**COMPUTER FUNDAMENTALS**

**Brief Introduction**

1. Operating System is system software. It has 2 components shell and kernel
2. DOS - Disk Operating System
3. GUI-based Operating. Most companies are using command prompt-based operating systems because it's more secure.
4. MV is the command to move from one secure target to the destination target
5. Kernal is always running in the background. The time and date getting updated even after shutting down the device is an example of this.
6. Meta Data is the data providing information about one or more aspects of data
7. Log Data is each and every data that happened in the timeline
8. Metadata should be searched where OS is installed.
9. There are different flavors or extensions of operating systems. For example ubuntu and kali Linux, both are Linux os. Both have the same kernel means the back end is the same but different shell, and the font end is different.
10. . bash = bourne again shell

. csh

. TSH

11. Echo $0 helps to find the shell

12. Windows is a single-user os

Linux is a multi-user os

There are multiuser, multi-taker, and multi-processor

13. Process Management

System calls are created, load, execute, suspend, resume, kill, and store

Multiple processors in a main memory

Decision on sharing or deleting the permission/resources

14. I/O management

Hardware information

Share your inputs using the devices

Cache memory, buffer

15. File management

A file is a collection of data

We can delete, modify, open, and close

It can be stored from one drive to another

16. Network management

Keeping track of your network

Administrate your network

Maintenance

Security

Provisioning

Operation

**SCHEDULING**

Process scheduling is the activity of the process manager that handles the removal of the running process from the CPU and the selection of another process on the basis of a particular strategy. Process scheduling is an essential part of Multiprogramming operating systems. Such operating systems allow more than one process to be loaded into the executable memory at a time and the loaded process shares the CPU using time multiplexing.

There are two categories of scheduling:

* Non-preemptive: Here the resource can’t be taken from a process until the process completes execution. The switching of resources occurs when the running process terminates and moves to a waiting state.
* Preemptive: Here the OS allocates the resources to a process for a fixed amount of time. During resource allocation, the process switches from the running state to the ready state or from the waiting-for state to the ready state. This switching occurs as the CPU may give priority to other processes and replace the process with higher priority with the running process.

**Types of schedulers:**

* **Long-Term Scheduler**
  + It is a job scheduler
  + Speed is lesser than the short-term scheduler
  + It controls the degree of multiprogramming,
  + It is almost absent or minimal in the time-sharing system,
  + selects processes from the pool and loads them into memory for execution
* **Short-Term Scheduler**
  + It is a CPU scheduler,
  + Speed is the fastest among the other two,
  + It provides lesser control over the degree of multiprogramming,
  + It is also minimal in time-sharing systems,
  + It selects those processes which are ready to execute
* **Medium-Term Schedule**
  + It is a process-swapping scheduler,
  + Speed is in between both short and long-term schedulers,
  + It reduces the degree of multiprogramming,
  + It is a part of Time-sharing systems,
  + It can re-introduce the process into memory and execution can be continued.

**TYPES OF OPERATING SYSTEMS**

**What is an OS?**

An operating system is a well-organized collection of programs that manages computer hardware. It is a type of system software that is responsible for the smooth functioning of the computer system. The functions of the OS are:-

* **Processor Management**
* **Memory Management**
* **Device Management**
* **File Management**
* **Security**
* **Error Detection**
* **Job Scheduling**

**Components of the OS:-**

* **Shell**
* **Kernel**

**There are several different types of operating systems present. They are:-**

* **Batch OS:** Batch OS is the first operating system for second-generation computers. This OS does not directly interact with the computer. Instead, an operator takes up similar jobs and groups them together into a batch, and then these batches are executed one by one based on the first-come, first, serve principle.
* **Distributed OS:** A distributed OS is a recent advancement in the field of computer technology and is utilized all over the world that too with great pace. In a distributed OS, various computers are connected through a single communication channel. These independent computers have their memory unit and CPU and are known as loosely coupled systems. The system processes can be of different sizes and can perform different functions. The major benefit of such a type of operating system is that a user can access files that are not present on his system but in another connected system. In addition, remote access is available to the systems connected to this network.
* **Multitasking OS:** The multitasking OS is also known as the time-sharing operating system as each task is given some time so that all the tasks work efficiently. This system provides access to a large number of users, and each user gets the time of CPU as they get in a single system. The tasks performed are given by a single user or by different users. The time allotted to execute one task is called a quantum, and as soon as the time to execute one task is completed, the system switches over to another task.
* **Network OS:** Network operating systems are the systems that run on a server and manage all the networking functions. They allow sharing of various files, applications, printers, security, and other networking functions over a small network of computers like LAN or any other private network. In the network OS, all the users are aware of the configurations of every other user within the network, which is why network operating systems are also known as tightly coupled systems.
* **Real-OS:** Real-Time operating systems serve real-time systems. These operating systems are useful when many events occur in a short time or within certain deadlines, such as real-time simulations.
* **Mobile OS:** A mobile OS is an operating system for smartphones, tablets, and PDAs. It is a platform on which other applications can run on mobile devices.

**WHAT IS A SHELL**

Shell is an interface between the operating system and the user. The primary responsibility of the shell is to collect the user’s input and execute the program on the operating system based on that input. Then it displays the output obtained after the execution of the program.

## **Types of Shell**

The shell is mainly of two types, then these two types are further categorized; types of shell are

* Bourne Shell
  + Bourne shell (sh): Bourne shell is known as the first shell to be introduced, it is represented by “sh”. This shell got popular because of its quite compact nature.
  + Korn Shell (ksh): This shell was developed by David Korn in AT & T bells lab, this was introduced as an improved version or superset of the Bourne shell. It is represented by “ksh”. It has all the features and functionalities of Bourne Shell and also provides some new functionalities to the users.
  + Bourne Again shell (bash): It is also known as Bash Shell, This shell combines features of the Korn shell and C shell. This shell was designed as an extended version of the Bourne shell. Bourne Again Shell can automatically load previously used commands and can be edited with the help of the arrow keys of the keyboard.
* C Shell: The C shell was designed by Bill Joy at the University of California. It is represented using “csh”. The C shell was designed with the purpose of supporting programming languages. It was specifically designed to support in-built features like solving arithmetic operations and syntax of programming languages like C. Unlike Bourne and other Linux shells, the C shell can maintain and history of previously used commands, and those commands can be used whenever required.

**STACK AND HEAP (PURPOSE AND HOW ITS STORED)**

**STACK:** A stack is a special area of a computer’s memory that stores temporary variables created by a function. In stack, variables are declared, stored, and initialized during runtime.

It is a temporary storage memory. When the computing task is complete, the memory of the variable will be automatically erased. The stack section mostly contains methods, local variables, and reference variables.

The purpose is :

* A Stack can be used for evaluating expressions consisting of operands and operators.
* Stacks can be used for Backtracking, i.e., to check parenthesis matching in an expression.
* It can also be used to convert one form of expression to another form.
* It can be used for systematic Memory Management.

Operations:

**PUSH:** PUSH operation implies the insertion of a new element into a Stack.

**POP:** POP means to delete an element from the Stack. Before deleting an element, make sure to check if the Stack Top is NULL, i.e., TOP=NULL. If this condition goes true, it means the Stack is empty, no deletion operation can be performed, and even if we try to delete, then the Stack underflow message will be generated.

**HEAP:** The heap is a memory used by programming languages to store global variables. By default, all global variables are stored in heap memory space. It supports Dynamic memory allocation.

The heap is not managed automatically for you and is not as tightly managed by the CPU. It is more like a free-floating region of memory.

The purpose is:

* Heap helps you to find the greatest and minimum number
* Garbage collection runs on the heap memory to free the memory used by the object.
* The heap method is also used in the Priority Queue.
* It allows you to access variables globally.
* Heap doesn’t have any limit on memory size.

Operations:-

* **Heapify:** a process of creating a heap from an array.
* **Insertion:** process to insert an element in existing heap time complexity O(log N).
* **Deletion:** deleting the top element of the heap or the highest priority element, and then organizing the heap and returning the element with time complexity O(log N).
* **Peek:** to check or find the most prior element in the heap, (max or min element for max and min heap).