- **Artificial Intelligence Applications:** Artificial intelligence means a situation where a computer can think like a human. The applications of this concept are varied and include logical games like chess, bridge etc. Most of these applications use LISP and PROLOG languages, which are logic based.

Advantages of High-Level Languages:

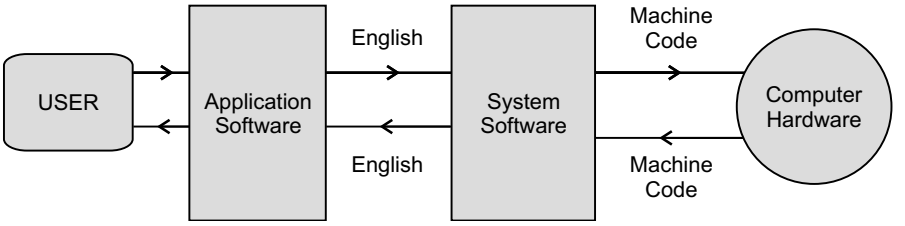
- easy to learn
- easy to understand
- easy to program
- easy to maintain
- easy to document
- easy to debug
- less time-consuming
- portable

Here, *debugging* means the removal of errors in the program, and portability means when a program is written in a high-level language, it can run on any computer irrespective of the hardware configuration.

2.6 Operating Systems

2.6.1 Definition

An *operating system* is system software. It is a set of programs that provides an interface between the user and the computer system (hardware). In other words, it coordinates the flow of information from the computer to the user and vice versa.



2.6.2 Functions of an Operating System

The operating system performs a number of services and/or the functions for the users of the computer. These operating system (OS) services or functions that might be performed are as follows:

- ***Memory Management*:** Here memory means random access memory (RAM). When the programs are loaded into the memory and are executed, the OS reads them from the hard disk and loads them into the memory (RAM). But before it loads, the OS checks whether the memory is available or not, and if available, it allocates it to the program. Once the program execution is over, the OS removes the program from the memory (RAM) and this freed memory can be used for another program.
- ***CPU Management*:** A microprocessor executes a number of processes at one time. Deciding which process is to be executed first, and which next, is a complex problem. This problem is effectively handled by the OS and is known as *job scheduling*. The OS helps the processor to schedule its activities so as to fulfil the requirements of various parts of the computer system. The processor gets the job from the memory, processes the job, and passes the result onto the predetermined place and makes itself ready for the next job.
- ***Disk Management*:** Data and program storage on the hard disk or any other storage device is a complex phenomenon. It is very difficult for a common user to understand this. Thus, the OS spares the user from understanding this process as it coordinates the storing and retrieval of files.
- ***Input/Output Management*:** Numbers of peripheral devices such as keyboard, mouse, printer, projectors, etc., are attached to a computer system. To make these devices work properly, the system requires supporting software called *device drivers* in the memory. The device drivers are also known as the *support utility programs* of the OS and are controlled by it. All these devices communicate with the computer for obtaining the job and getting the job done, which is again monitored by the OS.
- ***User Interface*:** The system, but not the user, understands machine language, which consists of 0s and 1s. The OS bridges the gap between the machine understandable language and human understandable language by providing a command interpreter.

- ***Communication*:** A computer system runs a number of processes at a given point of time and there exist situations where one process needs to exchange information with another process. The OS facilitates these communication processes within a system and among the systems. Communication is done either via shared memory or by the technique of *message passing* in which packets of information are moved between processes by the OS.
- ***Error Detection*:** While working on a system, it is quite possible that a user may commit a number of errors. The OS constantly keeps track of all possible errors. Errors may be typing a wrong file name, wrong syntax of a command, network failure, a printer jam, or running out of paper. These are errors that are frequently committed by a user. For each type of error, the OS takes an appropriate action by showing an appropriate message on the computer screen prompting the user to correct their action.
- ***Resource Allocation*:** The OS does the proper allocation of computer resources as there might be multiple users or multiple jobs running at the same time. Resources like processing time, main memory (RAM), file storage, devices, etc. must be allocated to all of the users equally, and this is handled by the OS.
- ***Accounting*:** The OS keeps full track of how much and what kind of computer resources are used by a user. This record keeping can be used for improving computing services.
- ***Protection*:** The OS provides a multilevel protection mechanism to users. This protection is required by users from other users or an outsider. The OS provides a mechanism by which a user can protect his data, and this is done by controlling the access to system resources. For this, the OS uses the authentication process in which each user has to authenticate himself to the system. This is done by means of a multilevel password protection system provided by an OS.

2.6.3 Types and Classifications of Operating Systems

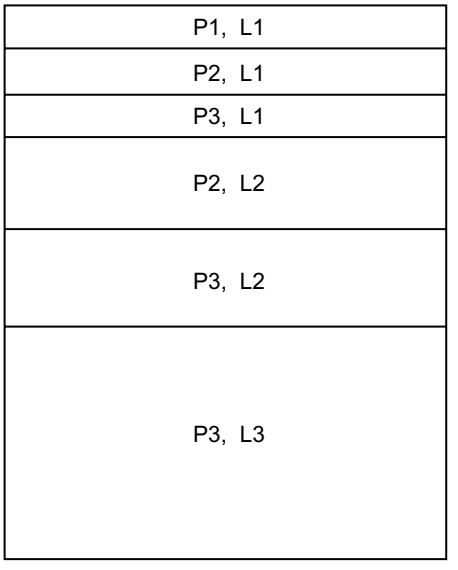
System software can be broadly classified into the following categories on the basis of their usage:

- **Batch Processing System Software:** A negligible interaction between the user and the program *batch processing system software*. In this type of system jobs are processed in the order in which they are entered, that is, on a "first in, first out" basis (FIFO). In a batch processing

system, memory is divided into two parts; one is permanently occupied by the software, whereas the other is used as per the need of the user. It simplifies the processing operations because the instructions are executed in batches, and thus saves the processor time.

- **Multi-User Operating System Software:** The *multi-user operating system* supports the multiple units of PCs called "terminals" that are attached to the main computer system as in mini and mainframe computer systems. It consists of only one central processing unit (a microprocessor) that performs all the operations. These systems are used when two or more users try to run programs at the same time. Examples of the multi-user operating system are UNIX, MSV, etc.
- **Multiprogramming or Multi-Tasking System Software:** This is the system software that is capable of running more than one program at the same time. *Multiprogramming* can be defined as a process of creating a situation in which more than one program may be held in the main memory at one time, thus making it possible to process several programs at a time. The main objective of developing this kind of system software is to minimize unused microprocessor time. A computer switches from one job to another at a rapid rate under the time-sharing mode. Different terminals are used to enter jobs into the computer. After processing the first user's job, it proceeds to the second and then to the third and so on for a short period of time called the "time slices," before returning to the first user's job from where it earlier started. This cycle continues indefinitely. When one program is finished the other program replaces it. UNIX, OS/2, and Windows are commonly used multiprogramming or *multitasking operating systems*. The processor is kept busy while channels and buffers are occupied with a job of bringing data and writing out information.

For example, let us assume that three users are working on a system simultaneously. In this concept the program of each user will be divided into a number of pages (layers) of equal size. During execution, the processor will divide its processing time equally among all of the users. It will first process the layer 1(L1) of the *program 1* (P1), then *layer 1* (L1) of the program 2 (P2), and finally the layer 1 (L1) of the program 3 (P3). The processor will give equal time to all the users, but it will appear to all of the users as if processor is giving its full time to them. When program 1 (P1) is finished, the processor will divide its time equally among the remaining programs (P2 and P3) and finally to the program 3 (L3).



- **Multiprocessing Operating System Software:** In a *multiprogramming operating system*, more than one program is processed by an operating system, whereas in a *multiprocessing operating system*, one program is processed by more than one processor. *A multiprocessing operating system* software uses multiple processors that share a common memory. Instructions from different and independent programs are processed at the same time by different processors. On the contrary, the processors may simultaneously execute different instructions from the same program. Examples of commonly used multiprocessing operating systems are OS/2, UNIX, MSV, etc. Multiprocessing systems can be classified as:
  - loosely coupled multiprocessing
  - functionally specialized processors
  - tightly coupled multiprocessing
  - parallel processing

Multiprocessor systems usually consist of two or more processors. Each processor has its own CU, ALU, etc. An interconnection mechanism allows each processor to access shared main memory and input/output devices. The processors not only communicate with each other through memory

but also are able to directly exchange signals. Memory is organized in such a manner such that it provides a multiple simultaneous access to a separate block of memory. The operating system controls this entire system and provides interaction between processors and their programs.

- **Real-Time System Software:** Real-time systems are the systems in which response time is critical. These are the systems that are involved with the immediate processing of data, machines, and records. These systems are designed to accept the data in real time, which means as soon as an activity occurs, the system processes the data immediately and generates the output in time to have an effect on the ongoing activity. Real-time systems are online systems with tighter constraints on response time. Examples of the real-time operating systems are C Executive, communications control program (CCP), CTOS, CTRON, FADOS, etc.

2.6.4 Components of an Operating System

An operating system consists of two primary components:

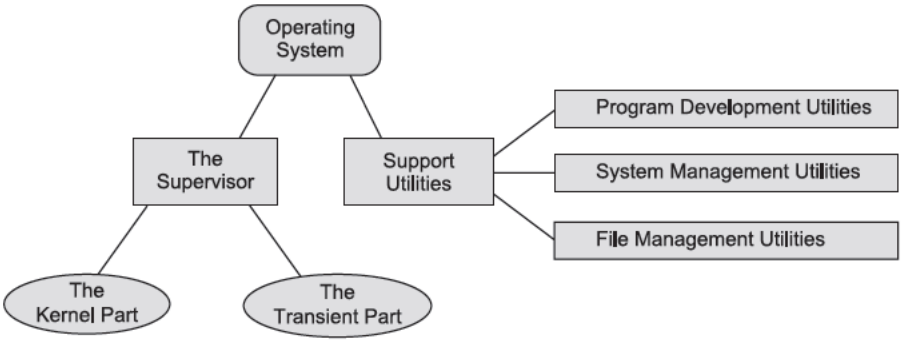
- a supervisor
- an integrated set of support utilities

1. **The Supervisor (or Control Program):** The *supervisor* is defined as the component of an operating system that takes care of the overall working of the computer system. It is a set of programs that are integrated to one another. It performs the following three basic functions:
   - It initializes the system at the time of start-up
   - It allows running of the application programs
   - It controls input and output devices attached to the system.

The supervisor also performs some additional functions such as keeping a track of computer time for different users, etc. The OS is generally found on a hard disk and sometimes in the form of a chip called *firmware*. The supervisor consists of two portions: a *kernel* and a *transient* portion. When the supervisor is loaded into the memory for the first time, both portions are loaded. The kernel part of supervisor always remains in the memory with an application program. This is to monitor the system operations. It's the part of an operating system that directly interacts with the hardware, and therefore it must be present in the memory as long as the computer is

being used. The transient portion need not always be present in the memory. When executed, a program may overwrite this part of the OS. Once the program execution is over, the transient portion is reloaded into the memory.

2. **The Support Utilities:** The *support utilities* are the system programs that perform useful functions. They are classified as follows:
   - ***Program Development Utilities*:** These include assemblers, editors, interpreters, compilers, and linkers.
   - ***System Management Utilities*:** These include the programs that keep track of more than one user, diagnostic routines, etc.
   - ***File Management Utilities*:** These include programs for copying files, erasing files, printing files, renaming files, etc.



2.7 What is Graphical User Interface (GUI)?

Microsoft Corporation brought a revolution in the area of system software development when they marketed software called Windows. Prior to the launch of Windows it was not possible for the user to work on a computer with the help of graphics. What the disk operating system (DOS) gave users was termed command line interface (CLI), where users were required to type the right command with the help of the keyboard on the command prompt (C:>_). Remembering the exact syntax and spelling of the command was a tedious task; however with Windows the concept of a *graphical user interface* (GUI) was introduced. With a GUI, the user did not get the