Questions

1	With a neat diagram, discuss the split view of the kernel in detail.	5M
2	Disuss the role of a device driver. How to load modules in the kernel subsystem.	5M
3	Show the implementation of loading the modules in a kernel.	3M
4	Discuss the commands to load and remove the modules in a kernel.	3M
5	Represent the read() and write() implementation in a scull enivornment.	5M
6	List and write any 10 fields in file structures of a scull environment.	5M
7	List and discuss the configuration options that should be enabled for kernels used for development during debugging process.	3M
8	Discuss the different loglevels used in printk() statement.	3M
9	Show the implementation of semaphore mechanism to avoid race condition while accessing the scull_dev data structures.	4M
10	Imagine for a moment that your driver acquires a spinlockand goes about its business within its critical section. Somewhere in the middle, your driver loses the processor. Perhaps it has called a function that puts the process to sleep. Or, perhaps, kernel preemption kicks in, and a higher-priority process pushesyour code aside. Your code is now holding a lock that it will not release any time inthe foreseeable future. If some other thread tries to obtain the same lock, it will, in the best case, wait (spinning in the processor) for a very long time. In the worst case,the system could deadlock entirely. Analyze the given scenario and discuss which technique is used to resolve it.	4M
11	Scullc is a cut-down version of the scull module that implements only the bare device. Unlike scull, which uses kmalloc, scullc uses memory caches. The size of the quantum can be modified at compile time and at load time, but not at runtime—that would require creating a new memory cache, and we didn't want to deal with these unneeded details. Illustrate the technique used to solve this scenario.	3M
12	Discuss the vmalloc() and kmalloc() function used in the Kernel subsystem.	3M
13	Draw a neat diagram for the peripheral interface for running digital I/O sample code on a computer.	2M
14	Discuss the short module to access I/O memory and ports.	2M
15	Discuss the portability issue in rewriting data type in the kernel.	3M
16	Describe the concept of logic programming model.	3M
17	Difference between parallel and distributed approach.	2M

18	Explain the inference mechanism in logic programming model.	3M
19	Explain the different functions used to access the process address space in kernel mode.	3M
20	Explain the relationship among the application program that invokes a system call.	3M
21	Write a short note on common file model.	4M
22	What is the role of virtual file system? Explain the three classes of filesystems supported by virtual file system.	4M