Q.1 What are the two main functions of an operating system ?

1. The control and management of the hardware components that make up the computer (input/output processes to and from the peripherals connected to the system)
2. Manage file storage and access. The programs can manage data storage on mass memory (obtaining complex structures, such as a database), using the procedures made available to the operating system. The component of the OS that takes care of all this is called the file system.

Q.2 What is multiprogramming?

* Without interrupts, it would be impossible to implement multiprogramming or timesharing. A busy-wait would be required so one job couldn’t run during another job’s I/O wait. Without a timer interrupt, time slices can’t be created to divide the CPU among jobs. Interrupts themselves must be synchronized.

Q.3 List some differences between personal computer operating systems and mainframe operating systems.

First, the hardware that they run on. Mainframes typically run on large boxes with many processors and tons of storage, as well as high-bandwidth busses. PCs are desktop or mobile devices with a single multi-core processor and typically less than 32GB of memory and a few TBs of disco space.

Q.4 What is the key difference between a trap and an interrupt?

* A trap is a special kind of interrupt which is commonly referred to as a software interrupt. An interrupt is a more general term which covers both hardware interrupts (interrupts from hardware devices) and software interrupts (interrupts from software, such as traps ).

Q.5 On early computers, every byte of data read or written was directly handled by the CPU (i.e.there was no DMA. What implications does this organization have for multiprogramming ?)

Some CPU time is consumed by the Operating System kernel's scheduler and dispatcher. It interrupts programs who use up their little time slice of CPU time. The scheduler keeps a list of programs that are ready (not still waiting for I/O to complete) and picks which one to dispatch to which core, based on priorities and other factors. It then interrupts what is running on that core and sets up the core's registers to resume running the to-be-dispatched program.

Q6. Which of the following instructions should be allowed only in kernel mode?

(a) Disable all interrupts.

(b) Read the time-of-day clock.

(c) Set the time-of-day dock.

(d) Change the memory map.

Q7. Can the

count = write(fd, buffer, nbytes);

call return any value in *count* other than *nbytes*? If so, why?

Q8. A file whose file descriptor is *fd* contains the following sequence of bytes: 3, 1, 4, 1, 5, 9, 2, 6, 5, 3, 5. The following system calls are made:

lseek(fd, 3, SEEK\_SET);

read(fd, &buffer, 4);

where the lseek call makes a seek to byte 3 of the file. What does *buffer* contain after the read has completed?

Q9.

A computer uses the relocation scheme of Fig. 1-9(a). A program is 10,000 bytes long and is loaded at address 40,000. What values do the *base* and *limit* register get according to the scheme described in the text?