Semester-3

Created by Sudhanshu Sharma

Operating System

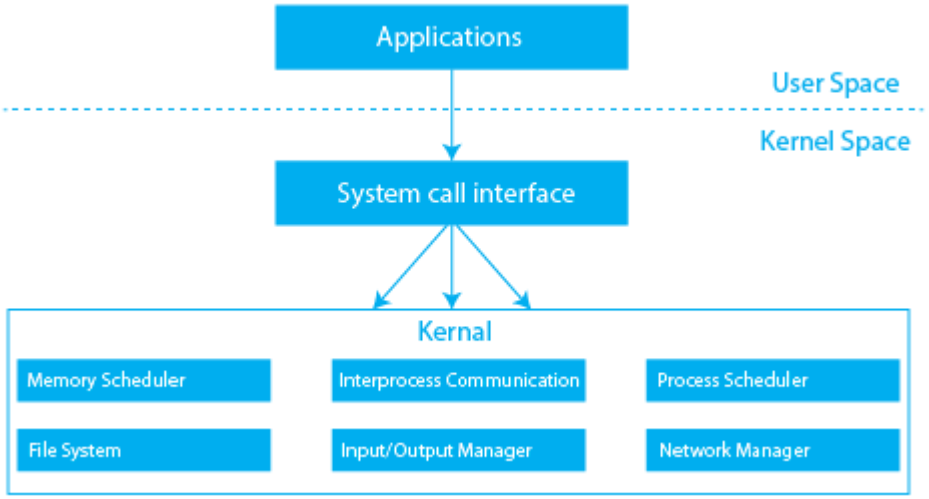
(According to Purvanchal University Syllabus)

Unit – 1

Introduction

- An Operating System is the Set of instructions which is provide the conventional way between the Hardware and Software.

Architecture of Operating System –



Functions Of Operating System –

- There are Various Type of Functions of Operating System 1. Memory Management
  2. Processor Management
  3. Device Management
  4. File Management
  5. Security
  6. Control over system performance
  7. Job accounting
  8. Error detecting aids
  9. Coordination between other software and users

Memory Management–

Memory management refers to management of Primary Memory or Main Memory. Main memory is a large array of words or bytes where each word or byte has its own address.

Main memory provides a fast storage that can be accessed directly by the CPU. For a program to be executed, it must in the main memory. An Operating System does the following activities for memory management.

- Keeps tracks of primary memory, i.e., what part of it are in use by whom, what part are not in use.
- In multiprogramming, the OS decides which process will get memory when and how much.
- Allocates the memory when a process requests it to do so. De-allocates the memory when a process no longer needs it or has been terminated.

Processor Management -

In multiprogramming environment, the OS decides which process gets the processor when and for how much time. This function is called process scheduling. An Operating System does the following activities for processor management –

- Keeps tracks of processor and status of process. The program responsible for this task is known as traffic controller.
- Allocates the processor (CPU) to a process.
- De-allocates processor when a process is no longer required.

Device Management -

An Operating System manages device communication via their respective drivers. It does the following activities for device management –

- Keeps tracks of all devices. Program responsible for this task is known as the I/O controller.
- Decides which process gets the device when and for how much time.

- Allocates the device in the efficient way.
- De-allocates devices.

File Management -

A file system is normally organized into directories for easy navigation and usage. These directories may contain files and other directions.

An Operating System does the following activities for file management –

- Keeps track of information, location, uses, status etc. The collective facilities are often known as file system.
- Decides who gets the resources.
- Allocates the resources.
- De-allocates the resources.

Security Management –

By means of password and similar other techniques, it prevents unauthorized access to programs and data.

Control over system performance −

Recording delays between request for a service and response from the system.

Job accounting −

Keeping track of time and resources used by various jobs and users.

Error detecting aids –

Production of dumps, traces, error messages, and other debugging and error detecting aids.

Coordination between other software and users

–Coordination and assignment of compilers, interpreters, assemblers and other software to the various users of the computer systems.

Evaluation Of Operating System –

- The evolution of operating systems is directly dependent on the development of computer systems and how users use them.

Here is a quick tour of computing systems through the past fifty years in the timeline.

Early Evolution

- 1945: ENIAC, Moore School of Engineering, University of Pennsylvania.
- 1949: EDSAC and EDVAC
- 1949: BINAC - a successor to the ENIAC
- 1951: UNIVAC by Remington
- 1952: IBM 701
- 1956: The interrupt
- 1954-1957: FORTRAN was developed

Operating Systems - Late 1950s

- By the late 1950s Operating systems were well improved and started supporting following usages:
- It was able to perform Single stream batch processing.
- It could use Common, standardized, input/output routines for device access.
- Program transition capabilities to reduce the overhead of starting a new job were added.
- Error recovery to clean up after a job terminated abnormally was added.
- Job control languages that allowed users to specify the job definition and resource requirements were made possible.

Operating Systems - In 1960s

- 1961: The dawn of minicomputers
- 1962: Compatible Time-Sharing System (CTSS) from MIT
- 1963: Burroughs Master Control Program (MCP) for the B5000 system
- 1964: IBM System/360
- 1960s: Disks became mainstream
- 1966: Minicomputers got cheaper, more powerful, and really useful.
- 1967-1968: Mouse was invented.
- 1964 and onward: Multics
    - 1969: The UNIX Time-Sharing System from Bell Telephone Laboratories.

Supported OS Features by 1970s

- Multi User and Multi tasking was introduced.
- Dynamic address translation hardware and Virtual machines came into picture.
- Modular architectures came into existence.
- Personal, interactive systems came into existence.

Accomplishments after 1970

- 1971: Intel announces the microprocessor
- 1972: IBM comes out with VM: the Virtual Machine Operating System
- 1973: UNIX 4th Edition is published
- 1973: Ethernet
- 1974 The Personal Computer Age begins
- 1974: Gates and Allen wrote BASIC for the Altair
- 1976: Apple II
- August 12, 1981: IBM introduces the IBM PC
- 1983 Microsoft begins work on MS-Windows
- 1984 Apple Macintosh comes out
- 1990 Microsoft Windows 3.0 comes out
- 1991 GNU/Linux
- 1992 The first Windows virus comes out
- 1993 Windows NT
- 2007: iOS
- 2008: Android OS

Batch–

- The users of a batch operating system do not interact with the computer directly. Each user prepares his job on an off-line device like punch cards and submits it to the computer operator. To speed up processing, jobs with similar needs are batched together and run as a group. The programmers leave their programs with the operator and the operator then sorts the programs with similar requirements into batches.

The problems with Batch Systems are as follows −

- Lack of interaction between the user and the job.
- CPU is often idle, because the speed of the mechanical I/O devices is slower than the CPU.

- Difficult to provide the desired priority.

Interactive–

- An interactive operating system is one that allows the user to directly interact with the operating system whilst one or more programs are running.

Time Sharing & Real Time Systems–

`Time Sharing –

- Time-sharing is a technique which enables many people, located at various terminals, to use a particular computer system at the same time. Time-sharing or multitasking is a logical extension of multiprogramming. Processor's time which is shared among multiple users simultaneously is termed as time-sharing.

Advantages of Timesharing operating systems are as follows −

- Provides the advantage of quick response.
- Avoids duplication of software.
- Reduces CPU idle time.

Disadvantages of Time-sharing operating systems are as follows −

- Problem of reliability.
- Question of security and integrity of user programs and data.
- Problem of data communication.

Real Time –

- A real-time system is defined as a data processing system in which the time interval required to process and respond to inputs is so small that it controls the environment. The time taken by the system to respond to an input and display of required updated information is termed as the response time. So in this method, the response time is very less as compared to online processing.

System Protection–

- The processes in an operating system must beprotected from one another's activities. To provide such protection, we can use various mechanisms to ensure that only processes that have gained proper authorization from the operating system

can operate on the files, memory segments, CPU, and other resources of a system.

System Components–

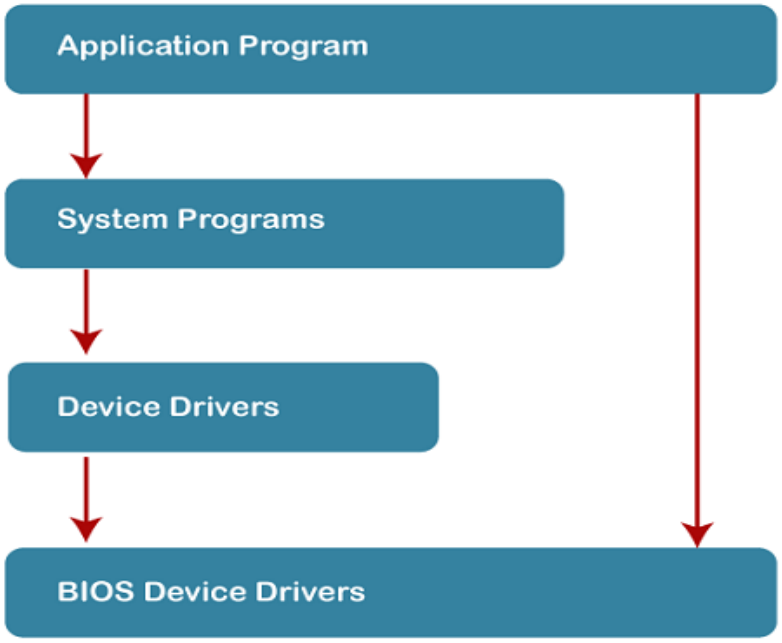
- Process management
- I/O management
- Main Memory management
- File & Storage Management
- Protection
- Networking
- Protection
- Command Interpreter

System Structure–

Simple Structure

- There are many operating systems that have a rather simple structure. These started as small systems and rapidly expanded much further than their scope. A common example of this is MS-DOS. It was designed simply for a niche amount for people. There was no indication that it would become so popular.

An image to illustrate the st
An image to illustrate the structure of MS-DOS is as follo

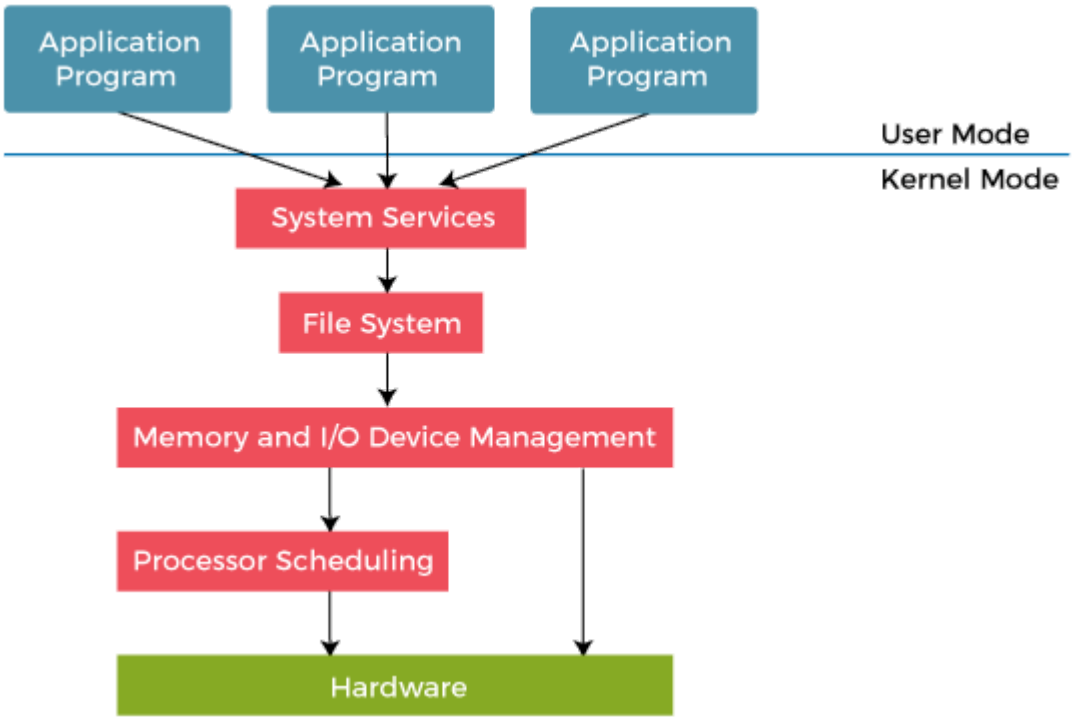


Layered Structure

- One way to achieve modularity in the operating system is the layered One way to achieve modularity in the operating system is the layered

  One way to achieve modularity in the operating system is the layered approach. In this, the bottom layer is the hardware and the topmost approach. In this, the bottom layer is the hardware and the topmost approach. In this, the bottom layer is the hardware and the topmost layer is the user interface. layer is the user interface.

An image demonstrating the layered approach is as follows:



Operating System Services–

There are various type of services such as-

- Process Management
- Memory Management
- Security Management
- I/O Management
- Error Detection

- Communication

I/O -

Operating System Check all I/O Devices Proper functioning

or not

Error Detection -

If Process have Occur any error than O.S. Detect These error and short out them.

Communication -

The OS stands the communication b/w Two Computer.

Unit – 2

Concurrent Process

- When a multiuser access in a single machine at a time is known as Concurrent process.

In Other words

- Concurrent processing is a computing model in which multiple processors execute instructions simultaneously for better performance. Concurrent means something that happens at the same time as something else.

Process–

- The Flow of Execution is known as process.
- A process is basically a program in execution. The execution of a process must progress in a sequential fashion.
- When a program is loaded into the memory and it becomes a process, it can be divided into four sections ─ stack, heap, text and data. The following image shows a simplified layout of a process inside main memory −

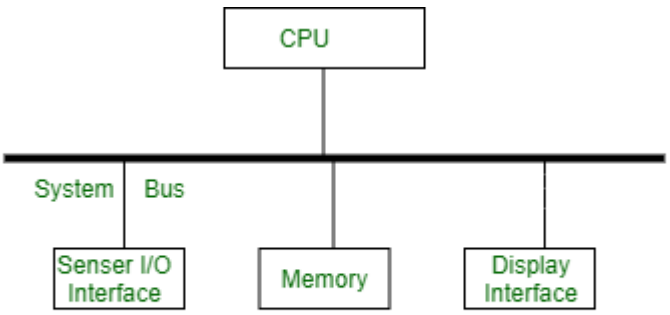
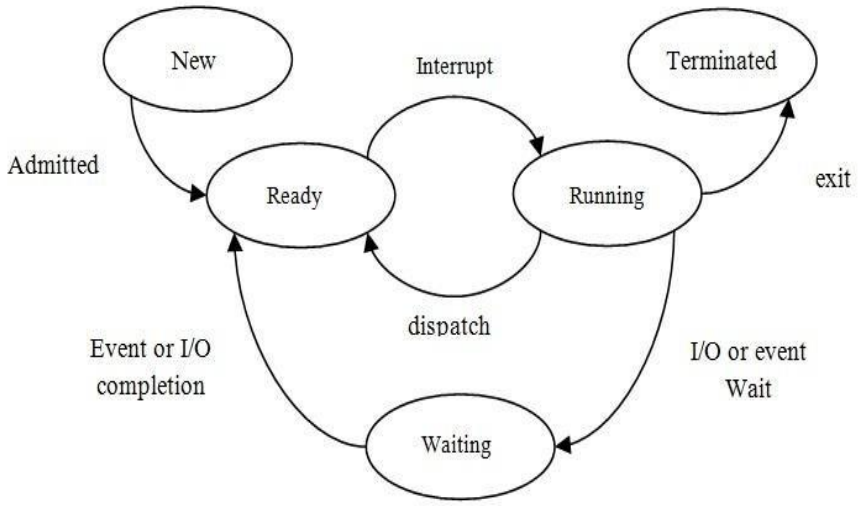


Figure - Multiprogramming (Single CPU) Environment

State Transition–



Interrupts–

- An interrupt is a function of an operating system that provides multi process multi-tasking. The interrupt is a signal that prompts the operating system to stop work on one process and start work on another.

Process Control Block–

- PCB is a data structure of O.S which is store the all Information about your process.Such as –

1. Process state
2. Process ID
3. Program Counter
4. CPU Register
5. Memory Management
6. I/O Status

S.N. Information & Description

1	Process State The current state of the process i.e., whether it is ready, running, waiting, or whatever.

2 Process privileges-

This is required to allow/disallow access to system resources.

3 Process ID-

Unique identification for each of the process in the operating system.

4 Pointer-

A pointer to parent process.

5 Program Counter-

Program Counter is a pointer to the address of the next instruction to be executed for this process.

6 CPU registers-

Various CPU registers where process need to be stored for execution for running state.

7 CPU Scheduling Information-

Process priority and other scheduling information which is required to schedule the process.

8 Memory management information-

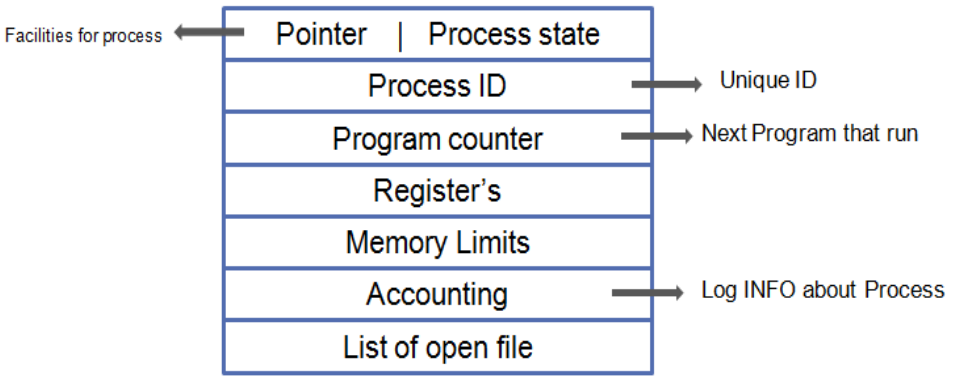
This includes the information of page table, memory limits, Segment table depending on memory used by the operating system.

9 Accounting information-

This includes the amount of CPU used for process execution, time limits, execution ID etc.

10	IO status information This includes a list of I/O devices allocated to the process.

PROCESS CONTROL BLOCK (PCB)



PCB Diagram

Principle of Concurrency–

- When a multiuser access in a single machine at a time is known as Concurrent process.

There are 2 examples for concurrent processing:

- In a single processor multi programming system, processes are interleaved in time to yield the appearance of simultaneous execution.
- In a multiprocessor system, it is possible not only to interleaved the execution of multiple processes but also to overlap them.

Producer-Consumer Problem –

- The Producer Consumer problem also known as bounded buffer problem is classic example of multi-process synchronization problem. 🎬 The problem describes two processes the producer and the consumer who share a common fix size buffer used as a cube.
- The producer job is generate data put it into a buffer and the same time the consumer is consuming the data.

Critical Section –

- In a critical section multiple process work in a single data is known as critical section. So it's change the state of the data.

Unit – 3

CPU Scheduling

Scheduling Concept–

- CPU Scheduling is the process which allow one process to use the CPU while the execution of other process on the hold.
- There are various type of scheduling
  - o Long Term Scheduler
  - o Short term
  - o Medium term

Long Term Scheduler – It is also called a job Scheduler. A long-term scheduler determines which programs are admitted to the system for processing.

It selects Processes from the queue and loads them into memory for Execution. Process loads into the memory for CPU Scheduling.

Short Term Scheduling –

- Short term scheduler also known as disnature makes the decision of which process to execute next over the RAM.

Medium Term Scheduler –

- MTS is the part of Swapping. It removes the process from the memory.

The MTS is in charge and handling the swap out process.

Types Of CPU Scheduling

Thereare 2 types of CPU Scheduling -

1. Non Primitive
2. Primitive

Non-Primitive –

Under non primitive scheduling one the CPU has been allotted to the process the process keeps the CPU until it Release or Terminate.

Primitive -

In this types of scheduling the task are usually assign with priority.

Performance Critaria–

1. CPU Utilization –
   To make out the best use of CPU and not to waste the CPU Cycle. 2. Turn Around –
   It is the amount the time taken to execute a particular process. 3. Throw Put –
   The total number of process completed per unit time.
4. Waiting Time –
   The sum of period spends waiting in the ready Queue.
5. Response Time –
   Amount of time is takes from when a request was

Scheduling Algorithms–

- There are various types of Algorithm such as –
  1. FCFS
  2. SJF
  3. PriorityScheduling
  4. Round Robin

FCFS–

- Jobs are executed on first come first serve.
- It is a non-Primitive scheduling Algorithm.

Q.

Process Arrival Time Execution Service Time P0 0 5 0 P1 1

3 5 P2 2 8 8 P3 3 6 16 Giant Chart –

P0	P1	P2	P3

0 5 8 16 22

Waiting Time – P0 = 0-0 =0

P1 = 5-1 =4

P2 = 8-2 =6

P3 = 16-3 =13

Average Waiting Time = 0+4+6+13

4

= 5.75 Ans…

SJF –

- Stands for shortest job First. It is a non-primitive algorithm.
- It is a best approach to minimise waiting time.

Q.

Process Arrival Time Execution Service Time P0 0 5 0 P1 1

3 5 P2 2 8 8

P3 3 6 16

P1	P0	P3	P2

0 3 8 14 22 Waiting Time – P0 = 3-0 =3

P1 = 0 = 0

P2 = 14-2 =12

P3 = 8-3 = 5

Average waiting time – 3+0+12+5

5

= 5

PriorityScheduling –

- Priority scheduling Non-Primitive Algorithm each process is assign a priority.
- Processes with the same priority are executed on

  FCFS. Q.

Process Arrival Time ExecutionPriorityService Time P0 0 5

1 0 P1 1 3 2 5 P2 2 8 1 8 P3 3 6 3 16

P3	P1	P0	P3

0 6 9 14 22 Waiting time = P0 = 9-0 = 9

P1= 6-1 = 5

P2 = 14-2 =12

P3 = 0

Average waiting time = 9+5+12+0

4

= 6.5 Ans…

Round Robin –

- Round robin is the pre-empting process scheduling.
- Each process is provide a fix time to execute it is called

  quantum.

  Q.

Process Arrival Time Execution Service Time P0 0 5 9 P1 1

3 6 P2 2 8 14 P3 3 6 0 Quantum time = 3

2 0 5 3 0 2 0 0

P0	P1	P2	P3	P0	P2	P3	P2

0 2 6 9 12 14 17 20 22 P0 = (0-0) + (12-3) = 9

P1 = 3-1 = 2

P2 = (9-3) + (14-9) =11

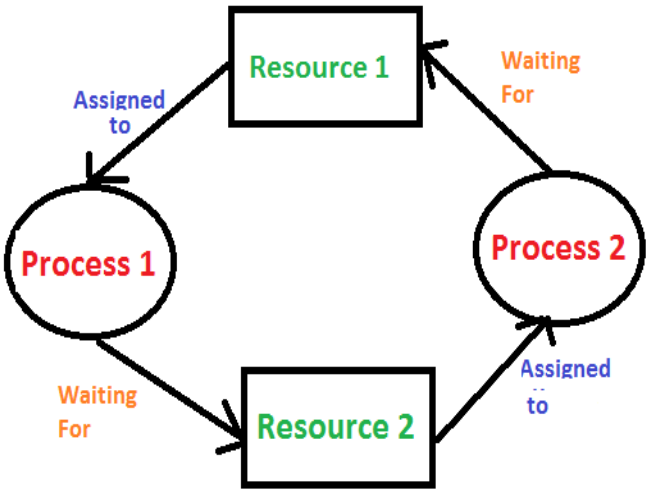
Average time = 9+2+12+11

4

= 8.5 Ans…

Deadlock

- Deadlock is a situation where a set of process are block because each process is holding a resource and waiting for another acquiring by some other process.



Method for deadlock handling

There are 3 way for handling deadlock

1. **Deadlock Prevention or avoidance:** The idea is to not let the system into deadlock state.
2. **Deadlock detection and recovery:** Let deadlock occur then do pre-emption to handle it once occurred.
3. **Ignore the problem all together:** If dead is very rare then let it happen and reboot the system. This is the approach that both windows and UNIX take.

System Model–

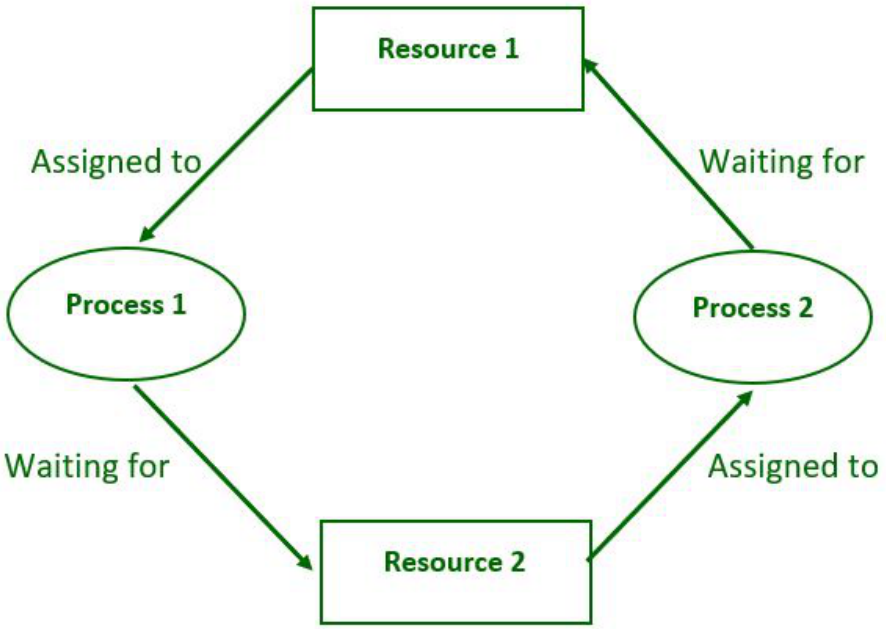


Figure: Deadlock in Operating system

DeadlockCharacterization–

- Deadlock can arises in 4 conditions hold simultaneously

1. Mutual exclusion

- One or more than one resource non-sharable.

2. Hold and wait –

- A process is holding at least one resource and waiting for another resource.

3. No pre-emption –

- A resource cannot be taken from a process unless the process releases the resource.

4. Circular wait –

- A set of are process waiting for each other.

Prevention–

- Deadlock prevention algorithms ensure that at least one of the necessary conditions (Mutual exclusion, hold and wait, no pre-emption and circular wait) does not hold true. However most prevention algorithms have poor resource utilization, and hence result in reduced throughputs.

Unit – 4

Memory Management

Real Storage–

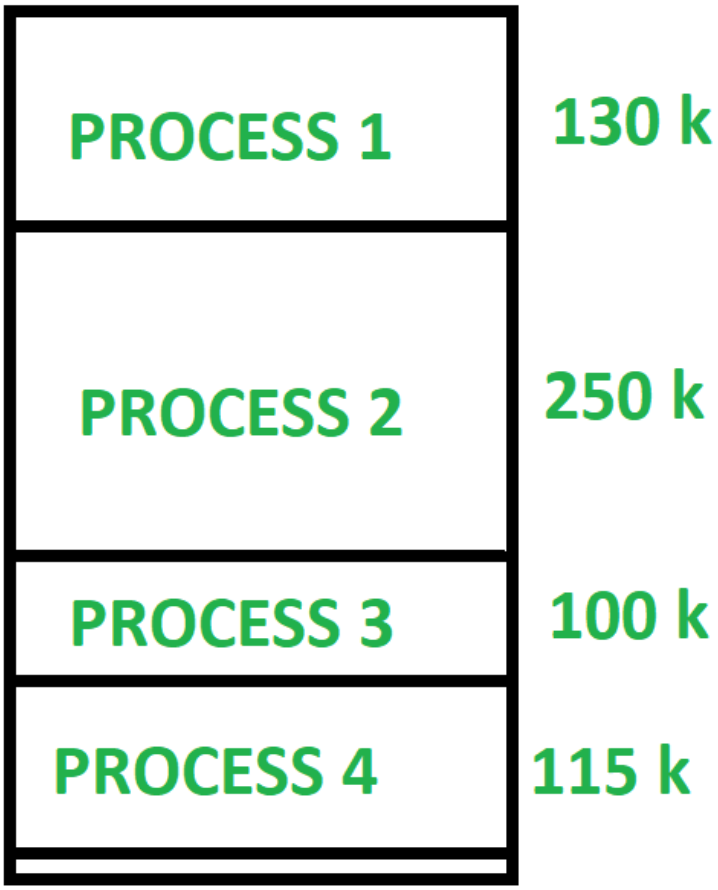
- The real memory is a obsilary memory in the main memory. Like Hard Disk, RAM Etc…

Resident Monitor–

- In computing, a resident monitor is a type of system software program that was used in many early computers from the 1950s to 1970s.
- It can be considered a precursor to the Operating system. 🎬 The name is derived from the program which is always present in the computer's memory thus being "resident".

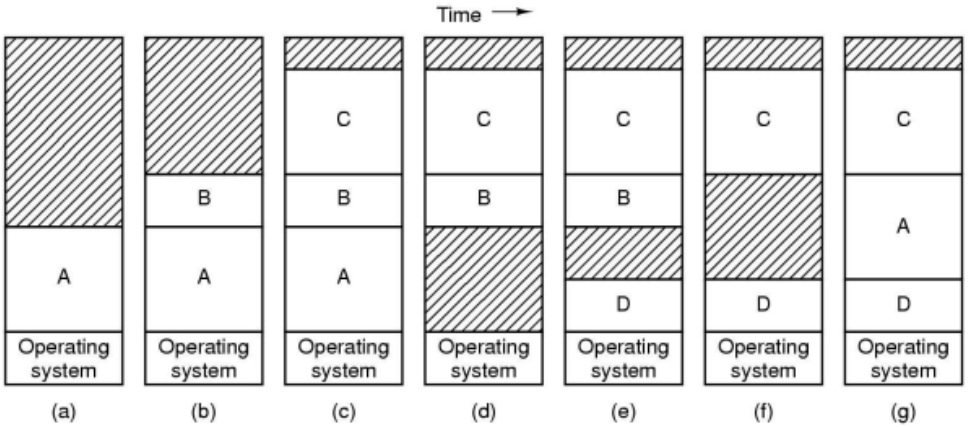
Multiprogramming with fixed Partition–

- In a multiprogramming environment several programs reside in primary memory at a time and the CPU pauses its control rapidly between these programs.
- On way to support multiprogramming is to divide the main memory into several partitions each of which is allocated to a single process. 🎬 Depending upon how and when partitions are created, there may be two types of memory partitioning: (1) Static and (2) Dynamic.
- Static partitioning implies that the division of memory into number of partitions and its size is made in the beginning (during the system generation process) and remain fixed thereafter. In dynamic partitioning, the size and the number of partitions are decided during the run time by the operating system.



Multiprogramming with variable Partition–

- Both the number and size of the partitions change with time.
- IBM OS/MVT (Multiprogramming with a varying number of tasks). Must perform an additional on every memory reference (i.e. on every address translation) to add the start address of the partition.
- Called a DAT (dynamic address translation) box by IBM.

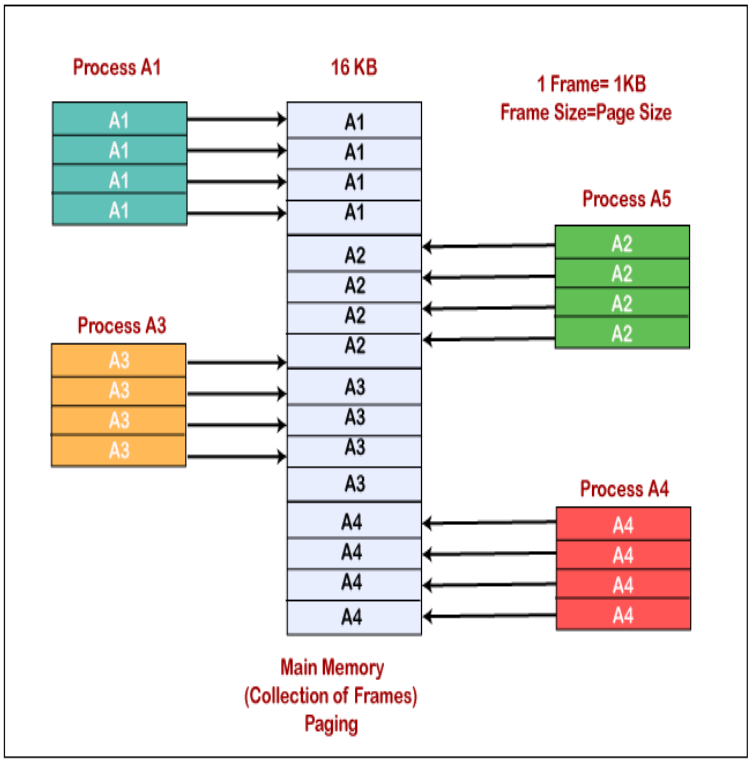


Multiple Base Register–

- The base registers are those that hold the address of the variable.
- There are various type of base register such as –
  - o Base Register
  - o Pointer Register
  - o Stack Register

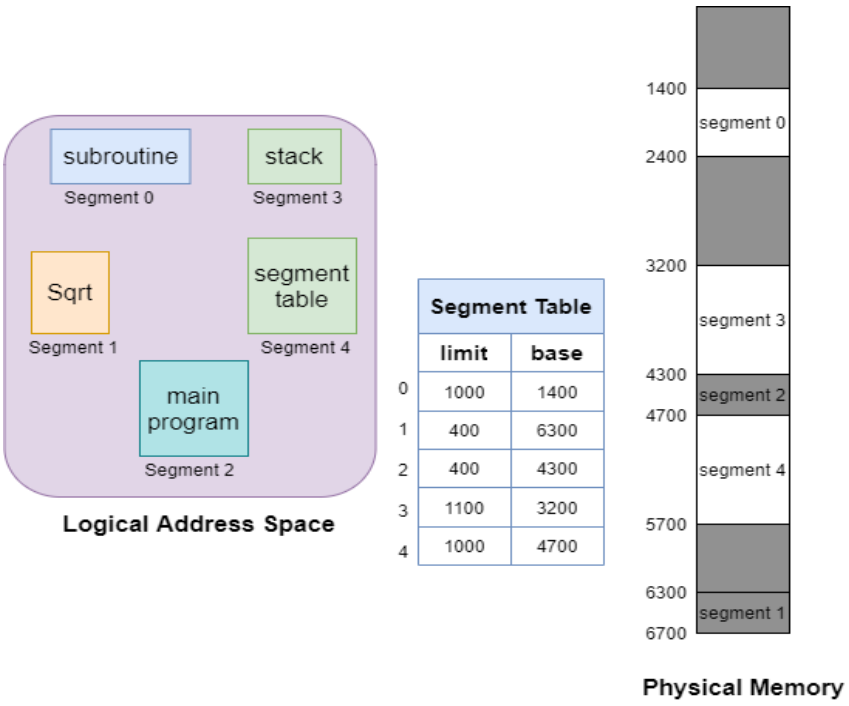
Paging–

- A computer can address more memory than the amount physically installed on the system.
- This extra memory is actually called virtual memory and it is a section of a hard that's set up to emulate the computer RAM.
- Paging technique plays an important role in implementing virtual memory.
- Paging is a memory management technique in which process address space is broken into blocks of the same size called pages.



Segmentation–

- Segmentation is a memory management technique in which each job is divided in to several segments of different sizes.
- Each segment is actually a different logical address space of the program.
- When a process is to be executed, its corresponding segmentation are loaded into non-contiguous memory though every segment is loaded into a contiguous block of available memory.
- Segmentation memory management works very similar to paging but here segments are of variable-length whereas in paging pages are of fixed size.



Paged Segmentation–

- The division can be done by the paging of size.

Virtual Memory Concept–

- Virtual memory is a memory management capability of an OS that uses hardware and software to allow a computer to compensate for physical memory shortages by temporarily transferring data from random access memory (RAM) to disk storage.
- Virtual address space is increased using active memory in RAM and inactive memory in hard disk drives (HDDs) to form contiguous addresses that hold both the application and its data.
- When the memory is small and data is big then we can use virtual memory concept.

Demand Paging–

- A demand system paging is quite similar to a paging system with swapping where processes reside in secondary memory and pages are loaded only on demand not in advance.
- When a contest switch occur, the OS does not copy any of the old program's pages out to the disk or any of the new program's pages into the main memory instead, it just begin executing the new program after loading the first page and fetches that program's

pages as they are referenced.

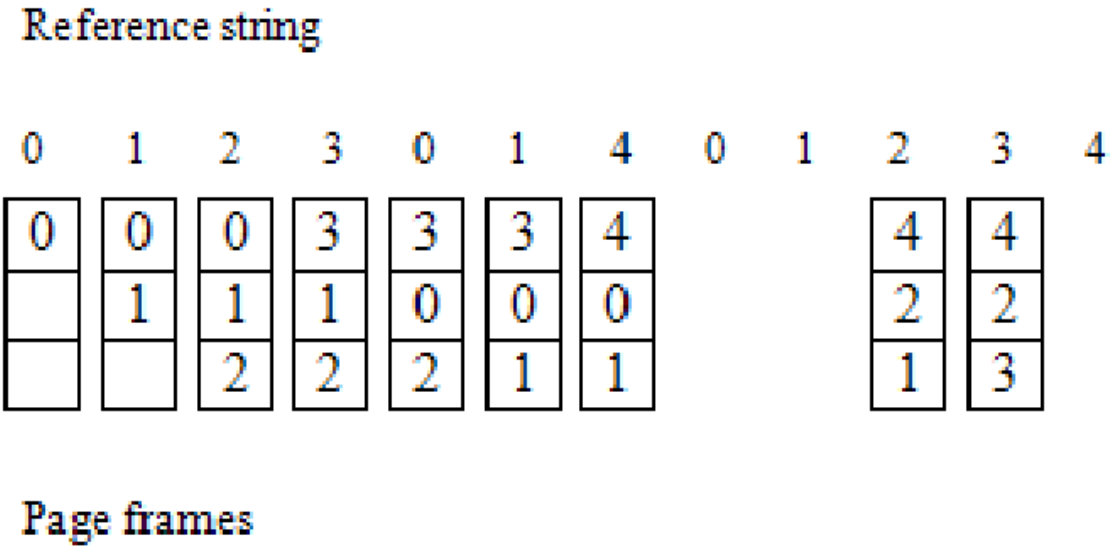
Page Replacement Algorithm–

- Page replacement algorithm are the techniques using which an Operating system decides Which memory pages to swap out, write to disk when a page of memory needs to be allocated.
- Paging happens whenever a page fault occurs and a free page cannot be used for allocation purpose accounting to region that pages are not available or the number of free pages is lower than required pages.

First in First Out (FIFO) Algorithm –

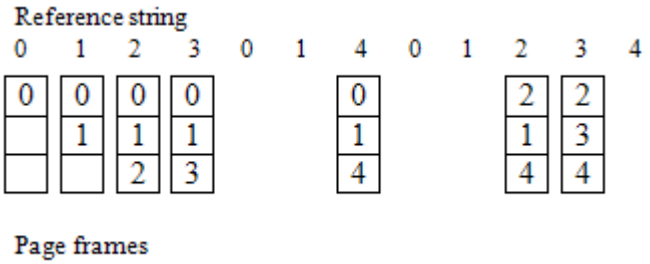
- Oldest page in main memory is the one which will be selected for replacement.

- Easy to implement, keep a list, replace pages from the tail and new pages at the head.



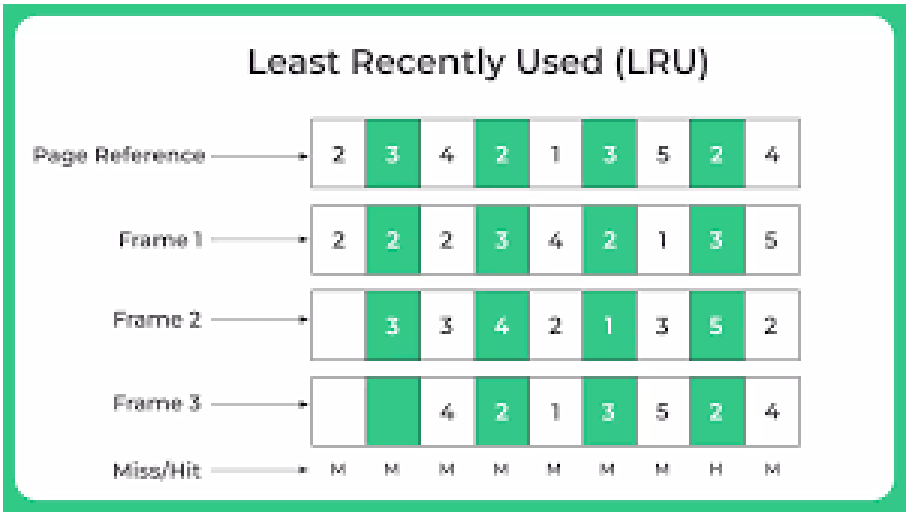
Optimal Page algorithm-

- An optimal page-replacement algorithm has the lowest page-fault rate of all algorithms. An optimal page-replacement algorithm exists, and has been called OPT or MIN.
- Replace the page that will not be used for the longest period of time. Use the time when a page is to be used.



Least Recently Used (LRU) algorithm-

- Page which has not been used for the longest time in main memory is the one which will be selected for replacement.
- Easy to implement, keep a list, replace pages by looking back into time.



Page buffering algorithm-

- To get a process start quickly, keep a pool of free frames.
- On page fault, select a page to be replaced.
- Write the new page in the frame of free pool, mark the page table and restart the process.
- Now write the dirty page out of disk and place the frame holding replaced page in free pool.

Least frequently Used (LFU) algorithm-

- The page with the smallest count is the one which will be selected for replacement.
- This algorithm suffers from the situation in which a page is used heavily during the initial phase of a process, but then is never used again.

Most frequently Used (MFU) algorithm-

- This algorithm is based on the argument that the page with the smallest count was probably just brought in and has yet

Allocation of Frames–

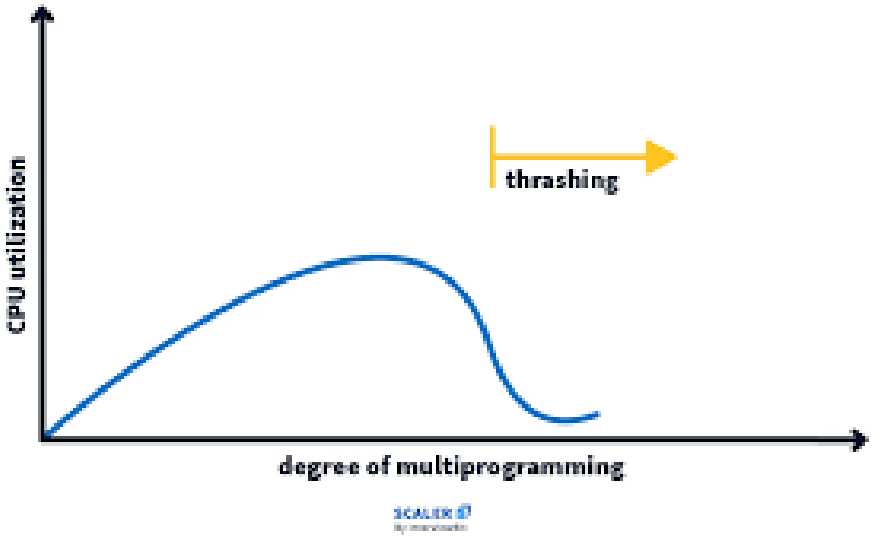
- Frames are the space available on the RAM which is allocated by the CPU.
    - Frame allocation algorithms are used if you have multiple processes; it helps decide how many frames to allocate to each process.

There are various constraints to the strategies for the allocation of frames:

- You cannot allocate more than the total number of available frames.
- At least a minimum number of frames should be allocated to each process. This constraint is supported by two reasons. The first reason is, as less number of frames are allocated, there is an increase in the page fault ratio, decreasing the performance of the execution of the process. Secondly, there should be enough frames to hold all the different pages that any single instruction can reference.

Thrashing–

- Thrashing is computer activity that makes little or no progress, usually because memory or other resources have become exhausted or too limited to perform needed operations. When this happens, a pattern typically develops in which a request is made of the operating system by a process or program, the operating system tries to find resources by taking them from some other process, which in turn makes new requests that can't be satisfied
- When the number of process too high and CPU utilization is too low.



Cache Memory Organization–

- A cache memory is a fast random access memory where the computer hardware stores copies of information currently used by programs (data and instructions), loaded from the main memory. The cache has a significantly shorter access time than the main memory due to the applied faster but more expensive implementation technology. The cache has a limited volume that also results from the properties of the applied technology. If information fetched to the cache memory is used again, the access time to it will be much shorter than in the case if this information were stored in the main memory and the program will execute faster.
  - Cache memory organization is the hierarchal model of the memory.

Impact on Performance–

- The CPU cache is the peace of hardware which is used to access time to data in the memory by keeping some part of frequently used data of the main memory.

Unit – 5

Unix/Linux

Unix System Kernel & Utilities–

- Kernel is the core component of the Operating system interacts directly with the hardware. Theshell takes command from the user and executes kernel function.
- Utility program that provide the user most of the functionality of an Operating system.

File & Directories–

- Files are those which are storing the content and the directory are the place where we store our files or subdirectory.

Single & Compound Statement–

- Compound statement for a block allows a sequence of statement to treat as a single statement.
- It is use {forcreating compound statement.
- Single Line are those which is write the statement in single line.

Basic Commands–

1. **PWD –** When you first open the terminal you are on the home directory of your user. To know which directory you are in you can use the PWD command.
2. **LS –** Use the LS command to known what files in the directory you are in.
3. **CD –** Use the CD Command to go to a directory.
4. **MKDIR –**Use the MKDIR Command when you want to create a folder or directory.
5. **RM –** The use of RM command to remove the directory into Linux
6. **Touch –** The touch is used to create a file.

7. **Help –** To know more about a command and how to use it.
8. **CP –** Use the CP Command to copy files through the command line.
9. **MV –** Use the MV Command to move files through the command line.
10. **Locate –** The locate command is use to locate a file. 11. **Echo –** The echo command helps us move some data, usually text into a file.
12. **Cat –** Use the cat command to display the content of a file. 13. **Sudo –**A widely used command in the Linux command line, sudo stands for “Super user do”, if you want to any command to be done with administrative or root privileges, you can use the sudo command.

Bourn Shell–

- The Bourne shell (sh) is a shell, or command-line interpreter, for computer operating systems.
- The Bourne shell was the default shell for Version 7 Unix. Most Unix like systems continue to have /bin/sh—which will be the Bourne shell, or a symbolic link or hard link to a compatible shell—even when other shells are used by most users.
- Developed by Stephen Bourne at Bell Labs, it was a replacement for the Thompson shell, whose executable file had the same name—sh. It was released in 1979 in the Version 7 Unix release distributed to colleges and universities. Although it is used as an interactive command interpreter, it was also intended as a scripting
  language and contains most of the features that are commonly considered to produce structured programs.

Korn Shell–

- Korn Shell scripting language as well as a user level login shell.
- Scripting command tend to be more readable then low level tool.
- It is a power tool

C Shell–

- A shell is a program which provide as user interphase with a shell user can type the command and run programs on Unix system.

Shell Meta Characteristics–

- Meta characters are the special characters such %, ?, () and – etc…
- These characters are used as wild cart.

Shell Variables & Scripts–

- Shell variable is a special variable that is set by shell and is required by the shell in order to function correctly.
- Shell script defines only those environment variables that are needed by the program that it runs.

Environment–

- Environments are those where we can run our scripting.

Integer Arithmetic & String Manipulation–

- The following arithmetic operators are supported by Bourne Shell. Assume variable a holds 10 and variable b holds 20 then