1.4 Keeping in mind the various definitions of operating system, consider whether the operating system should include applications such as web browsers and mail programs. Argue both that it should and that it should not, and support your answers.

Operating system can be defined as the set of softwares that a vendor ships when one orders an Operating System. This sometimes can include Web Browsers and Mail programs. This definitions inclines towards the argument that web browsers and mail programs should be a part of the Operating System. This has been a popular choice since Windows, when Microsoft bundled Internet Explorer along with its Operating system. This trend has been carried along with android and iOS devices. In fact, the argument gains more strength in todays market, where data is gathered and transferred between applications and used in the Operating system. For iOS devices, the updates for the mail and safari roll out with the iOS updates. And these applications are used in a lot of the system calls to gather and redirect data. Hence, they become a necessary part of the operating system.

Operating system can also be defined as the program which is always running in a computer system. It is a control program that that manages the execution of user programs and allocation of resources to prevent errors and improper use of the computer. When these definitions are taken into account, web browsers and mail fail to satisfy the criteria. Both mail and web browsers are application programs which can be stopped from running without affecting the functioning of the Computer system. There are many operating systems which don't come with these applications preinstalled and still continue to function properly. Additionally, neither of these two manages the execution of other user programs or allocate resources. Browsers and mails are not an integral part of the operating system.

1.14 Under what circumstances would a user be better off using a time-sharing system than a PC or a single-user workstation?

A time-sharing system is an extension of the multiprogramming feature of the operating system. This ensures that the resources are efficiently used and the CPU doesn't stay idle.

The main advantage a time-sharing system provides a user over a single-user workstation/PC is provided by the Virtual memory technique. This enables the user to run a program that is larger than the actual physical memory. The use of the virtual memory technique allows the OS to execute a process that is not completely in the memory. This technique also separates the logical memory as viewed by the user from the physical memory. Hence, a user would be better off using a time-sharing system, when he has concern form memory-storage limitations.

1.19 What is the purpose of interrupts? How does an interrupt differ from a trap? Can traps be generated intentionally by a user program? If so, for what purpose?

An interrupt is an indicator used to notify the Operating system the occurrence of an event. Since, the modern OS are interrupt driven, the occurrence of an interrupt tells the OS the service/instruction that needs to be executed. Usually interrupts are caused when there is a change in the status or an event occurs in the hardware triggering the service of a routine called the Interrupt Service Routine, containing specialized set of instructions to service the hardware.

An interrupt is triggered by a hardware and a trap is caused by a software or an user level program. An interrupt is usually triggered between instructions while a trap occurs within instructions.

Yes, a trap can generated intentionally by a user program, if the user program wants a set of instruction to be carried out in the kernel mode. A trap is generated intentionally by the user program in the form of a system call. A system call is a method used by the user program to request the operating instruction to perform tasks reserved for the operating system on the user program's behalf.

1.27 Describe some of the challenges of designing operating systems for mobile devices compared with designing operating systems for traditional PCs.

Since, the mobile devices at targeted at convenience more than performance, the designers of operating systems face challenges due to the limiting size of the device.

The small size of the device in turns limits the memory between 1 MB and 3 GB(in the higher end devices) compared to its counter parts, traditional PCs, having several GBS. Hence, the operating systems and applications should manage memory every efficiently and frugally. The lack of usage of virtual memory technique in many devices restricts the developers to work within the confines of physical memory.

Another significant challenge faced by the developers is the speed of the processors and power management. In order to maintain luxury and convenience, the mobile devices are equipped with smaller battery, hence, limiting the power available , resulting in the use of low power processors. These low power processors run at a fraction of a speed compared to its counterparts. Hence, the operating system and applications should be designed not to tax the processor.

The lack of physical space limits the input and output methods. Hence, the developers need to condense the actual content into smaller displays and retrieve data from limited input options.

2.9 List five services provided by an operating system, and explain how each creates convenience for users. In which cases would it be impossible for user-level programs to provide these services? Explain your answer.

Some of the services provided by an operating system are as follows:

- *User Interface* - Most of the OS provides this service in one of the three forms and some provide all the three variations. Each of the variations help the user in their own way. For example, the GUI interface provides a more convenient and simple overview of the OS. It provides a smooth way of interacting with the Operating system. Whereas, the command line interface provides a more powerful interaction with the system calls, that is suitable for advanced users/ programmers.

- *Program Execution* - This service of the operating system is responsible for the loading of a program into the memory and its execution. It also takes care of the end of executions either normally or abnormally.
  This service would be impossible to be provided by a user program as this requires memory management. Memory management is a privileged operation controlled only by the OS.

- *I/O Operations* - This service enables I/O operations to/from a file/ I/O devices. For efficiency and protection, an user-level program is restricted from providing these services.

- *File System Manipulation* - This allows the programs to read, write, create and delete files and directories. This also takes care of the permission management to allow/deny access to files and directories.

- *Communications* - The OS also manages communication between two processes. This is usually implemented via shared memory or through message passing.

2.18 What are the two models of interprocess communication? What are the strengths and weaknesses of the two approaches?

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

Shared memory model

Shared memory model allows maximum speed and convenience of communication, as it can be performed at memory transfer speeds when it occurs inside a computer.

The processes are responsible for ensuring that they are not writing into the same memory location simultaneously. Hence, its main disadvantage lies in the areas of protection and synchronization between the processes.

Message passing Model

This model is useful for exchanging smaller amounts of data. It is much easier to implement compared to the Shared memory model.

This process has the overhead of opening a connection and closing a connection when the communication is over. This is possible only when the recipient name is known by the process.

2.21 What is the main advantage of the microkernel approach to system design? How do user programs and system services interact in a microkernel architecture? What are the disadvantages of using the microkernel approach?

A microkernel architecture organizes the operating systems by removing all the non-essential parts of the kernel and redistributing them as system and user-level programs. This results in a smaller kernel.

The main advantage of the microkernel approach is the ease of extending the operating system. All the new services can be implemented in the user-space and hence do not require modification of the kernel. Even a modification of the kernel would be minimal due to small size of the kernel. This results in an operating system that is easier to port from one hardware design to another. As most of the services are running in user-space, this results in enhanced security and reliability.

The user programs and system services interact via the communication facility provided by the micro kernel. The communication is provided by message passing in which messages are exchanged between the programs and micro kernel.

The main disadvantage of microkernel architecture is the decreased performance due to increased system function overhead.

2.24 Explain why Java programs running on Android systems do not use the standard Java API and virtual machine.

Instead of using the standard Java API, google has designed a separate Android API for Java development. Android also uses the specially designed Dalvik virtual machine instead of the standard virtual machine or JVM. In android, the java classes are first compiled to Java bytecode which is then translated into an executable file to run on the Dalvik virtual machine. This Dalvik virtual machine is specially designed for Android which comes optimized for mobile devices with limited memory and CPU processing capability. Hence, the Java programs running on Android don't use the standard Java API or the virtual machine.