**What is an operating system?**

FUNCTIONS - DEVICE MANAGEMENT (The processes may require devices for their use. This management is done by the OS). FILE MANAGEMENT (The files on a system are stored in different directories). MEMORY MANAGEMENT (It is the management of the main or primary memory. Whatever program is executed, it has to be present in the main memory and there can be more than one program present at a time and it is required to manage the memory). SECURITY (The OS keeps the system and programs safe and secure through authentication). OTHER FUNCTIONS (Some other functions of the OS, like Error detection a.o.). PROCESSOR MANAGEMENT/SCHEDULING (When more than one process runs on the system the OS decides how and when a process will use the CPU. Hence, the name is also CPU Scheduling). NETWORK MANAGEMENT (

**1 - What is an operating system - Operating System API?**

What is operating system operating system API? Operating system is middleware between user and hardware. It provides abstraction and arbitration. OS is a software program that manage and control a computer, hardware and provide services for computer program. Helps hardware and software application interact with each other. API application program interface. Low level programming languages will use to communicate with hardware. Each OS has its own API. API is a set of rules or tools that allow different software applications to communicate with each other. It defines the methods and data formats that applications can use to repeat or exchange information. API enables different software systems to work together, allowing them to share and utilize each other functionalities without to need developers to understand working of systems.

**2 - What is an operating system - Abstraction vs Arbitration?**

Abstraction and arbitration. Abstraction involves simplifying complex ideas or systems by processing on essential characteristics and ignoring unnecessary details. Arbitration - Operating system managed access to shared hardware resources. So that multiple application can run on the same hardware at the same time, without interfering with one another. It's a process of resolving conflicts by having neutral 3rd part decision. Security and isolation between different applications in the access of resources.

**3 - What is an operating system - Concurrency vs Parallelism?**

Concurrency versus parallelism. We have single core CPU and we running multiple application at the same time. Each process is executing concurrently without waiting one process finish before we get another one. We can prioritize different processes. Concurrency - 1 worker does multiple jobs. Parallelism - multiple workers do multiple jobs. Prioritize it`s advantage for concurrency. Parallelism doing multiple tasks at the exact same time, often with the help of multiple processors or cores. It is like having multiple hands actively working on different things simultaneously.

**23 - What is an Operating System. OS design goals?**

Convenience - Abstraction of hardware resource for user programs.

Efficiency - Utilize CPU at max.

Isolation - Between multiple processes, security. Ease of use - Provide a user-friendly interface that is easy to understand and navigate, making the system accessible to both novice and experienced users.

Reliability - Build a robust and reliable system that minimize the likelihood of failures, crushes or unexpected errors. This is crucial for maintaining system availability.

Portability - Design operating system to be easily adaptable and runnable on different hardware platforms with minimal modification. This enhances flexibility and facilitates migration between systems.

Security - Implement security measures to protect the system, user data and applications from unauthorized access, malicious activities and potential vulnerabilities.

**24 - Operating System products. Mobile operating systems.**

A mobile operating system allows the user to run other different application software on the mobile, tablets, etc. Moreover, we can say that it is a type of operating system which is specially designed for mobiles, tablets, smartwatches, etc. Furthermore, they are a mixture of computer OS with some additional features for mobiles. Also, they are comparatively light and simple. An operating system (OS) is a program that acts as an interface between the system hardware and the user. Moreover, it handles all the interactions between the software and the hardware. Before knowing different mobile OS, let us study some functions of an operating system.

The Android OS - is the most common operating system among the mobile operating system. Furthermore, Google is the developer of Android. Moreover, it is an open source and free operating system. This OS is based on the Linux kernel. The name for every new version of update is based on desserts.

IOS - After android, it is one of the most popular OS. It is designed to run on Apple devices such as iPhones, iPad etc. Moreover, like the android devices have the play store for apps download. Likewise, apple iOS contains the app store.

BADA - Samsung is the launcher of this operating system. It came into market in 2010. Moreover, it includes features like 3-D graphics, application installation, multipoint touch etc.

**8 - What is an Operating System. OS hardware/software components.**

**Hardware Components**: Central Processing Unit (CPU): The CPU is the primary hardware component responsible for executing instructions of a computer program. The OS interacts with the CPU to schedule and manage processes.

*Memory (RAM)*: Random Access Memory (RAM) is used by the OS to store data and programs that are actively being used or processed.

*Storage Devices:* The OS manages storage devices, such as hard drives and SSDs, for long-term data storage. It handles file systems, data retrieval, and storage allocation.

*Input Devices*: Devices like keyboards, mice, and touchscreens are managed by the OS to receive input from users.

*Output Devices*: The OS controls output devices like monitors and printers to display information and produce tangible results.

*Network Interface*: The OS interacts with network interfaces to enable communication between the computer and other devices on a network.

*Peripheral Devices:* OS manages communication with various peripherals like USB devices, cameras, scanners, etc.

**Software Components**:

*Kernel*: The kernel is the core part of the OS that interacts directly with the hardware. It manages system resources, provides essential services, and handles communication between hardware and software.

*Device Drivers*: These are software modules that allow the OS to communicate with and control specific hardware devices.

*File System:* The file system organizes and manages the storage of data on storage devices. It provides a hierarchical structure for files and directories.

*User Interface:* The user interface enables interaction between users and the computer. It can be a command-line interface (CLI) or a graphical user interface (GUI).

*Utilities:* Utilities are specialized programs that perform specific tasks, such as disk cleanup, antivirus scanning, and system maintenance.

**17 - What is an operating system - Isolation and Security?**

Isolation - isolation of operating system involves keeping different processes or users independent encrypted by each other. It ensures that errors in one part of the system don't impact other parts.

Security - Security in operating system protecting system, its data and its resources from unauthorized access, malicious attacks, other potential threats. Secure the safeguards intentional unintentional threats. Ensure confidently, integrity, availability of data and resources.

**5 - CPU Timesharing. Operating System scheduling policies.**

CPU Timesharing, or multitasking, is a technique that allows multiple processes to share a single CPU by dividing its available time. Each process is allocated a small time slice to execute, creating the illusion of simultaneous execution for multiple tasks on a single processor. This enhances overall system efficiency and responsiveness.

Process is running program in memory. Operating system will distribute CPU resources between processes and it will do it efficiently and secure.

CPU scheduler - policy to choose which process to run, mechanism for context switch.

Scheduling policies:

First come, First served. First come First serve is a type of scheduling algorithm used by operating system and networks efficiently and automatically execute by the order of their arrival. The convoy effect in the FCFS If the CPU acquires the processes with higher burst time at the front end of the ready queue then those processes which have lower burst time may get blocked.

Shortest Job Next is the scheduling policy that selects the waiting process with the smallest execution time. It gives maximum throughput and minimum average waiting time.

Priority based scheduling. Method of scheduling based on priority. The process with the higher priority carried out first. There is one disadvantage of priority is starvation. Lower priority never gets a chance to execute because higher priority process continuously occupies the CPU. For solving this problem (starvation) we just increase priority by 1 point of each process after cycle.

Shortest Remaining Time. That is preemptive version of shortage job next scheduling. Amount of time remaining until completion is selected to execute.

Round Robin when the processes are given turns at running one after the other in a repeating sequence and each one is preempted when it has used up its time slice. That slice called quantum.

Multiple level queues. It's designed to dynamically adjust the priority of a process based on its waiting time, keeping process with the short response times given higher priority than those with long response time.

Context switch - when we switch, process one to another.

**9 - Operating system types. Single vs Multitask. Real-time OS. Embedded OS.**

Single-tasking Operating System: (Handles one task at a time. Examples: MS-DOS).

Multitasking Operating System: (Manages multiple tasks concurrently. Examples: Windows, Linux, macOS).

Real-time Operating System (RTOS): (Responds to events within a specific time frame. Critical for applications with strict timing requirements. Examples: VxWorks, FreeRTOS).

Embedded Operating System: (Designed for specific hardware and applications. Compact and efficient. Examples: Embedded Linux, FreeRTOS for embedded systems).

**6 - Operating System calls. POSIX standard.**

System calls is a function that provided by operating system to our program in order to communicate with the hardware components. For example: To read data from hard disk/Reserve memory from RAM. We have standard API sets that is collection of the functions.

POSIX (Portable Operating System Interface) API sets are implemented by Linux operating system. If operating system is POSIX compliant, it means it has predefined functions to work with the hardware components. Fork () function is predefined for POSIX, but Windows doesn't use because it isn't POSIX compliant.

Functions for Process Control:

fork (): Creates a new process.

exec (): Executes a new program in the current process.

wait (): Waits for a child process to exit.

exit (): Exits the current process.

Programming languages libraries provides abstraction over system calls. Users don't invoke system calls directly.

**18 - DOS operating system.**

DOS operating system is a common line operating system, operating system that runs from a disk drive.

1) The ROM bootstrap loader reads the Master Boot Record and passes control over it.

2) The boot record loads the disk operating system into memory and it takes control of the machine.

3) The computer transferred data stored on a magnetic disk to its main memory. The Random-Access Memory land to external device attached to the computer, such as computer screen on printer. The computer provides various application programming interfaces for programs like I/O, memory management, program loading and termination. The operating system also provides file management that organizes, reads and writes file on storage. The files organized in a hierarchical structure of directors, subdirectors and files. computer, such as computer screen on printer. The computer provides various application programming interfaces for programs like I/O, memory management, program loading and termination. The operating system also provides file management that organizes, reads and writes file on storage. The files organized in a hierarchical structure of directors, subdirectors and files.

**12 -** **Process Management. Process states and lifecycle.**

Process has unique identifiers PID. In task manager we can check PID for each process. Hip is memory. Bullet open that's bad. Stack is for holding function called stack. Heap is memory. Process states: (*Running* - currently executing), (*Ready* - waiting for execution/be scheduled), (*Block* - waiting for event (Reading data from disk) (If it takes some time to take data when we waiting data to be downloaded from Internet the process goes to Block state, if it happens so fast it directly goes to ready state), (*New state* - Being created), (*Terminate* - it when process terminated or killed)).

Scheduling policy - if we have policy like each time it's executed equal amount of time when execution time goes beyond the Internet time, it will go to block state.

Example: Error happened, Ram overloaded, Malicious activity.

Process lifecycle:

When our process is running and it needs to retrieve some data from hard disk, then operating system will put it to block state and when data is arrived, it will go to Ready state. It can be happened without Block state.

**13 - Process list. Process components. Process control block (PCB).**

When we open task manager, it will show us list of processes. Operating system is responsible for maintaining processes. Process control block is unit of information about each process and holds about each of them.

PCB: (PID, State - state of the process, CPU context -> Program counter - the line of the code that had been executed, Pointers (to the memory, to the input output devices, files, Relation with the other processes)

Each process has its child process. When I open terminal (CMD) and execute Ls function, it will have own id and ls function will be child of terminal.

Memory Image (Code, data, stack, heap) are allocated by operating system.

CPU context initialized, then PC (Program Counter) is set to the first instruction.

OS is out and process is executed by CPU.

Function call translates to a jump instruction and a new frame pushed onto the stack and SP (Stack Pointer) is updated.

CPU Privileged modes - User Mode (Regular calls), Kernel Mode (System Calls).

Kernel has its own stack.

Kernel mode has interrupt Descriptor Table - it holds jump addresses of system calls.

**4 – 10 – Virtualization vs Containerization. Hypervisor types.**

*Virtualization* - Creating one operating system inside another. Operating system create useful services using resource that are traditionally band to hardware. It allows you to use a physical machine`s full capacity by distributing it capabilities among many users.

Utilization is about reserving resources and providing to virtual machine completely isolation and security, we can manage our resources efficiently.

Technology – Hypervisor

Isolation - Each virtual machine is an isolated environment with its own operating system and applications.

*Hypervisor* - is software or firmware layer that enables multiple operating systems to share single physical hardware Host.

Hypervisors has two types:

Type 1 - Has directly installed to hardware (Example: pare metal ESXi)

Let's say we have a huge server. With the help of ESXI, we can install different kind of operating system to our server.

Type 2 - installed to the OS (Example: VM Ware, Virtual box).

We installed software like VirtualBox and we install operating system with a shared memory. If we have windows and also installed 2 Linux and 1 more window. Each of them will have our kernel and memory and none of them will have information about the other.

*Containerization* - There is one host kernel that each of them shared between different containers. Virtualization is more secure. Containerization is comfortable for working and easy for work with memory.

Technology – Docker

Isolation: Containers provide resource and dependency isolation, but they share the host OS kernel.

**11 - Windows 11 new features. Differences with previous versions.**

The main feature that Windows 11 has opportunities to run Android applications. So, this function introduced with Windows 11. You can download Android application natively only in Windows 11.

The Other: Voice typing, Multiple Desktops (CTRL + WIN + ARROW Keys), Windows snapping, Windows widgets, build in Microsoft Teams, Live captions (CTRL + WIN + L), Snipping tool, Paint features (Layers and AI co-creator), Gaming Improvements, Improved Microsoft Edge Browser.

Build in COPILOT - is a tool developed by OpenAI in collaboration with GitHub. It's designed to assist developers by providing code suggestions and autocompletions using natural language descriptions.

**15 - What is Cloud Computing.**

Cloud computing is like renting a computer or software over the internet. Instead of owning and maintaining your own computer servers, you can use someone else's servers to store data, run programs, or perform tasks. It's like paying for what you need, when you need it, without the hassle of managing the hardware yourself.

Examples: Cloud Storage (Google Drive, DropBox).

Advantages: Cost efficiency, Accessibility (We can get access to our application and our data from anywhere).

Disadvantages: Security concerns, Dependence on Internet connection.

**16 - Linux in Windows - WSL.**

Windows Subsystem for Linux (WSL) is a compatibility layer for running Linux binary executables natively on Windows. It enables developers to use a Linux distribution directly within a Windows environment without the need for a virtual machine.

**19 - Windows Command Line - File system, navigation.**

cd - cd working directory

cd.. - go one level up

mkdir - create new directory

dir - list contents of working directory

move - move folder/file

copy - copy folder/file

rename - rename files

rmdir - remove directory (add /s option to remove non-empty folder)

D: - change current drive (by typing its letter)

**20 - Volatile vs Non-Volatile memory**

Volatile memory - is a type of computer memory that loses its stored information when the power is turned off. It requires a constant power supply to retain data. Examples: RAM (Random-Access Memory), Cache Memory, Register Memory.

Non-volatile memory - is a type of computer memory that retains its stored information even when the power is turned off. It does not require a constant power supply to preserve data. Example: ROM (Read-Only Memory), Flash Memory (used in USB drives, memory cards, SSDs), Hard Disk Drives (HDDs), Optical Drives.

SRAM - is a type of volatile semiconductor memory that uses flip-flop circuits to store data. It retains data without the need for constant refreshing. SRAM is commonly used in cache memory for processors.

DRAM - is another type of computer memory. It needs constant refreshing to keep information, and it's a bit slower than SRAM.

**21. Linux command line package management utilities.**

*Install package*

dnf install package\_name - *(centos)*

yum install package\_name - *(centos, dnf is preferred)*

apt-get install package\_name - *(ubuntu)*

*User management*

cat /etc/passwd - *list users*

useradd user1 - *create new user*

adduser user1 - *adduser is high level and preferred*

userdel user1 - *delete user*

cat /etc/group - *list groups*

groupadd group1 - *add new group*

groupdel group1 - *remove group*

passwd user1 - *set/change user password*

groups user1 - *view groups of user1*

usermod [options] user1 - *modify user*

usermod -a -G group\_name username - *add user to additional groups*

yum install sudo -y - *install sudo or apt install sudo for debian/ubuntu*

cat /etc/sudoers - *view sudoers, to make user sudo we add user to wheel group or directly modify /etc/sudoers file*

su -l user1 - *login as user1, change the current user to user1*

sudo <cmd> - *execute command with super privileges, commands like shutdown, poweroff, reboot require sudo*

*Permission management*

*Permission levels in Linux: u – User (Owner), g – Group, o – Others*

*Permission type in Linux: r – read (4), w – write (2), e - execute (1)*

chmod a+rw file1 - *give all user read/write access*

chmod a-rw file1 - *revoke read write access from all users*

chmod 775 file1 - *give owner full, group full and other read/write access*

chmod -r 775 /folder1/\* - *set 755 permissions to all files inside folder*

chown user: group file1 - *change file owner*

**7 - RAM generations - DDR1, DDR2, DDR3, DDR4.**

In order for program to execute first, it needs to load into memory RAM. After data is loaded into Ram CPU can access this data. When Ram is full, some space from HDD will be used and it will slow down PC. This is called swap.

DDR - double data rate: Rising and falling edges of clock signals to send data. DDR may include both the clock speed and the total bandwidth in its name. It has 333 megahertz and 8 bytes. So, 333\*8=2700 MB/sec. Has 184 pins.

DDR2: 6400 MB/sec is Twice faster than previous. Has 240 Pins.

DDR3: 12800 MB/sec is Twice faster than previous. Has 240 Pins. Bottom notches are in different places.

DDR4: 34100 MB/sec. Has 288 Pins.

DDR5: 51200 MB/sec.

**7 - RAM 64bit vs 32bit. SRAM and DRAM. RIMM.**

*RIMM* - is Rambus Inline Memory Module. It was debuted in 1999 and was dropped out due to technology advance in DIMMS. It has only 2-byte wide bus and 800 MHz. 800 MHz \* 2 Bytes = 1600 MB/sec

*SRAM* - Data is stored in transistors and requires a constant power flow. Because of the continuous power, SRAM doesn’t need to be refreshed to remember the data being stored. SRAM is called static as no change or action i.e. refreshing is not needed to keep the data intact. It is used in cache memories.

*DRAM* - Data is stored in capacitors. Capacitors that store data in DRAM gradually discharge energy, no energy means the data has been lost. So, a periodic refresh of power is required in order to function. DRAM is called dynamic as constant change or action (change is continuously happening) i.e. refreshing is needed to keep the data intact.

The process of updating data in DRAM is associated with the peculiarities of storing information in memory of this type. In DRAM, each bit of data is stored in a capacitance (capacitor), and this capacitance tends to lose charge over time due to leaks.

To store data in DRAM, capacities need to be refreshed periodically. The update process involves recharging the capacity to the maximum level to prevent data loss.

*64 bits vs 32 bits RAM* - refer to the width of the data bus, which is the number of bits that can be transferred in parallel.

32-bit RAM: Can address up to 4 gigabytes (2^32 bytes) of memory.

64-bit RAM: Can theoretically address up to 18.4 million terabytes (2^64 bytes) of memory.

**25 - ECC RAM vs non-ECC RAM.**

*ECC* - Error correcting code: Make sure that the data is correctly processed by memory, it prevents Memory data corruption (damage, повреждение). ECC RAM will have 9 memory chips instead of 8. Most of Ram are non-ECC due to advance of technology.

*ECC RAM:* ECC RAM includes additional bits for error correction. These extra bits allow the RAM module to detect and correct certain types of memory errors. If a single-bit error is detected, ECC RAM can correct it on the fly. Ifmultiple errors occur in the same data word, ECC RAM can detect the issue but may not be able to correct it completely.

*Non-ECC RAM:* Non-ECC RAM lacks error-correcting capabilities. Any errors that occur in the data stored in non-ECC RAM are detected, but the system doesn't have the ability to automatically correct them.

Difference:

1) ECC RAM is more expensive than non-ECC RAM

2) The error correction process in ECC RAM can introduce a slight performance overhead compared to non-ECC RAM. However, this impact is often negligible for many applications.

**22 - DRAM vs SDRAM.**

*DRAM* - Data is stored in capacitors. Capacitors that store data in DRAM gradually discharge energy, no energy means the data has been lost. So, a periodic refresh of power is required in order to function. DRAM is called dynamic as constant change or action (change is continuously happening) i.e. refreshing is needed to keep the data intact.

*DRAM*: DRAM is a type of memory that stores each bit of data in a separate capacitor within an integrated circuit. It requires constant refreshing to maintain the charge on the capacitors.

*SDRAM (Synchronous Dynamic Random-Access Memory)*: SDRAM is an evolved version of DRAM that synchronizes its operations with the system clock. This means that data access and operations are coordinated with specific phases of the clock signal, such as rising or falling edges. Events like the start of data read or write occur at specific moments relative to the system clock signal.

*Data Locking:*

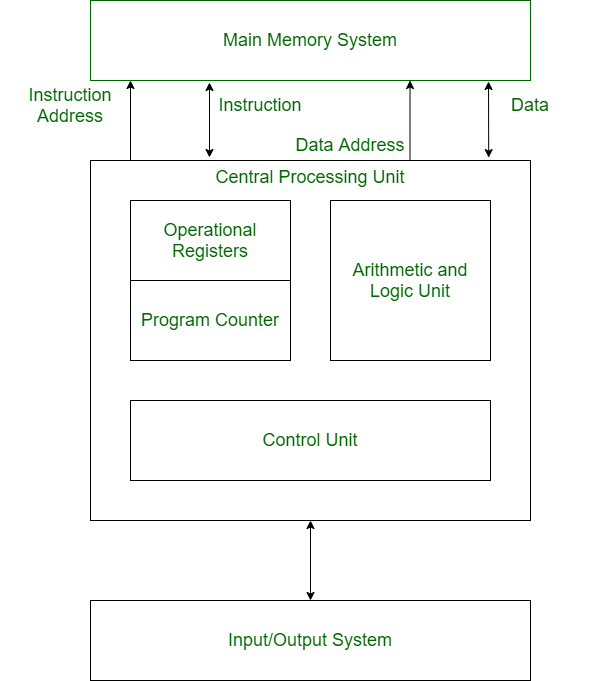
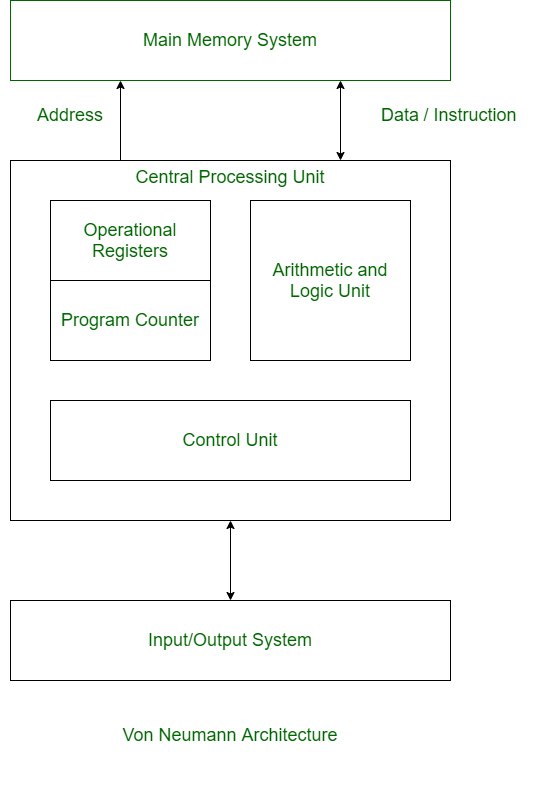
Read and write operations in SDRAM may occur at specific clock signal intervals. The system needs to be configured to expect data access at these specific moments to interact correctly with SDRAM.

**14 - Von Neumann Architecture. CPU, RAM, I/O devices.**

*VNA* - is a computer architecture model that forms the basis for most modern computers. The key components of the von Neumann architecture include the CPU (Central Processing Unit), RAM (Random Access Memory), and I/O (Input/Output) devices.

*Control Bus and Address/Data Bus:* The von Neumann architecture utilizes buses for communication between different components. The control bus carries signals that control the operations of the CPU, memory, and I/O devices.

The address bus is used to specify the memory address for read or write operations, and the data bus carries the actual data between the CPU, memory, and I/O devices.



*Difference:*

*In VNA* - Same physical memory address is used for instructions and data and There is common bus for data and instruction transfer.

*In Harvard* - Separate physical memory address is used for instructions and data; also, Separate buses are used for transferring data and instruction.