ECE 405

MICROCOMPUTER & MICROPROCESSOR

WEEK THREE LECTURES

Introduction to Operating Systems

- In brief, an operating system is the set of programs that controls a computer. Some examples of operating systems are UNIX, Mach, MS-DOS, MS-Windows, Windows/NT, OS/2 and MacOS.
- A computer system can be divided roughly into four components: the *hardware, the operating system,* the *application programs and the users*

Introduction to Operating Systems

- An *operating system is a program that acts as an* intermediary between a user of a computer and the computer hardware.
- The purpose of an operating system is to provide an environment in which a user can execute programs.
- The primary goal of an operating system is thus to make the computer system *convenient to use.*
- A secondary goal is to use the computer hardware in an *efficient manner.*

Objectives of Operating Systems

- To hide details of hardware by creating abstraction.
- To allocate resources to processes (Manage
- resources).
- Provide a pleasant and effective user interface by Simplify the execution of user programs and make solving user problems easier.
- OS simplifies and manages the complexity of running application programs efficiently .

Objectives of Operating Systems

- Make application software portable and versatile.
- Provide isolation, security and protection among user programs.
- Improve overall system reliability (error confinement, fault tolerance, reconfiguration)

Operating Systems component

- **Process Management.**
- **Main-Memory Management.**
- **File Management.**
- **I/O System Management.**
- **Secondary-Storage Management.**
- **Networking,**
- **Protection System.**
- **Command Interpreter System**

Operating Systems component

- **Process Management**
- A *process is only ONE instant of a program in execution.*
- *The five major activities of an operating system in regard to process management are:*
- *Creation and deletion of user and system processes.*
- *Suspension and resumption of processes.*
- *A mechanism for process synchronization.*
- *A mechanism for process communication.*
- *A mechanism for deadlock handling*

Process Management Cont

- **Process, on the other hand, includes:**
- Current value of Program Counter (PC)
- Contents of the processors registers
- Value of the variables
- The processes stack (SP) which typically contains temporary data such as

  subroutine parameter, return address, and temporary variables.
- A data section that contains global variables.

Process Management Cont

- ***Process State***
- As a process executes, it changes state. The state of a process is defined in part by the current activity of that process. Each process may be in one of the following states:
- **New State: The process being created.**
- **• Running State: A process is said to be running if it has the CPU, that is,**
- process actually using the CPU at that particular instant.

Process Management Cont

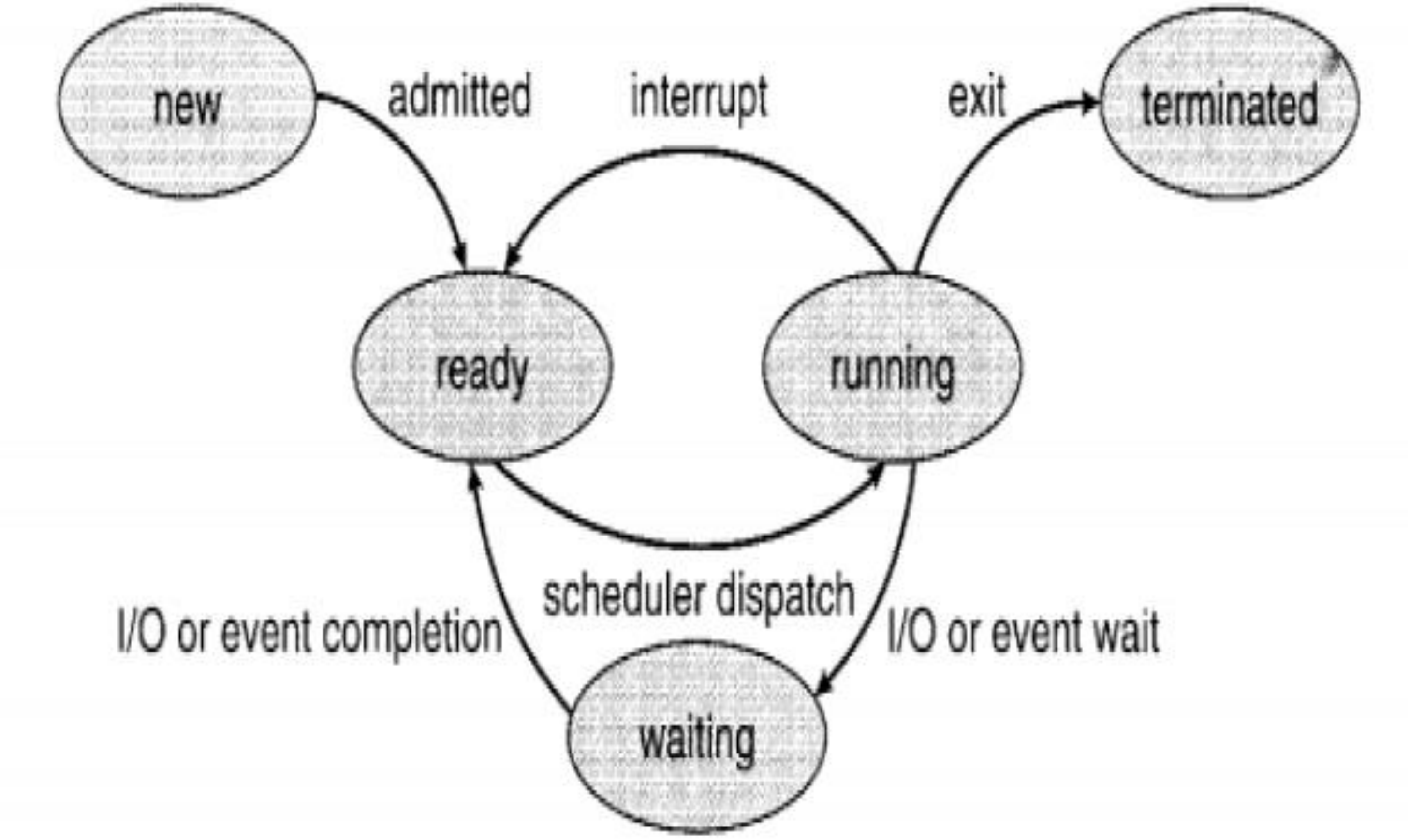

Figure : Diagram of process states.

Process Management Cont

- **Blocked (or waiting) State:** A process is said to be blocked if it is waiting for some event to happen such that as an I/O completion before it can proceed. Note that a process is unable to run until some external event happens.
- **Ready State:** A process is said to be ready if it is waiting to be assigned to a processor.
- **Terminated state:** The process has finished execution

Operating Systems component

- **Main-Memory Management**
- Main-Memory is a large array of words or bytes. Each word or byte has its own address. Main memory is a repository of quickly accessible data shared by the CPU and I/O devices.
- The major activities of an operating system in regard to memory-management are:
- *Keep track of which part of memory are currently being used and by whom.*
- *Decide which processes are loaded into memory when memory space becomes available.*
- *Allocate and deallocate memory space as needed*

Operating Systems component

- **File Management**
- A file is a collected of related information defined by its creator. Computer can store files on the disk (secondary storage), which provide long term storage.
- *The creation and deletion of files.*
- *The creation and deletion of directions.*
- *The support of primitives for manipulating files and directions.*
- *The mapping of files onto secondary storage.*
- *The backup of files on stable storage media.*

Operating Systems component

- **I/O System Management**
- One of the purposes of an operating system is to hide the peculiarities of specific hardware devices from the user.
- **Secondary-Storage Management**
- Generally speaking, systems have several levels of
- storage, including primary storage, secondary storage and cache storage. Instructions and data must be placed in primary storage or cache to be referenced by a running program

Operating Systems component

- **Networking**
- A distributed system is a collection of processors that do not share memory, peripheral devices, or a clock. The processors communicate with one another through communication lines called network.
- **Protection System**
- Protection refers to mechanism for controlling the access of programs, processes, or users to the resources defined by a computer system.
-

Operating Systems component

- **Command Interpreter System**

  A command interpreter is an interface of the operating system with
- the user. The user gives commands with are executed by operating system
- (usually by turning them into system calls).

Operating Systems Services

- **Program Execution**
- The system must be able to load a program into memory and to run it. The program must be able to end its execution, either normally or abnormally (indicating error).
- **I/O Operations**
- A running program may require I/O. This I/O may involve a file or an I/O device.

Operating Systems Services

- **File System Manipulation**

The output of a program may need to be written into new files or input taken from some files. The operating system provides this service.

- **Error Detection**

An error is one part of the system may cause malfunctioning of the complete system. To avoid such a situation the operating system constantly

monitors the system for detecting the errors

System Calls and System Programs

- System calls provide the interface between a process and the operating system. These calls are generally available as assembly-language instructions, and are usually listed in the

  manuals used by assembly-language programmers.

  It involves Passing parameters between a running program and OS via registers, memory tables or stack.