**What's OPERATING SYSTEM?**An Operating System, or OS, is a software program that enables the computer hardware to communicate and operate with the computer software. Without a computer Operating System, a computer would be useless.

**OPERATING SYSTEM TYPES**   
As computers have progressed and developed so have the types of operating systems. Below is a basic list of the different types of operating systems and a few examples of Operating Systems that fall into each of the categories. Many computer Operating Systems will fall into more than one of the below categories.   
  
**GUI** - Short for Graphical User Interface, a GUI Operating System contains graphics and icons and is commonly navigated by using a computer mouse. See our GUI dictionary definition for a complete definition. Below are some examples of GUI Operating Systems.   
  
System 7.x  
Windows 98  
Windows CE   
  
**Multi-user** - A multi-user Operating System allows for multiple users to use the same computer at the same time and/or different times. See our multi-user dictionary definition for a complete definition for a complete definition. Below are some examples of multi-user Operating Systems.   
  
Linux  
Unix  
Windows 2000  
Windows XP  
Mac OS X   
  
**Multiprocessing** - An Operating System capable of supporting and utilizing more than one computer processor. Below are some examples of multiprocessing Operating Systems.   
  
Linux  
Unix  
Windows 2000  
Windows XP  
Mac OS X   
  
**Multitasking** - An Operating system that is capable of allowing multiple software processes to run at the same time. Below are some examples of multitasking Operating Systems.   
  
Unix  
Windows 2000  
Windows XP  
Mac OS X   
  
**Multithreading** - Operating systems that allow different parts of a software program to run concurrently. Operating systems that would fall into this category are:   
  
Linux  
Unix  
Windows 2000  
Windows XP  
Mac OS X

**What are the basic functions of an operating system?**

- Operating system controls and coordinates the use of the hardware among the various applications programs for various uses. Operating system acts as resource allocator and manager. Since there are many possibly conflicting requests for resources the operating system must decide which requests are allocated resources to operating the computer system efficiently and fairly. Also operating system is control program which controls the user programs to prevent errors and improper use of the computer. It is especially concerned with the operation and control of I/O devices.

**Why paging is used?**   
-Paging is solution to external fragmentation problem which is to permit the logical address space of a process to be noncontiguous, thus allowing a process to be allocating physical memory wherever the latter is available.

**While running DOS on a PC, which command would be used to duplicate the entire diskette?**   
diskcopy

**What resources are used when a thread created? How do they differ from those when a process is created?**   
When a thread is created the threads does not require any new resources to execute the thread shares the resources like memory of the process to which they belong to. The benefit of code sharing is that it allows an application to have several different threads of activity all within the same address space. Whereas if a new process creation is very heavyweight because it always requires new address space to be created and even if they share the memory then the inter process communication is expensive when compared to the communication between the threads.

**What is virtual memory?**   
Virtual memory is hardware technique where the system appears to have more memory that it actually does. This is done by time-sharing, the physical memory and storage parts of the memory one disk when they are not actively being used.

**What is Throughput, Turnaround time, waiting time and Response time?**   
Throughput – number of processes that complete their execution per time unit. Turnaround time – amount of time to execute a particular process. Waiting time – amount of time a process has been waiting in the ready queue. Response time – amount of time it takes from when a request was submitted until the first response is produced, not output (for time-sharing environment).

**What is the state of the processor, when a process is waiting for some event to occur?**   
Waiting state

**What is the important aspect of a real-time system or Mission Critical Systems?**   
A real time operating system has well defined fixed time constraints. Process must be done within the defined constraints or the system will fail. An example is the operating system for a flight control computer or an advanced jet airplane. Often used as a control device in a dedicated application such as controlling scientific experiments, medical imaging systems, industrial control systems, and some display systems. Real-Time systems may be either hard or soft real-time. Hard real-time: Secondary storage limited or absent, data stored in short term memory, or read-only memory (ROM), Conflicts with time-sharing systems, not supported by general-purpose operating systems. Soft real-time: Limited utility in industrial control of robotics, Useful in applications (multimedia, virtual reality) requiring advanced operating-system features.

**What is the difference between Hard and Soft real-time systems?**- A hard real-time system guarantees that critical tasks complete on time. This goal requires that all delays in the system be bounded from the retrieval of the stored data to the time that it takes the operating system to finish any request made of it. A soft real time system where a critical real-time task gets priority over other tasks and retains that priority until it completes. As in hard real time systems kernel delays need to be bounded

**What is the cause of thrashing? How does the system detect thrashing?**   
Once it detects thrashing, what can the system do to eliminate this problem? - Thrashing is caused by under allocation of the minimum number of pages required by a process, forcing it to continuously page fault. The system can detect thrashing by evaluating the level of CPU utilization as compared to the level of multiprogramming. It can be eliminated by reducing the level of multiprogramming.

**What is multi tasking, multi programming, multi threading?**Multi programming: Multiprogramming is the technique of running several programs at a time using timesharing. It allows a computer to do several things at the same time. Multiprogramming creates logical parallelism. The concept of multiprogramming is that the operating system keeps several jobs in memory simultaneously. The operating system selects a job from the job pool and starts executing a job, when that job needs to wait for any i/o operations the CPU is switched to another job. So the main idea here is that the CPU is never idle. Multi tasking: Multitasking is the logical extension of multiprogramming .The concept of multitasking is quite similar to multiprogramming but difference is that the switching between jobs occurs so frequently that the users can interact with each program while it is running. This concept is also known as time-sharing systems. A time-shared operating system uses CPU scheduling and multiprogramming to provide each user with a small portion of time-shared system. Multi threading: An application typically is implemented as a separate process with several threads of control. In some situations a single application may be required to perform several similar tasks for example a web server accepts client requests for web pages, images, sound, and so forth. A busy web server may have several of clients concurrently accessing it. If the web server ran as a traditional single-threaded process, it would be able to service only one client at a time. The amount of time that a client might have to wait for its request to be serviced could be enormous. So it is efficient to have one process that contains multiple threads to serve the same purpose. This approach would multithread the web-server process, the server would create a separate thread that would listen for client requests when a request was made rather than creating another process it would create another thread to service the request. To get the advantages like responsiveness, Resource sharing economy and utilization of multiprocessor architectures multithreading concept can be used.

**What is hard disk and what is its purpose?** Hard disk is the secondary storage device, which holds the data in bulk, and it holds the data on the magnetic medium of the disk. Hard disks have a hard platter that holds the magnetic medium, the magnetic medium can be easily erased and rewritten, and a typical desktop machine will have a hard disk with a capacity of between 10 and 40 gigabytes. Data is stored onto the disk in the form of files.

**What is fragmentation? Different types of fragmentation?**

- Fragmentation occurs in a dynamic memory allocation system when many of the free blocks are too small to satisfy any request. External Fragmentation: External Fragmentation happens when a dynamic memory allocation algorithm allocates some memory and a small piece is left over that cannot be effectively used. If too much external fragmentation occurs, the amount of usable memory is drastically reduced. Total memory space exists to satisfy a request, but it is not contiguous. Internal Fragmentation: Internal fragmentation is the space wasted inside of allocated memory blocks because of restriction on the allowed sizes of allocated blocks. Allocated memory may be slightly larger than requested memory; this size difference is memory internal to a partition, but not being used

**What is DRAM? In which form does it store data?**   
- DRAM is not the best, but it’s cheap, does the job, and is available almost everywhere you look. DRAM data resides in a cell made of a capacitor and a transistor. The capacitor tends to lose data unless it’s recharged every couple of milliseconds, and this recharging tends to slow down the performance of DRAM compared to speedier RAM types.

**What is Dispatcher?**   
- Dispatcher module gives control of the CPU to the process selected by the short-term scheduler; this involves: Switching context, Switching to user mode, Jumping to the proper location in the user program to restart that program, dispatch latency – time it takes for the dispatcher to stop one process and start another running.

**What is CPU Scheduler?**   
- Selects from among the processes in memory that are ready to execute, and allocates the CPU to one of them. CPU scheduling decisions may take place when a process: 1.Switches from running to waiting state. 2.Switches from running to ready state. 3.Switches from waiting to ready. 4.Terminates. Scheduling under 1 and 4 is non-preemptive. All other scheduling is preemptive.

**What is Context Switch?**   
- Switching the CPU to another process requires saving the state of the old process and loading the saved state for the new process. This task is known as a context switch. Context-switch time is pure overhead, because the system does no useful work while switching. Its speed varies from machine to machine, depending on the memory speed, the number of registers which must be copied, the existed of special instructions(such as a single instruction to load or store all registers).

**What is cache memory?**   
- Cache memory is random access memory (RAM) that a computer microprocessor can access more quickly than it can access regular RAM. As the microprocessor processes data, it looks first in the cache memory and if it finds the data there (from a previous reading of data), it does not have to do the more time-consuming reading of data from larger memory.

**What is a Safe State and what is its use in deadlock avoidance?**   
- When a process requests an available resource, system must decide if immediate allocation leaves the system in a safe state. System is in safe state if there exists a safe sequence of all processes. Deadlock Avoidance: ensure that a system will never enter an unsafe state.

**What is a Real-Time System?**   
- A real time process is a process that must respond to the events within a certain time period. A real time operating system is an operating system that can run real time processes successfully