CPU burst durations for P1, P2, and P3 are 8.5, 4.6, and 1.0 seconds respectively.

A vendor claims that if his proprietary CPU scheduling policy is used then a mean turn around time of 6.9 seconds will be achieved for the three processes described above. The systems engineer for the company (that owns the computer), however, says that the vendor cannot deliver what he has promised.

Who is right: the vendor or the systems engineer? Justify your answer.

Answer question 3 OR question 4

3 [20 marks].

(a) What is meant by "locality" of a program? Describe briefly, the Working Set based memory management policy and explain how it dynamically controls the degree of multiprogramming to achieve high system performance.

(b). 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 T1 milliseconds to service a page fault if an empty frame is available or the replaced page is not modified, and T2 milliseconds if the replaced page is modified. Memory access time is A nanoseconds.

Assume that for P% 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 2A nanoseconds?

4 [20 marks].

(a). Consider the following page reference string on a demand paged virtual memory system:

21,22,23,24,22,21,25,26,22,21,22,23,27,26,23,22,21,22,23,26

Determine the number of page faults that would occur with the following page replacement algorithms when 5 frames are allocated to the program. [Remember that the frames are initially empty and the first reference to a page will cause a page fault].

(i) First In First Out (FIFO)
(ii) Least Recently Used (LRU)

(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:

100K, 40K, 200K, 180K, 70K, 90K, 120K, and 150K

The free list is also ordered in the sequence given above: the first hole in the list is of size 100K words which is followed by a hole of size 40K words and so on. Jobs with different memory requirements arrive on the system in the following order:

	Arrival Time	Memory Requirement
Job 1	t1	120K
Job 2	t2	100K
Job 3	t3	90K
Job 4	t4	200K

[Given t1 < t2 < t3 < t4]

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

Answer any THREE of questions 5, 6, 7, and 8.

5 [20 marks].

(a) Consider the following semaphore-based solution to the producer consumer problem. Assume that the size of the circular buffer is N.

Identify the distinct problems in the solution. 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 Producer

```
repeat
        produce item in nextp;
        wait (mutex);
        while full do
        begin
                signal (mutex);
                wait (mutex);
        end
        counter := counter + 1;
        if counter = N then
                full := true;
        empty := false;
        signal (mutex);
        buffer [in] := nextp;
        in := (in + 1) mod N;
until false;
```

Algorithm Consumer

```
repeat
        wait (mutex);
        while empty do
        begin
                signal (mutex);
                wait (mutex);
        end
        counter := counter - 1;
        if counter = 0 then
                empty := true;
        full := false;
        signal (mutex);
        nextc := buffer [out];
        out := (out + 1) mod N;
        consume item in nextc;
until false;
```

[**NOTE:** The initial values for variables "empty" and "full" are true and false respectively. Assume that the variable "counter" (used for keeping track of the number of unconsumed elements), the index variables "in" and "out" as well as the semaphore "mutex" are appropriately initialized].

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

6 [20 marks].

(a). Briefly discuss the following disk space allocation strategies: indexed allocation, and linked allocation. Compare the two strategies focusing on their merits and deficiencies.

(b). Consider a moving head hard disk which consists of a single platter (surface) with 200 tracks on it. The tracks are numbered 0 to 199. The disk is currently serving a request at track 130 and has just finished a request at track 128. The queue of pending requests in FIFO order is:

76, 137, 81, 167, 84, 140, 92, 165, 120.

What is the total head movement (in number of tracks) needed to satisfy all these requests for the following disk scheduling algorithms?

(i) Shortest Seek Time First (SSTF) (ii) SCAN

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

(c). Briefly explain how the appropriate data block on disk is identified when a read system call is made on the Unix operating system.

7 [20 marks].

(a) Consider a multiprogrammed system with **R** devices of the same type. Six processes P1.. P6 run concurrently on the system and use these devices. If any process requests for a device then any of the unused devices can be given to the process to satisfy the request.

Show that the system is deadlock free when the sum of the maximum needs (for these devices) of all these processes is less than R+6.

Note: "Maximum need of a process is M" means that a maximum of M devices can be held simultaneously by this process.

Once a device 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 device at a time and the maximum need for each process is between 1 and R.

[A system is said to be deadlock free if there are enough resources in the system and we do not need to employ any deadlock handling technique. If a device is free, it can be immediately allocated to a requesting process, yet no deadlock can occur.]

(b) What is a deadlock? Discuss briefly the necessary conditions for the occurrence of a deadlock.

8 [20 marks]. Answer any *five* of the following questions.

(a). Briefly describe the data encryption mechanism employed by an operating system for the transmission of data over unreliable links.

(b) Address binding: the binding of instructions and data to memory addresses can be done at different times. Briefly differentiate among the three different methods (each of which can be applied at a specific time in the lifetime of a program) available for address binding.

(c) Describe the bit vector-based strategy for managing free space on a disk. Is it sufficient to store the bit vector in the RAM-based main memory ?

(d) Differentiate between symmetric and asymmetric multiprocessing. Discuss why symmetric multiprocessing is considered to be harder in comparison to asymmetric multiprocessing.

(e). Discuss briefly the method based on access lists that can be used for file protection by an operating system. Include the merits and deficiencies (if any) of the method in your discussion.

(f). What is a Redundant Array of Inexpensive Disks (RAID) ? Discuss the motivations behind using a RAID on a computer system.