CS 346 – Review for Test #2 (Scheduling, deadlock, memory and disk management)

1. Consider a system with 4 types of resources: R1 (3 units), R2 (2 units), R3 (3 units), R4 (2 units). A non-preemptive resource allocation policy is used. At any given instant, a request is not entertained if it cannot be completely satisfied. A process unable to obtain a resource will sleep until its request can be granted. For example, if a process requests an R3 and an R4 and only an R4 is available, then the process is not given either resource and it goes to sleep. A sleeping process cannot do any computation. It will automatically awaken and acquire the resources when they are all available. Three processes, P1, P2 and P3, request resources as follows:

Process P1: request 2 units of R2 1 second of computation request 1 unit of R3 2 seconds of computation request 2 units of R1 2 seconds of computation *release* 1 unit of R2 *release* 1 unit of R1 2 seconds of computation *release* 1 unit of R3 1 second of computation request 2 units of R4 2 seconds of computation	Process P2: request 2 units of R3 2 seconds of computation request 1 unit of R4 2 seconds of computation request 1 unit of R1 2 seconds of computation *release* 1 unit of R3 2 seconds of computation	Process P3: request 1 unit of R4 2 seconds of computation request 2 units of R1 3 seconds of computation *release* 2 units of R1 2 seconds of computation request 1 unit of R2 1 second of computation request 1 unit of R3 1 second of computation

Assume that:

- Processes can do their computations in parallel on different CPUs.
- A request and release of a resource take an insignificant amount of time.
- When a job is finished, it releases all its resources.
- All processes begin at time 0.

a. Create a table itemizing what occurs to each process and each resource at each point in time. In other words, state what happens at time = 0 seconds, time = 1 second, etc. Continue the table until all jobs have completed. Mention whether resource requests are granted and if a job goes to sleep.

b. Will the processes all be able to finish without any deadlock? If there is deadlock, when does it occur and which processes are involved?

c. Do any processes have to sleep? If so, which one(s) and when?

2. Assume that a certain system has a total of 12 DVD drives. There exist four processes P1, P2, P3 and P4 requiring a maximum of 6, 5, 4, and 7 DVD drives respectively. At a given time, there is 1 drive allocated to P1, 5 to P2, 2 to P3 and 4 to P4. Is this a safe state? Why or why not? If it is safe, what is the safe sequence?

3. Suppose an operating system uses the non-preemptive Shortest Job Next scheduling algorithm. Consider the arrival and execution times for the following four processes.

Process #	Execution time	Request/arrival time
1	20	0
2	25	15
3	15	30
4	10	45

a. Construct a Gantt chart showing what interval of time each process executes and is in its ready state. Also indicate the system load during each interval of time.

b. What is the turnaround time of each process?

c. What is the average system load between time = 0 and the time when the last process has finished executing?

d. What is the maximum system load? During what interval(s) of time is this maximum realized?

4. Suppose we wish to run two periodic tasks, T1 and T2, in a hard real-time system. The periods of the tasks are 9 for T1 and 12 for T2. The system uses the rate monotonic scheduling algorithm. Compute the maximum CPU utilization for the following five scenarios: when the execution time of task T1 is 1, 2, 3, 4 and 5.

5. Which type of memory or disk allocation is particularly susceptible to external fragmentation? How does it occur?

6. Briefly explain the rationale of RAID bit striping.

7. What is the difference between a page fault and a segmentation fault?

8. What is a working set? What is its purpose? What information is needed to compute the working set?

9. Suppose main memory consists of 3 frames. Consider the following reference string of logical page numbers. 6, 1, 5, 6, 2, 1, 8, 1, 5. Determine the contents of RAM after each memory access in the reference string, and indicate which accesses are page faults for the following page replacement algorithms:
   a. Second chance FIFO

   b. LRU

   c. clairvoyant

10. In disk scheduling, the initial state of a disk is the list of track numbers that need to be visited, the current track location of the read/write head, and the direction (ascending or descending) of the read/write head.
    a. Give an example of an initial disk state for which the elevator algorithm has better performance than shortest seek next. Justify your answer.

    b. Give an example of an initial disk state for which shortest seek next has better performance than the elevator algorithm. Justify your answer.

11. Suppose a paging memory system has a RAM size of 128 MB and a page size of 8 KB. Assume that we wish to support a logical address space of 4 GB.
    a. For a logical address, how many bits are in its page offset?

    b. For a physical address, how many bits are in its page offset?

    c. How many bits are allocated to a logical page number?

d. How many bits are allocated to a physical page number?

e. What is the total number of logical pages?

f. What is the total number of physical pages (frames)?

g. How much space is needed to store the page table?

h. Suppose the integer variable x has virtual (i.e. logical) address 0x12345678. What is the range of hexadecimal address numbers that are in the same virtual page as x?