CSC 224 Principles of OS

2. I/O Management and

OS Processes

I/O System Management

- The module that keeps track of the status of devices is called the I/O traffic controller.
- Each I/O device has a device handler that resides in a separate process associated with that device.
- I/O is the communication between an information processing system and the outside world

- The I/O subsystem consists of

  1. A memory management component that includes buffering, caching and spooling.
- 2. A general device driver interface.

Assembler

- Input to an assembler is an assembly language program.
- Output is an object program plus information that enables the loader to prepare the object program for execution.

Compiler

- A compilers is a program that accepts a source program in a ―high-level language and produces a corresponding object program.

Loader

- A loader is a routine that loads an object program and prepares it for execution.

OS Processes

- A process is a program in execution.
- A process is also defined as an entity which represents the basic unit of work to be implemented in the system.
- The execution of a process must progress in a sequential fashion.

Components of a Process

- 1. **Object Program**
- Code to be executed.

- **2. Data**
- Data to be used for executing the program

- 3. **Resources**
- While executing the program, it may require some resources

- **4. Status**
- Verifies the status of the process execution. A process can run to completion only when all requested resources have been allocated to the process.
- Two or more processes could be executing the same program, each using their own data and resources.

Process Vs Program

- A program by itself is not a process. It is a static entity made up of program statement while process is a dynamic entity. Program contains the instructions to be executed by processor.
- A program takes a space at single place in main memory and continues to stay there. A program does not perform any action by itself.

- A process is a sequence of information executions. Process exists in a limited span of time.
- Two or more processes could be executing the same program, each using their own data and resources

Process states

- As a process executes, it changes state.
- The state of a process is defined as the current activity of the process.
- Process can have one of the following five states at a time:
  - New
  - Ready
  - Running
  - Waiting
  - Terminated

Process state

1. New

- The process is being created.

2. Ready

- The process is waiting to be assigned to a processor. Ready processes are waiting to have the processor allocated to them by the operating system so that they can run.

3. Running

- Process instructions are being executed (i.e. The process that is currently being executed).

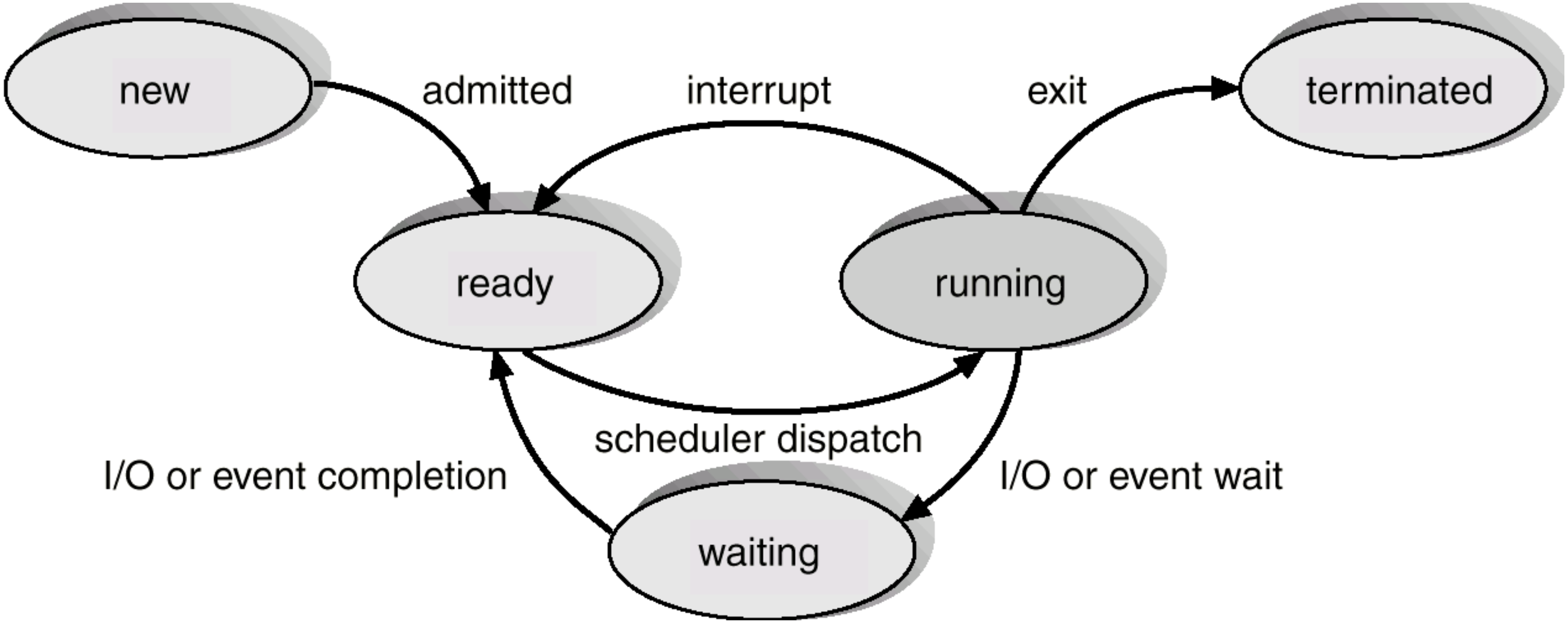
4. Waiting

- The process is waiting for some event to occur (such as the completion of an I/O operation).

5. Terminated

- The process has finished execution.

Diagram of a Process state



PCB

- Whenever processes changes state, the operating system reacts by placing the process PCB in the list that corresponds to its new state.
- Only one process can be running on any processor at any instant and many processes may be ready and waiting state.

Process Control Block

- Each process is represented in the operating system by a process control block (PCB) also called a task control block.
- PCB is the data structure used by the operating system.
- Operating system groups all information that needs about particular process.
- PCB contains many pieces of information associated with a specific process e.g.

1. Pointer

- Pointer points to another process control block. Pointer is used for maintaining the scheduling list.

2. Process State

- Process state may be new, ready, running, waiting and so on.

3. Program Counter

- Program Counter indicates the address of the next instruction to be executed for this process.

4. CPU registers

- CPU registers include general purpose register, stack pointers, index registers and accumulators etc. number of register and type of register totally depends upon the computer architecture.

5. Memory management information

- This information may include the value of base and limit registers, the page tables, or the segment tables depending on the memory system used by the operating system.
- This information is useful for deallocating the memory when the process terminates.

6. Accounting information

- This information includes the amount of CPU and real time used, time limits, job or process numbers, account numbers etc.

- PCB also includes the information about CPU scheduling, I/O resource management, file management information, priority and so on.

- Draw a diagram for process state
- Draw a PCB block

Suspended processes

- The following are characteristics of a suspended process:
- 1. It is not immediately available for execution.
- 2. It may or may not be waiting on an event.
- 3. For preventing the execution, process is suspend by OS, parent process, process itself and an agent.
- 4. Process may not be removed from the suspended state until the agent orders the removal.

Reasons for process suspension

1. **Swapping** : OS needs to release required main memory to bring in a process that is ready to execute.

2. **Timing** : Process may be suspended while waiting for the next time interval.

Reasons for process suspension

3. **Interactive user request :** Process may be suspended for debugging purpose by user.

4. **Parent process request :** To modify the suspended process or to coordinate the activity of various descendants.

Revision questions

- Differentiate between a Compiler and an Interpreter
- What is the distinction between buffering, caching and spooling?

Process Management/Scheduling