

**10. Why do some systems store the operating system in firmware, while others store it on disk?**

Some devices which use OS which is not big in size use firmware while devices which use larger OS don't use firmware as firmware is expensive. The OS is stored on disk for such devices.

**11. How could a system be designed to allow a choice of operating systems from which to boot? What would the bootstrap program need to do?**

A system can be designed to allow booting from a choice of OS by modifying the bootstrap to read a boot block program which is configured with the information about the OS available for loading and then presenting this option to the user for selection. Once the user selects the option, the selected OS is loaded from the disk by the program at the boot block.

**12. The services and functions provided by an operating system can be divided into two main categories. Briefly describe the two categories, and discuss how they differ.**

The services provided by the OS can be divided into the following categories:

- i) Services to user: These services enable the user to interact with the system in a consistent and friendly way. Services such as the UI, File manipulation, I/O operations, process execution, inter-process communication and error detection, etc. enables the user to perform certain desired operations with ease.
- ii) Services for self: The OS tracks and performs certain tasks which ensure its smooth functioning and thus the functioning of the entire system. These include optimization of resource allocation, security and protection, auditing, etc.

These two services differ in the sense that the former are exposed to the user directly to make sure the user is able to do what is desired whereas the later ensure that the system performs efficiently and effectively and provide a smooth and uniform platform for the user services and also makes sure the use of any services on the OS is safe.

**13. Describe three general methods for passing parameters to the operating system.**

The three methods are as follows:

- i) Pass parameter in registers: Parameter values are stored directly in the registers from which the OS could read them. This however puts a limit on the number of parameters that can be passed to the OS as the number of registers is limited.
- ii) Store parameters in block: Parameters are stored in a block, a table or in memory and the address of the block is passed to the register. The OS reads the address from the registers and then reads parameters from the address.
- iii) Via stack: Parameters can be pushed on a stack by a program and can be read by the OS by popping these parameters.

**15. What are the five major activities of an operating system with regard to file management?**

The five major activities with regards to file management are:

- i) Create a file
- ii) Write a file
- iii) Read a file
- iv) Reposition a file(move)
- v) Delete a file.

--Ref. Ch 10

**16. What are the advantages and disadvantages of using the same system call interface for manipulating both files and devices?**

The advantages of using the same system call interface for both file and device is a uniform view/access of a similar functionality like read, write and reposition common across the two types.

The disadvantage is the monolithic design of such an approach. The file and device have some things in common but adding a functionality or even changing an existing implementation of only one type is difficult as the interface has to stay same. Every change has to be modelled to be reflected using the same interface. This can put on constraints on the type of new functionality that can be developed.

**17. Would it be possible for the user to develop a new command interpreter using the system-call interface provided by the operating system?**

Yes, it is possible to develop a new command interpreter using system-call interface provided by the OS. The interpreter would need to map the user input to a specific system call and would then pass the parameters to the system call. The output is displayed back on the command line console. This can be done easily by writing a program in a high level language like C++ to parse the user input and validate the constraints (if any) and then call the system routine.

**18. What are the two models of interprocess communication? What are the strengths and weaknesses of the two approaches?**

The two models are message-passing and shared memory model.

Strength:

Message Passing: It is easier to implement inter-computer communication using this model and is useful for exchanging small amounts of data.

Shared Memory: The data access is fast and two processes can communicate with each other without interference from the OS.

Weakness:

Message Passing: The sharing is slow as there are a number of steps involved for the handshake between processes. The amount of data shared is generally small.

Shared Memory: As both processes are accessing the same memory, the memory can be corrupted easily. Both processes can override the memory at the same time. This may lead to data-loss. A synchronization mechanism should be in place.

References: Chapter 2, Operating System Concepts, by Abraham Silberschatz, Peter B. Galvin, Greg Gagne