**DAY 4 Assignment**

**Operating System Concepts**

An operating system (OS) is a collection of software that manages computer hardware resources and provides common services for computer programs. The operating system is a vital component of the system software in a computer system.An Operating System (OS) is an interface between a computer user and computer hardware. An operating system is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.

Important functions of an operating System.

* Memory Management
* Processor Management
* Device Management
* File Management
* Security
* Control over system performance
* Job accounting
* Error detecting aids
* Coordination between other software and users

**Process Management**

A process is an instance of a program, or part of a program, in memory. Processes are executed by the operating system or executive in order to perform complex tasks: such as play a movie or video, play a game, or even run the editor used to write this text in. In essence, one could say a process is a program – but a program could contain multiple processes. For example, a basic program to display a string might be built in its own program file. Loading the program might yield the operating system or executive to load other program files – dynamic loading of shared libraries containing executable code for the process to call and use. All of these program files are a part of the same process; its why a process can have an instance of multiple program files or even multiple instances.

Process management is the process by which operating systems manage processes, threads, enable processes to share information, protect process resources and allocate system resources to processes that request them in a safe manner. This can be a daunting task to the operating system developer and can be very complex in design. Lets take a closer look at each one of these.

Inter-Process Communication (IPC) is a technique employed by many operating systems to allow communication between processes. This is typically done by message passing: the process would request to send a message to another process to the operating system which would send and queue the message to the other process if it is able to. IPC can be implemented in a number of different ways, the most common being files, pipes, sockets, message passing, signals, semaphores, shared memory, and memory mapped files. Operating systems may implement any or all of these methods of IPC. IPC is heavily used in some hybrid and monolithic kernel designs but is arguably most prominent in micro kernel designs.

**Memory management**

Memory management is the process of controlling and coordinating computer memory, assigning portions called blocks to various running programs to optimize overall system performance. Memory management resides in hardware, in the OS (operating system), and in programs and applications.  
  
In hardware, memory management involves components that physically store data, such as RAM (random access memory) chips, memory caches, and flash-based SSDs (solid-state drives). In the OS, memory management involves the allocation (and constant reallocation) of specific memory blocks to individual programs as user demands change. At the application level, memory management ensures the availability of adequate memory for the objects and data structures of each running program at all times. Application memory management combines two related tasks, known as allocation and recycling.  
  
When the program requests a block of memory, a part of the memory manager called the allocator assigns that block to the program.  
When a program no longer needs the data in previously allocated memory blocks, those blocks become available for reassignment. This task can be done manually (by the programmer) or automatically (by the memory manager).

**File systems**

A file system can be thought of as an index or database containing the physical location of every piece of data on the hard drive or another storage device. The data is usually organized in folders called directories, which can contain other folders and files

A file system consists of two or three layers. Sometimes the layers are explicitly separated, and sometimes the functions are combined.  
  
The logical file system is responsible for interaction with the user application. It provides the application program interface (API) for file operations — OPEN, CLOSE, READ, etc., and passes the requested operation to the layer below it for processing. The logical file system "manage[s] open file table entries and per-process file descriptors.This layer provides "file access, directory operations, security and protection.  
  
The second optional layer is the virtual file system. This interface allows support for multiple concurrent instances of physical file systems, each of which is called a file system implementation.  
  
The third layer is the physical file system. This layer is concerned with the physical operation of the storage device (e.g. disk). It processes physical blocks being read or written. It handles buffering and memory management and is responsible for the physical placement of blocks in specific locations on the storage medium. The physical file system interacts with the device drivers or with the channel to drive the storage device.

Some examples of File Systems include FAT, NTFS, exFAT etc.