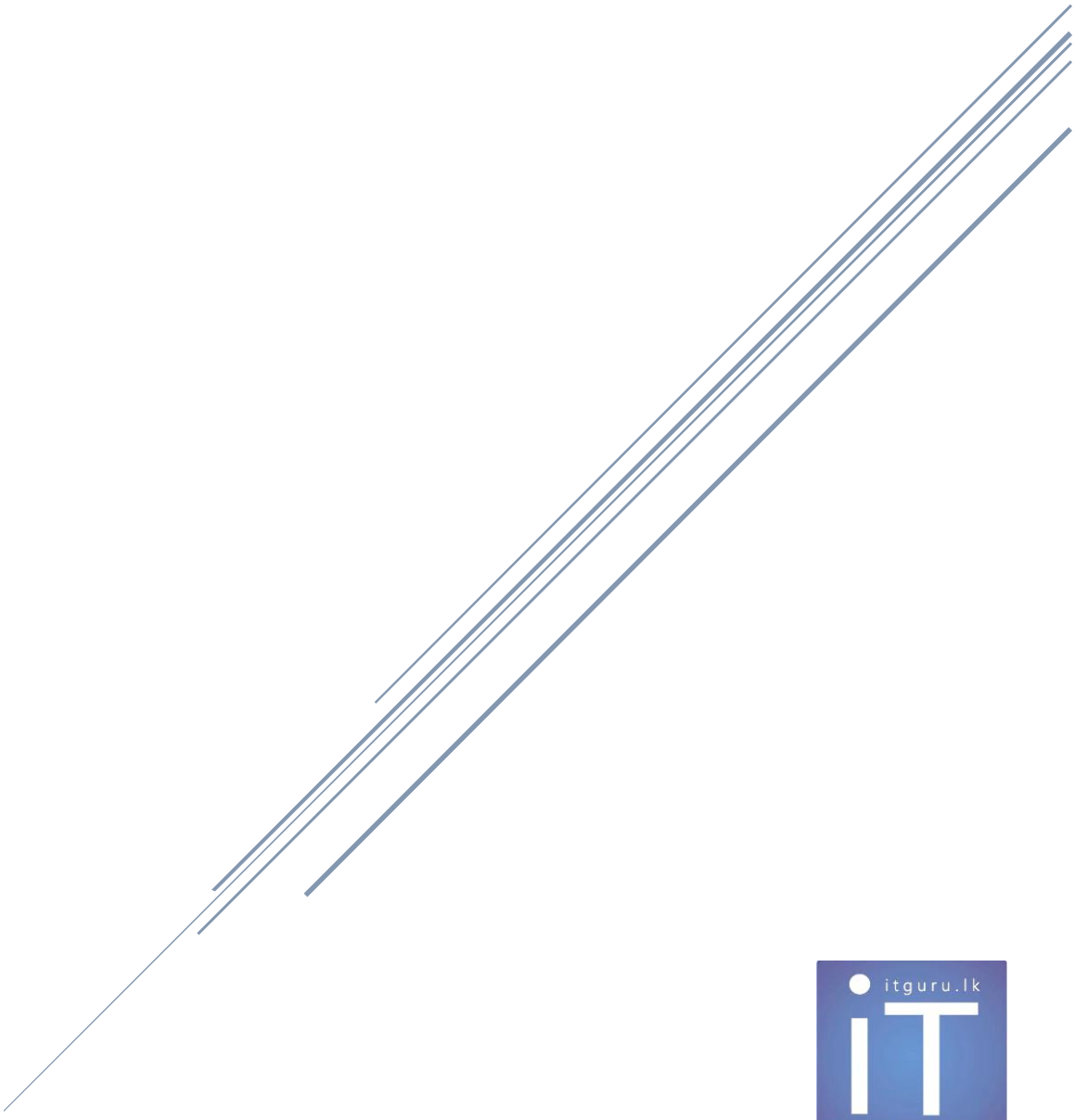
OPERATING SYSTEMS

G.C.E. Advanced Level ICT



itguru.lk
IT
Teran Subasinghe

Turning on your computer

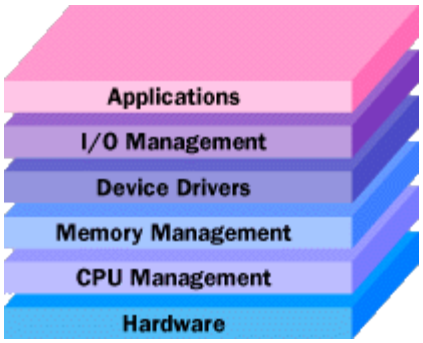
When you turn on the power to a computer, the first program that runs is usually a set of instructions kept in the computer's read-only memory (ROM). This code examines the system hardware to make sure everything is functioning properly. This power-on self-test (POST) checks the CPU, memory, and basic input-output systems (BIOS) for errors and stores the result in a special memory location. Once the POST has successfully completed, the software loaded in ROM (sometimes called the BIOS or **firmware**) will begin to activate the computer's disk drives. In most modern computers, when the computer activates the hard disk drive, it finds the first piece of the operating system: the **bootstrap loader**.

What is bootstrap loader

The bootstrap loader is a small program that has a single function: It loads the operating system into memory and allows it to begin operation. In the most basic form, the bootstrap loader sets up the small driver programs that interface with and control the various hardware subsystems of the computer. It sets up the divisions of memory that hold the operating system, user information and applications. It establishes the data structures that will hold the myriad signals, flags and semaphores that are used to communicate within and between the subsystems and applications of the computer. Then it turns control of the computer over to the operating system.

Tasks of an operating system

- Processor management
- Memory management
- Device management
- Storage management
- Application interface
- User interface



Processor Management

The heart of managing the processor comes down to two related issues:

- Ensuring that each process and application receives enough of the processor's time to function properly
- Using as many processor cycles as possible for real work

The basic unit of software that the operating system deals with in scheduling the work done by the processor is either a process or a thread, depending on the operating system.

What is a process?

It's tempting to think of a process as an application, but that gives an incomplete picture of how processes relate to the operating system and hardware. The application you see (word processor, spreadsheet or game) is, indeed, a process, but that application may cause several other processes to begin, for tasks like communications with other devices or other computers. There are also numerous processes that run without giving you direct evidence that they ever exist. For example, Windows XP and UNIX can have dozens of background processes running to handle the network, memory management, disk management, virus checks and so on.

> A process, then, is software that performs some action and can be controlled -- by a user, by other applications or by the operating system.

Attributes of a process

1. Process ID or Identification Number- Process id is given by the CPU when we Request for an Operation. Process id is also known as a unique identification Number which is Available when we request for a service. As we know there are many types of Operation those are performed on the Computer. So that for identification means which Process will be executed.

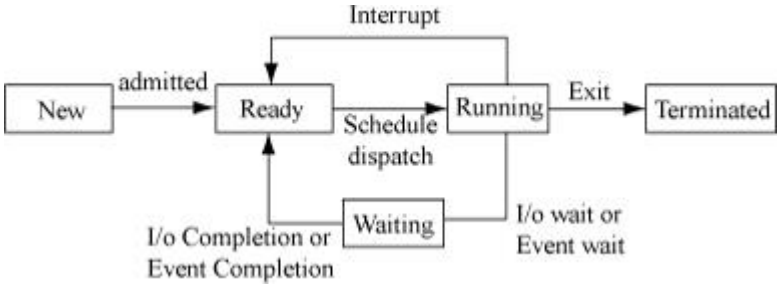
2. Process Name. : Name Specify the Description of the Process. Name of Operation which is performed by the Process. For Example Move the Mouse, Click on My Computer, Play the Song etc.

3. Process State (Ready, Active, Wait or Suspend). The Process has Some States, State Specify the Process State means whether a Process is running or not, Whether a Process wait for CPU etc. There are three Types of States
    i. Ready: Process wills Ready State, when we completed all the Input and Outputs. After giving the input, we wait for the Execution. After the Completion of user Interaction means after all the Inputs and Outputs.
    ii. Active: Active Means Process is running under the CPU.
    iii. Wait: When a Process is waiting for the Input and Outputs from the user then this is called as on Wait State.
4. Process Resources: For Running a Process, then there are many Resources used. For entering some data then we must use the Keyboard and for Sending data to the Computer CPU has used. So that For Performing any Operation, what types of Resources. All the Devices those are attached with the Computer are known as Resources of the System.

5. Scheduling Information. : Scheduling is used when there is Many Process those are running at a Time. Which Process will be executed by the CPU? So that we use the Scheduling, determines the Time of CPU, Means CPU Time Divided into the various Processes.

Process States

A process changes its state as it is executed. The various states that a process changes during execution are as follows:



- New—process is in a new state when it is created,
- Ready—process is in ready state when it is waiting for a processor,
- Running—process is in running state if processor is executing the process,
- Waiting—process is in waiting state when it waits for some event to happen (I/O etc), and
- Terminated—process that has finished execution is in terminated state.

It is processes, rather than applications, that the operating system controls and schedules for execution by the CPU. In a single-tasking system, the schedule is straightforward. The operating system allows the application to begin running, suspending the execution only long enough to deal with interrupts and user input.

What are interrupts?

Interrupts are special signals sent by hardware or software to the CPU. It's as if some part of the computer suddenly raised its hand to ask for the CPU's attention in a lively meeting. Sometimes the operating system will schedule the priority of processes so that interrupts are masked -- that is, the operating system will ignore the interrupts from some sources so that a particular job can be finished as quickly as possible. There are some interrupts (such as those from error conditions or problems with memory) that are so important that they can't be ignored. These non-makeable interrupts (NMIs) must be dealt with immediately, regardless of the other tasks at hand.

Single Tasking vs. Multi-Tasking

While interrupts add some complication to the execution of processes in a single-tasking system, the job of the operating system becomes much more complicated in a multi-tasking system. Now, the operating system must arrange the execution of applications so that you believe that there are several things happening at once. This is complicated because the CPU can only do one thing at

a time. Today's multi-core processors and multi-processor machines can handle more work, but each processor core is still capable of managing one task at a time.

In order to give the appearance of lots of things happening at the same time, the operating system has to switch between different processes thousands of times a second. Here's how it happens:

- A process occupies a certain amount of RAM. It also makes use of registers, stacks and queues within the CPU and operating-system memory space.
- When two processes are multi-tasking, the operating system allots a certain number of CPU execution cycles to one program.
- After that number of cycles, the operating system makes copies of all the registers, stacks and queues used by the processes, and notes the point at which the process paused in its execution.
- It then loads all the registers, stacks and queues used by the second process and allows it a certain number of CPU cycles.
- When those are complete, it makes copies of all the registers, stacks and queues used by the second program, and loads the first program.

Process Scheduling

The process scheduling is the activity of the process manager that handles the removal of the running process from the CPU and the selection of another process on the basis of a particular strategy.

Process scheduling is an essential part of a Multiprogramming operating system. Such operating systems allow more than one process to be loaded into the executable memory at a time and loaded process shares the CPU using time multiplexing.