C 3144

B.E./B.Tech. DEGREE EXAMINATION, MAY/JUNE 2007.

Fourth Semester

(Regulation 2004)

Computer Science and Engineering

CS 1252 — OPERATING SYSTEMS

(Common to B.E. (Part-Time) Third Semester Regulation 2005)

Time : Three hours Maximum : 100 marks

Answer ALL questions.

PART A — ($10 \times 2 = 20$ marks)

1. Under what circumstances would a user be better off using a timesharing system rather than a PC or single-user workstation?

2. Discuss the differences between symmetric and asymmetric multiprocessing.

3. Can a multithreaded solution using multiple user-level threads achieve better performance on a multiprocessor system than on a single-processor system? Justify.

4. Discuss why implementing synchronization primitives by disabling interrupts is not appropriate in a single-processor system if the synchronization primitives are to be used in user-level programs.

5. What is a translation look aside buffer used for?

6. What is the purpose of paging the page tables?

7. Is it possible for a process to have two working sets? One representing data and another representing code? Explain.

8. Give an example of an application that could benefit from operating system support for random access to indexed files.

9. What are the advantages of the variation of linked allocation that uses a FAT to chain together the blocks of a file?

10. What are the primary goals of the conflict resolution mechanism used by the Linux kernel for loading kernel modules?

PART B — (5 × 16 = 80 marks)

11. (a) (i) How do clustered systems differ from multiprocessor systems? What is required for two machines belonging to a cluster to cooperative to provide a highly available service? (4)

(ii) Which of the functionalities listed below need to be supported by the operating system for Real-time systems and Hand-held devices? (4)

(1) Batch programming

(2) Virtual memory

(3) Time sharing.

(iii) Distinguish between the client-server and peer-to-peer models of distributed systems. (4)

(iv) What are the advantages and disadvantages of using the same system call interface for manipulating both files and devices? (4)

Or

(b) (i) State the purpose of short-term, medium-term, and long term schedulers. Also discuss the differences among them. (6)

(ii) List and discuss three general methods for passing parameters to the operating system. (10)

12. (a) (i) Consider a multiprocessor system and a multithreaded program written using the many-to-many threading model. Let the number of user-level threads in the program be more than the number of processors in the system. Discuss the performance implications of the following scenarios. (6)

(1) The number of kernel threads allocated to the program is less than the number of processors.

(2) The number of kernel threads allocated to the program is equal to the number of processors.

(3) The number of kernel threads allocated to the program is greater than the number of processors but less than the number of user level threads.

(ii) Consider a system running ten I/O-bound tasks and one CPU-bound task. Assume that the I/O-bound tasks issue an I/O operation once for every millisecond of CPU computing and that each I/O operation takes 10 milliseconds to complete. Also assume that the context switching overhead is 0.1 millisecond and that all processes are long-running tasks. What is the CPU utilization for a round-robin scheduler when :

(1) The time quantum is 1 millisecond (3)

(2) The time quantum is 10 milliseconds (3)

(iii) What is a Process Control Block? Discuss. (4)

Or

(b) (i) Explain why interrupts are not appropriate for implementing synchronization primitives in multiprocessor systems. (4)

(ii) Servers can be designed to limit the number of open connections. For example, a server may wish to have only N socket connections at any point in time. As soon as N connections are made, the server will not accept another incoming connection until an existing connection is released. Explain how semaphores can be used by a server to limit the number of concurrent connections. (6)

(iii) How does the signal( ) operation associated with monitors differ from the corresponding operation defined for semaphores? (6)

13. (a) (i) State the critical section problem. Then, list and discuss the three requirements that a solution to the critical section problem must satisfy. (8)

(ii) Consider the deadlock situation that could occur in the dining-philosophers problem when the philosophers obtain the chopsticks one at a time. Discuss how the four necessary conditions for deadlock indeed hold in this setting. Discuss how deadlocks could be avoided by eliminating any one of the four conditions. (8)

Or

(b) (i) Given five memory partitions of 100 KB, 500 KB, 200 KB, 300 KB, and 600 KB (in order), how would each of the first-fit, best-fit, and worst-fit algorithms place processes of 212 KB, 417 KB, 112 KB, and 426 KB (in order)? Which algorithm makes the most efficient use of memory? (10)

(ii) Why are segmentation and paging sometimes combined into one scheme? (6)

14. (a) (i) What is the cause of thrashing? How does the system detect thrashing? Once it detects thrashing, what can the system do to eliminate this problem? (4)

(ii) Suppose that your replacement policy (in a paged system) is to examine each page regularly and to discard that page if it has not been used since the last examination. What would you gain and what would you lose by using this policy rather than LRU or second-chance replacement? (6)

(iii) Discuss the hardware support required to support demand paging. (6)

Or

(b) Most systems allow programs to allocate more memory to its address space during execution. Data allocated in the heap segments of programs is an example of such allocated memory. What is required to support dynamic memory allocation in the following schemes? Discuss. (16)

(i) Contiguous-memory allocation

(ii) Pure segmentation

(iii) Pure paging

15. (a) (i) What are the advantages of the variation of linked allocation that uses a FAT to chain together the blocks of a file? (6)

(ii) Some file systems allow disk storage to be allocated at different levels of granularity. For instance, a file system could allocate 4 KB of disk space as a single 4-KB block or as eight 512-byte blocks. How could we take advantage of this flexibility to improve performance? What modifications would have to be made to the free-space management scheme in order to support this feature? (6)

(iii) Discuss the relative advantages and disadvantages of sector sparing and sector slipping. (4)

Or

(b) (i) What are the various kinds of performance overheads associated with servicing an interrupt? (6)

(ii) The Linux scheduler implements soft real-time scheduling. What features are missing that are necessary for some real-time programming tasks? How might they be added to the kernel? (10)

———————