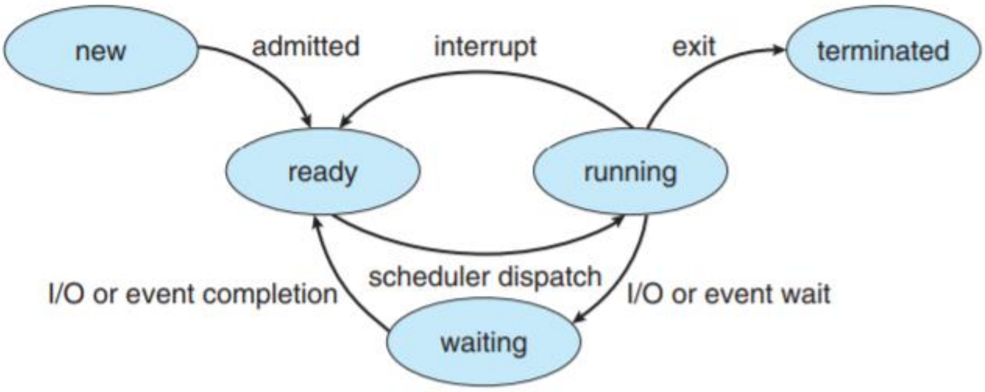
UNIT -2 OS

1. **Define process.**

- A process is a program under execution.
- In program code (text section) includes the value of the program counter and the contents of the processor's registers, process stack and a data section containing global variables.

2. **What are different states in which and when a process can exist?**

   New: A new process is being created
   Running: given Instructions are being executed
   Waiting: waiting for some event to occur (I/O completion or reception of a signal)
   Ready: process is waiting to be assigned to a processor
   Terminated: process has finished execution.

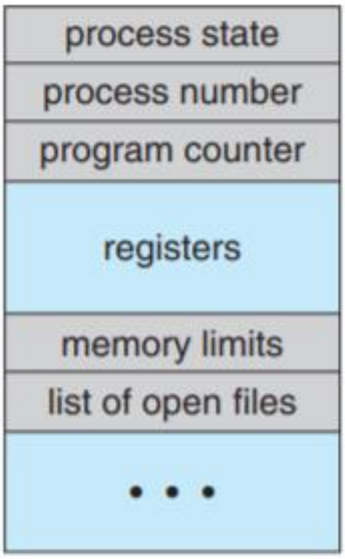


3. **What is PCB? Mention its contents.**

   PCB stands for Process Control Block.

   It including process state(run,wait..etc) program counter(loaction of instruction to excute), CPU registers(content of process), CPU scheduling information(priority,queue pointer), memory management information(memory of process), accounting information (cpu,clock used)and I/O status information(I/o device,open files)

   Process Control Block Diagram



What are the different types of schedulers? Define them.

Long term or job scheduler: Selects processes from the secondary storage and loads them in to main memory. This controls the degree of multiprogramming.

Short term scheduler: Selects processes from ready queue to execute, and allocates the CPU to one of them.runs very frequently

Medium term scheduler: Removes some process from main memory and swaps out to secondary storage. This is needed to improve process mix or when memory is overcommitted.and improve degree of multi programming,manage swap in/out memory

Differentiate between long-term and short-term schedulers.

- ***Long term scheduler*** selects processes from the secondary storage and loads in to the main memory and controls degree of multi programming. It is invoked less frequently i.e the period between two invokes is long and hence called long term scheduler.
- ***Short term scheduler*** selects a process from the ready queue based on a scheduling algorithm and assigns it to a processor. To keep the processor always busy this scheduler is invoked frequently. i.e the period between two invokes is short and hence called short term scheduler.

What is meant by context switch?

- When a process moves to waiting state, its status is saved and another ready process is loaded into the CPU.
- This operation is called **context switch**, and the time it takes is known as **switching overhead**.

What is cooperating process?

- A system dependent process which can affect or be affected by the other processes executing in the system are called cooperating processes.
- Simiple,any process that shares data with other processes is a cooperating process.

What is the need for process cooperation?

The major needs for process cooperation are Information sharing, Computation speedup, Modularity and Convenience.

Define thread. How is it related to a process?

- A thread is a basic unit of CPU utilization
- It consists of a program counter, a register set and a stack space. It shares with peer threads its code section, data section and OS resources collectively known as a task.
- The thread is called a Light weight process.
- A traditional or heavyweight process is equal to a task with single thread.

What is IPC?

- IPC stands for Inter Process Communication.
- It provided by OS for cooperating processes to communicate with each other.
- 2 types -shared memory and message passing.

What is meant by CPU bound and I/O bound programs?

- A program with more computation statements and less I/O statements is called CPU bound.
- A program with more I/O operations and less computations is called I/O bound.

Differentiate between preemptive and non preemptive scheduling.

non preemptive scheduling -

- process voluntarily releases the processor only when it switches from running state to waiting state and when it terminates.
- Control- cooperative
- Perfomance-stable and response- predictable

preemptive scheduling-

- ✧ In addition to above circumstances the processor can be forcibly removed from the process when a higher priority process is ready or when an interrupt occurs.
- ✧ Control- interruptable
- ✧ Perfomance-overheaded and response- responsive.

Why strict non preemptive scheduling is unlikely to be used in a computer?

If strict non preemptive scheduling is used then, urgent higher priority processes may have to weight and this may lead to some serious errors.

What is a dispatcher?

Dispatcher module gives control of the CPU to the process selected by the short-term scheduler;

this involves:

switching context

switching to user mode

jumping to the proper location in the user program to restart that program

Dispatch latency – time it takes for the dispatcher to stop one process and start another running

Cpu scheduler

CPU scheduling decisions may take place when a process:

1. Switches from running to waiting state
2. Switches from running to ready state
3. Switches from waiting to ready

Terminates

Scheduling under 1 and 4 is nonpreemptive

All other scheduling is preemptive

Define synchronization and asynchronization?

Message passing may block or non block

block sender-sender is blocked until the message is received

Block receiver-the receiver is blocked until a message is available

Non-blocking send -- the sender sends the message and continue

Non-blocking receive -- the receiver receives:A valid message, Null message

Define buffering

Buffering is the process of storing data in a temporary memory area (buffer) while transferring between two devices or between a device and an application.

Types - **single buffer**- one buffer is used

Double buffer - two buffer is used

Circular buffer - multiple buffer arranged in circular queue

Implemented in three ways

Zero capacity – no messages on link.

Sender must wait for receiver (rendezvous)

Bounded capacity – finite length of n messages

Sender must wait if link full

Unbounded capacity – infinite length

Sender never waits

List the objectives of scheduling.

The major objectives of CPU or process scheduling are to maximize the CPU utilization, to increase the throughput, to minimize the turnaround time, waiting time and response time.

Define various scheduling criteria used to compare the scheduling algorithms.

CPU utilization refers how much busy the CPU is.

Throughput is the number of processes that are completed per unit time

Turnaround time is the time interval between submission of a process and its completion. It is calculated from its arrival time till completion time including the waiting time.

Waiting time is the total amount of time a process waits in the ready queue from its arrival.

Response time is the time from the submission of a request until the first response is produced. This is used for interactive systems.

How do SRTN scheduling differs from SJF scheduling?

SJF is Shortest Job First scheduling and it is a non preemptive scheduling.

SRTN is Shortest Remaining Time Next scheduling and it is the preemptive version of SJF scheduling.

What is time slice or time quantum?

In Round Robin (RR) scheduling each process in the ready queue is allowed to use the processor in circular manner for a certain time period in the order in which they arrived. This time period is called time slice or time quantum.

What is critical section or critical region? Mention the requirements for a solution to the critical section problem?

Critical section or critical region is a segment of code in each cooperating process, in which the process may be changing common variables, updating a table, writing a file and so on.

Hence one process - executing critical section,

cooperating process - not allowed to execute in its critical section.

Requirment- mutal exclusion and bound waiting

What is race condition?

Race condition is a condition in which the two or more processes depends on the relative rate at which each process executes. A race condition can cause a pair of processes to violate a critical section or lead to deadlock.

What is meant by starvation?

Starvation is a phenomenon in many resource allocation strategies in which some set of processes are perpetually ignored because their priority is not as high that of other processes. Starvation can occur in CPU scheduling, dish scheduling or any other kind of resource allocation strategy.

Define semaphore.

A Semaphore S is an integer variable, which apart from initialization can be accessed only through standard atomic operations Wait and Signal.

```
Wait(S): while S <= 0 do no-op;
             S = S - 1;
Signal(S): S = S + 1;
```

Semaphore is a synchronization mechanism used in modern OS.

Differentiate between binary and counting semaphores.

Binary semaphore -only take 0 and 1 as values.

The counting semaphore - take any integer value.

Define deadlock. Specify the necessary conditions for deadlock to occur.

It can arise when two or more processes hold resources and request for other resources.

Some process holds a resource that another wants while requesting a second resource, and the other process holds the second resource while requesting the first.

Hence neither process can progress.

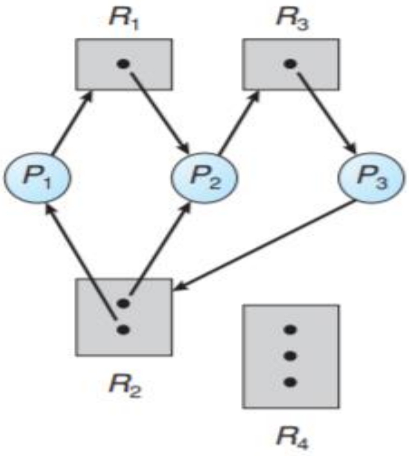
A system is said to be in a deadlock state when two or more processes are deadlocked.

The four necessary simultaneous conditions for deadlock to occur are Mutual exclusion, Hold and wait, No preemption and Circular wait.

Define resource allocation graph.

Resource allocation graph consists of a set of vertices V and a set of edges E. The set V has two types set of processes P and set of resources R. The set E has two types request edges (directed edges from a process to a resource) and assignment edges (directed edges from a resource to process).

Resource allocation graph example



Differentiate between deadlock prevention and deadlock avoidance techniques.

Feature	Deadlock Avoidance	Deadlock Prevention
Approach	Dynamically checks system state to avoid deadlock	Prevents deadlock by negating one of the four necessary conditions
Resource Declaration	Requires processes to declare maximum resource needs in advance	No need for maximum resource declaration
Resource Allocation	Allocates resources only if the system remains in a **safe state**	Restricts conditions like hold-and-wait or enforces ordering
System State Tracking	Continuously monitors resource allocation state	Does not require constant state monitoring
Mutual Exclusion	Assumed as necessary for non-sharable resources	Not required for sharable resources
Hold and Wait	Allowed but controlled via state checking	Not allowed; processes must request all resources at once or none

Feature	Deadlock Avoidance	Deadlock Prevention
No Preemption	Not emphasized	Enforced; preempts resources if request cannot be immediately fulfilled
Circular Wait	Avoided via dynamic safe-state checking	Prevented by defining a total ordering of resources
Resource Utilization	Typically higher (flexible allocation)	Lower (due to strict conditions)
Starvation Possibility	Low (decisions based on state)	High (due to forced waiting or repeated preemption)

What is meant by safe state?

A system is said to be in safe state, if the system can allocate resources to each process (up to its maximum) in some order and still avoid a deadlock.

Mention the data structures used in Banker's algorithm.

Let n be the number of processes in the system and m be the number of resource types. Then the data structures used in the banker's algorithms are,

Available: A vector of length m indicates the number of currently available resources under each type.

Max: An n x m matrix defines the maximum demand of each process under each resource type.

Allocation: An n x m matrix defining the number of resources under each type currently allocated to each process.

Need: A n x m matrix defining the remaining resource need of each process. Need matrix is obtained by subtracting Allocation matrix from Max matrix.

Is it possible to have a deadlock involving only one single process? Justify.

No. Since there is no other process holding or competing for resources in the system.

Define producer and consumer

Producer

- Generates data and places it into a shared buffer.
- Works concurrently with the consumer process
- Must check for buffer overflow before producing.
- Signals the consumer when new data is available.
- Can be blocked if the buffer is full.
- Typically used in data generation or input tasks

Consumer

- Retrieves data from the shared buffer for processing
- Waits for the producer to generate data.
- Must check for buffer underflow before consuming.
- Signals the producer when space is available
- Can be blocked if the buffer is empty.
- Common in applications like printers or data processing