

Date  
23/01/24

## UNIT - 1

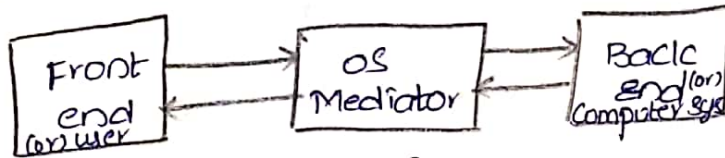
### operating system overview & structures

\* operating system: it is a collection of programs and utilities.

→ it acts as an interface or a mediator.

→ This interface helps to transfer any amount of information from back end to front-end.

[transfers information from user to computer system]



### \* OS Functions / Services

→ program creation.

→ program execution

→ I/O Devices

→ Error Detection & Correction

→ Resources Allocation

→ Communication

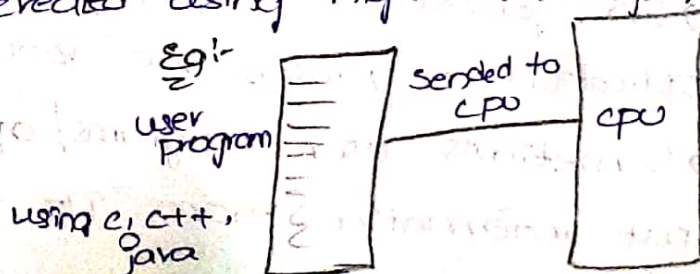
→ Protection & Security.

### \* program creation

→ Any user wants to create an application program, he depends upon two modes they are editor & debugging.

→ These two modes are provided implemented with the help of operating system.

→ Programs are created using high level language like C, C++, Java.



```
import java.lang.*
```

```
class student
```

```
{
```

```
String sname;
```

```
int SRLNO;
```

```
Student()
```

```
{
```

```
sname = "Meghana";
```

```
SRLNO = 10;
```

```
}
```

```
void display()
```

```
{
```

```
System.out.println("The student name is" + sname);
```

```
System.out.println("The student SRLNO is" + SRLNO);
```

\* program execution:

→ Any user successfully completed program/process/task will be moved from CPU to main memory

→ To properly loading of the program information from CPU to main memory using loader

→ Sometimes process/program required input information

→ Required information is properly brought from I/O

devices to CPU & these corresponding work roles are completed with the help of

\* I/O Devices Functions

→ In general to properly completing of any application or process we require I/O.

→ Sometimes user programs/applications require input information from I/O devices to CPU for



the purpose of execution.

→ Input/output information is initially available in I/O devices.

### \* Error detection & correction

→ Error detection is the process of finding different errors in an application

→ Different errors are maintained by different applications. They are.

1, Memory management error

2, CPU error

3, power failure

4, I/O dev errors

→ These errors are rectified with the help of different modes. They are.

1, Page Replacement Algorithm / Memory Management routine.

2, Swapping for memory

3, process management / process management routines for CPU

4, check points for power failure.

5, I/O routines for I/O devices.

### \* Resource Allocation

→ Resource Allocation means gathering different resources from the network environment

→ Resource Allocation maintains different

1) CPU

2) main memory

3) I/O devices

4) Secondary Memory.

\* communication :-

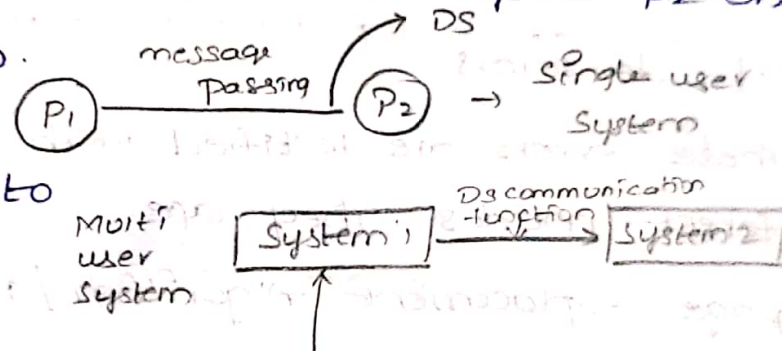
\* communication is the process of exchanging any information from one process to another process.  
It is possible to do using two criteria.

1. Single user
2. Multi user

\* Single user system is used to support successfully completion of  $P_1$  &  $P_2$  processes under single system.

\* Multi user system provides 2 different computers.  $P_1$  computed under 1st system  $P_2$  under another system.

\* one system into transfers to another system. 2.



\* Protection & Security

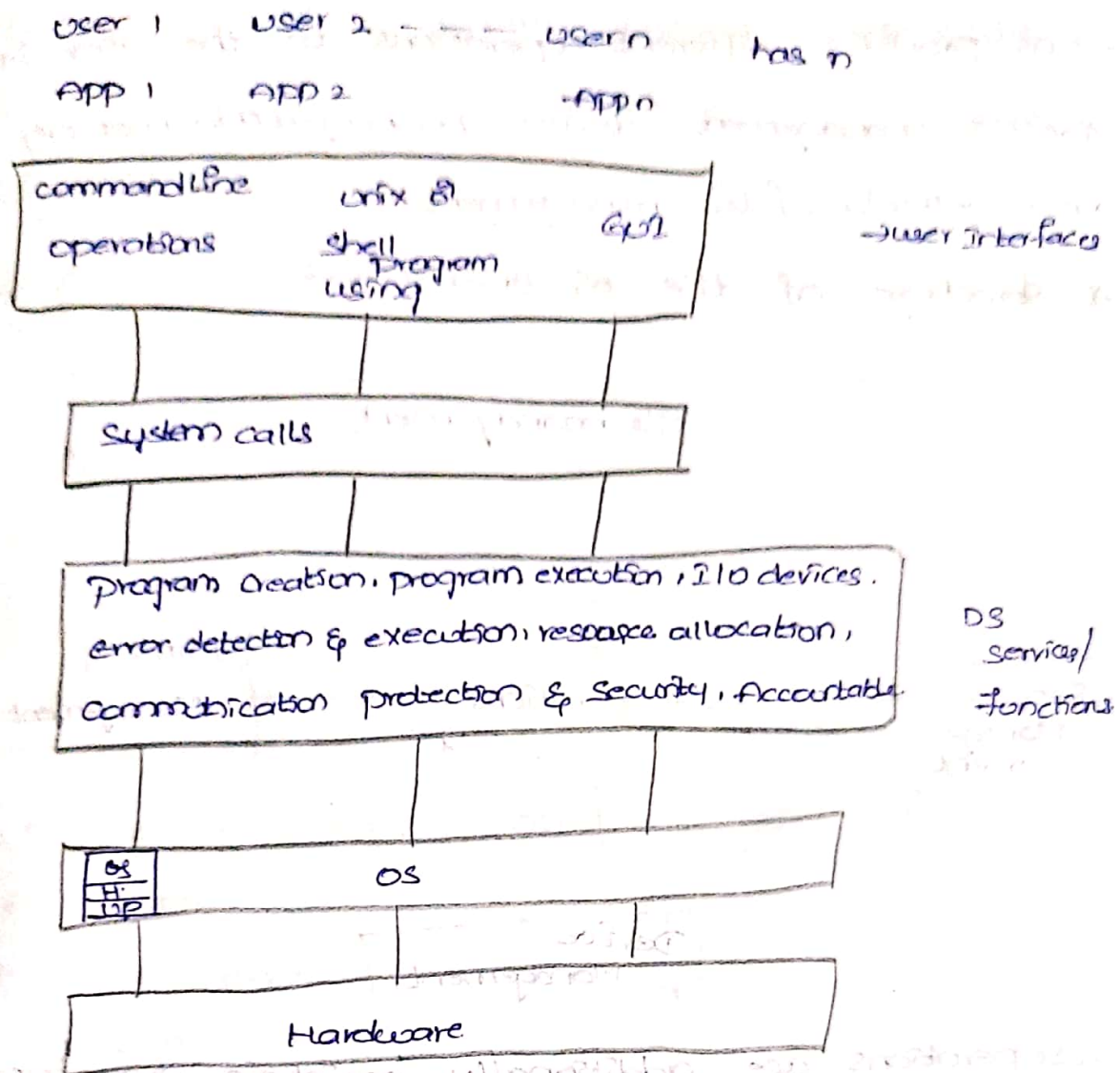
\* protection is the process of properly monitoring of internal functions i.e., different functions of CPU, main memory, I/O devices, secondary memory.

\* Security is the process of properly monitoring of external information of OS.

\* Example :- password mechanism, external communication modes [modems, printers, network adaptors], authentication.



## OS structure / system structure



→ User 1, User 2, User 3 --- can create different application

Programs using different interfaces. They are

→ Command line operations - Disk operating systems provides command line operations to the users.

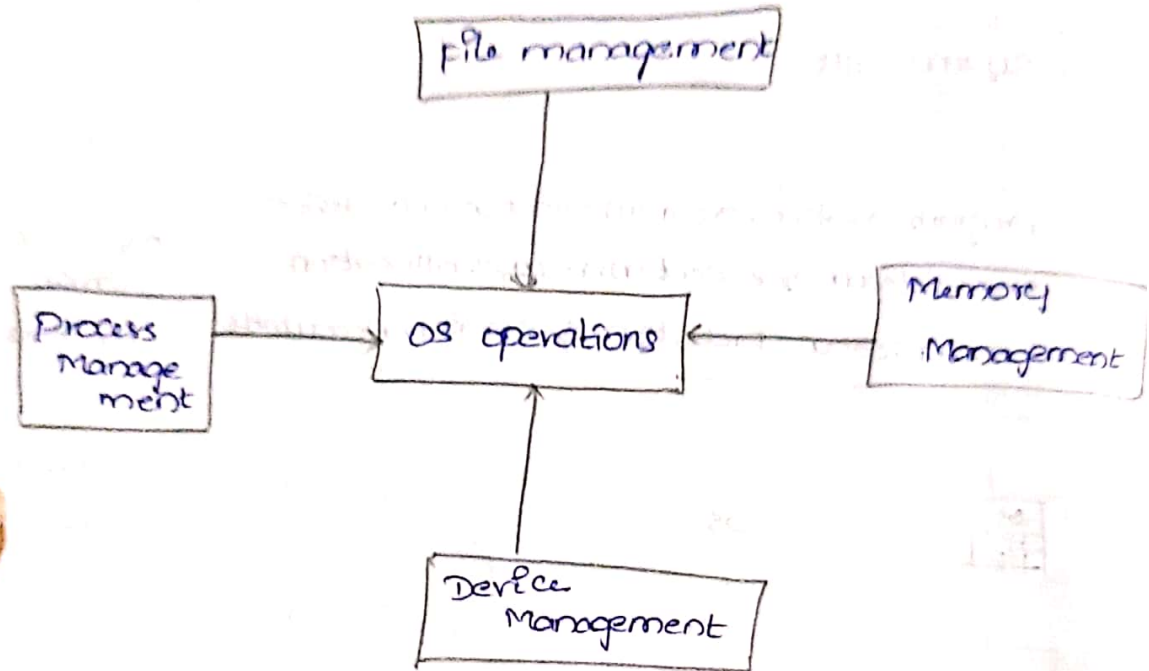
→ Unix (or) shell programming - Unix (or) shell programming provides batch processing system.

→ GUI - GUI provides different icons to the users. These icons are used to create different graphical shapes.

## OS operations

→ OS provides different operations to the user i.e. process management, device management, memory management, file management.

\* Structure of the OS operations



→ OS operations additionally maintain two types that are:

- 1, Communication Management
- 2, Information management

→ management is the process of

- 1, creating of a data
- 2, TO inserting of a data
- 3, TO updating of a data
- 4, TO Modifying of a data
- 5, TO Deleting of a data.

1, process management :- Process is a execution of entire application under the single processor or multiprocessor.

→ Process Management supported to

- 1, creating of a process (using editor & debugger)
- 2, execution of any process depends on different scheduling algorithms. (FCFS, SJF, SRTF, PRIORITY, ROUND ROBIN)

2) Device Management :-

→ It mainly supported different types of devices.

1, keyboard

2, Mouse

3, Monitor

4, Hardware (cpu, memory, I/O devices)

→ Device Management also supported diff types of system calls.

Eg:- 1, Set Device

2, Read Device

3, Write Device

4, Release Device

→ It is done between 1, cpu & I/O device

2, cpu & Main memory

3, Main memory & Secondary device

4, Main memory & I/O devices

3) Memory Management :-

→ It is mainly applied on main memory & Secondary memory.

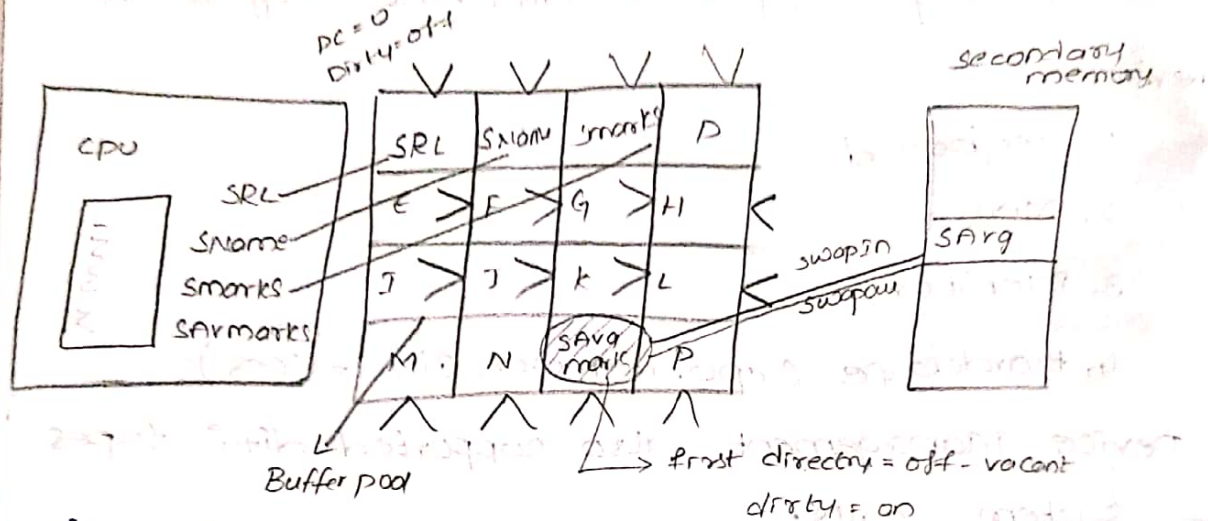
→ It is supported different types of softwares and techniques.



- 1, Buffer Manager
- 2, Swapping
- 3, Page replacement algorithms

↓  
scheduling

→ According to buffer manager available main memory is divided into no. of equal partitions



(According to main)

→ Each & Every partition should maintain two variables

- 1, Pincount
- 2, Dirty.

→ Initially pincount set to '0' and Dirty variable set to off state.

→ Swapping process supported two operations are

- 1) Swap In
- 2) Swap out

→ Page Replacement algorithms (FIFO, LRU, MRU, LFU, CLK, etc.) based on pincount values.

→ Swapping process is done based on page replacement algorithm using swap in & swapout operations.



## \* File Management :-

→ file is a collection of interrelated logical records

→ Record is a collection of attributes or fields

→ Record  $R = \text{attribute } A_1 + \text{attribute } A_2 + \dots + \text{attribute } A_n$

→ file is created with the help of using high level languages or database languages.

→ file provides a different operations are

1, create

2, insert

3, update

4, Delete

5, open

6, close

7, save

→ files are stored temporarily in the main memory

→ files are stored permanently in the secondary memory.

→ file management supported different criteria are.

1, disk scheduling

2, swapping process

File [is maintain] provides diff types of templates structures to the users.

File creation using high level languages

|                       | Attribute 1 | Attribute 2 | Attribute 3 |
|-----------------------|-------------|-------------|-------------|
| Record R <sub>1</sub> | R           |             |             |
| Record R <sub>2</sub> |             |             |             |
| Record R <sub>3</sub> |             |             |             |
| Record R <sub>4</sub> |             |             |             |

→ File is Above table or file is created using create command in database language.

Create table name (Attribute 1,)

Attribute 1 datatype (size),

attribute 2, datatype (size) :-

attribute n, datatype (size) :-

Eq: Create Table TCS (

SRLNO ~~int~~ number [7],

Sname varchar [10],

Smarks number [5,2]

### \* Communication Management

→ It supported to transfer any type of messages from one device to another device.

→ It supported different operations are

- 1, create message
- 2, Send message
- 3, Receive message
- 4, Delete message
- 5, Update message

→ It also supported establishing of communication path between sender & Receiver.

→ To destroy the path b/w sender & Receiver.

### \* Information Management

→ It is the process of Exchanging of any amount of data from one device to another device.

→ It using different types of operations are

- 1, Set date of the system
- 2, Set time of the system

Insert data

Delete data

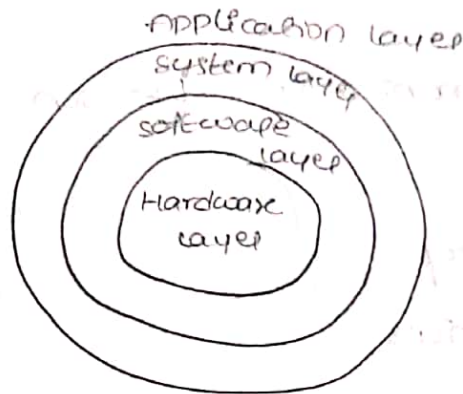
Insert time

Delete time.

## \* OS Structure

→ OS supported 4 layers that are

- 1, Application layer
- 2, System layer
- 3, Software layer / OS operating system
- 4, Hardware layer / Secondary memory layer.



## \* Application layer

→ It supported to create any application

programs (or) processors of the user.

→ Processors are created with the help of using high level language (C, C++, Java, .net, --- etc).

→ It mainly supported two provisions to the user.

1, Text editors

2, Debuggers

## \* System layer

→ It supported different functionalities are:

1, compiler

2, linker

3, loader

4, Assembly

5, Pre-defined methods / Pre defined functions / packages /

Header files

→ System programmes are supported to user application



## Programs.

### \* Software layer / OS layer:-

→ It supported different functionalities to the user.

- 1) program creation
- 2) Program execution
- 3) I/O devices
- 4) Error checking communication / detection
- 5) Accountability
- 6) Protection & Security
- 7) Different OS operations

i) process management

ii) file management

iii) communication management

iv) Device management

v) Information management

vi) Memory management

### \* Hardware layer:-

→ It supported to user using different functionalities

- 1) Swapping process (swap in, swap out)
- 2) Disk scheduling algorithms
- 3) Raid levels
- 4) Buffering process

→ Hardware layer is mainly used to any user applications permanently stored under the secondary memory

## \* System calls

→ system calls acts as interfaces. Interfacing is used to establish communication path between process & OS.

→ system calls are written using assembly level language. Sometimes they are also written in high level language.

→ System calls are implemented using two operational modes.

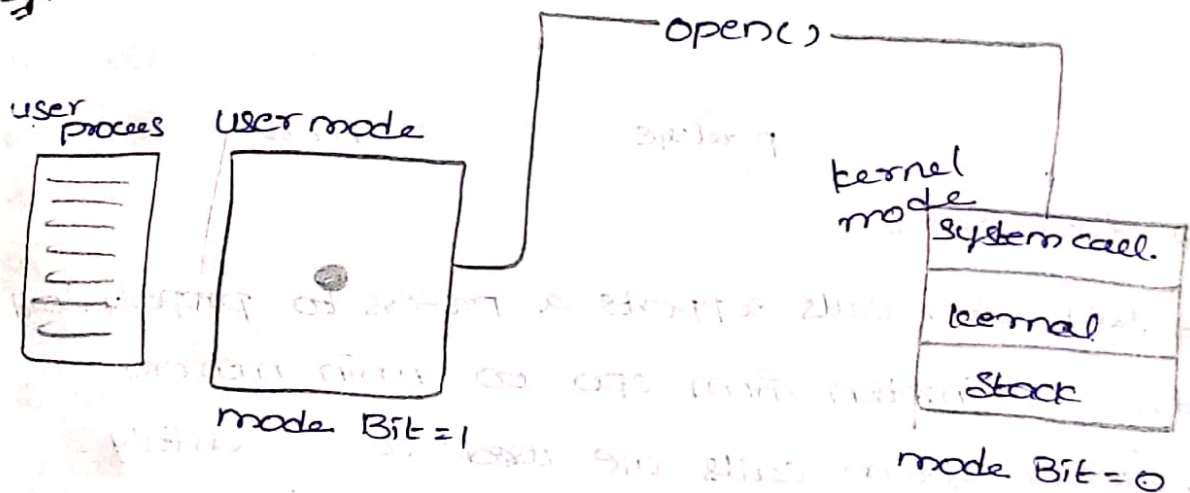
1) user mode

2) kernel mode

→ These two modes are maintained on the same mode.

→ This mode provides two integer values 0 (zero), 1 (one)

eg:



\* System calls are divided into 5 types. They are

1, process management

2, file management

3, Device Management

4, Information Management

5, Communication Management

## \* process control.

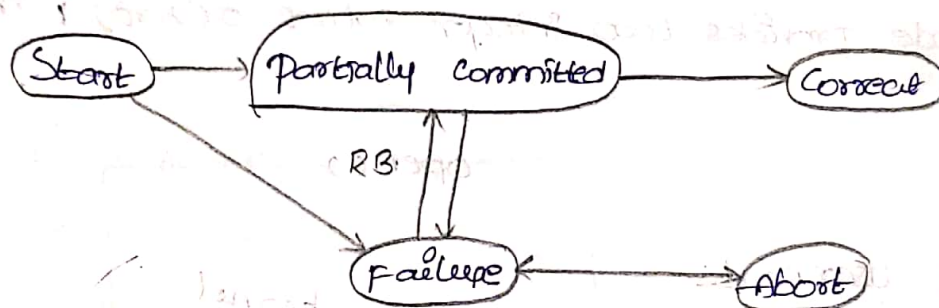
→ A user can successfully execute a process under CPU/processor.

→ process control provides different system calls to the user.

1) End, Abort

→ End - These calls can support any process to successfully execute.

\* Abort - these calls support failure process (or) collapsing process.



2. Load, Execute

→ Load system calls support a process to properly load the information from CPU to main memory.

→ Execute system calls are used to successfully complete a process under CPU.

3. Create, Terminate process:-

→ 3. Create system calls are used to create the processes successfully.

→ Terminate system calls are used to delete the process from the main memory.

4) Set process attribute

→ To properly add the available attributes to the process using set process attributes



5. Get process attribute :-

→ To properly retrieve the attributes from the process.

6. Allocate memory

7. Free memory

8. Wait for time

9. Wait for event.

2. File Management :-

→ File is a collection of records. Records means fields / attributes.

→ File management is used to properly arrange (or) manage any given files. It provides different system calls to the user.

1. Create a file

2. Delete a file

3. open()

4. close()

5. Read, write, Reposition

6. Set file attributes

7. Get file attributes

3. Device Management :-

→ Device Management provides I/O devices, processor, memory.

→ To properly establish of communication path

1. Cpu to I/O devices

2. I/O to cpu devices

3. Main memory to cpu

4. Cpu to main memory.

→ Device management provides different system calls to the users. They are.

- 1) Request device
- 2) Release device
- 3, Read, write, Reposition
- 4, Set device attributes
- 5, Get device attributes
- 6, Attach devices
- 7, Detach devices.

\* Information management

→ To properly send any type of information into the system.

→ Information management provides different system calls to the users.

- 1, Set time, Get time
- 2, Set date, Get date
- 3, Set system data, Get system data
- 4, Set process files, Get device attributes
- 5, Get process files, device attributes.

\* Communication management

→ It is the process of properly establishing a communication path between client and server. It cares about properly sending information from client to server.

→ communication management provides different system calls to the users. They are.

- 1) create, delete connection.
- 2) send message.
- 3, Receive message

4) Transfer Status Information.

5) Attach Remote Files

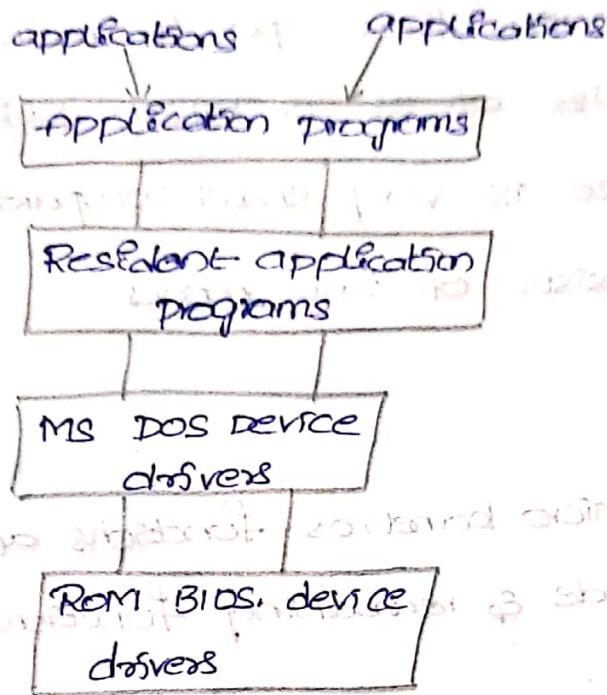
6) Detach Remote Files.

\* System Structure

→ These are different types of system structures.

1. MS DOS system structure.

2. UNIX system structure / traditional system structure.



\* Application programs

→ Application programs are used by different users using different high level languages like C++, Java

→ Sometimes required user applications are converted into perfect applications.

\* Resident Application programs

→ These are used to convert user applications to perfect applications using system programming functions like assembler, compiler.

\* MS DOS device drivers

→ These provides I/O devices to user applications.



## \* ROM BIOS Device Drivers

→ They provide loader functions to the user. loader functions are responsible for loading process information from cpu to main memory.

## \* Micro Kernel

1. It is a software process.

2. It loads information from cpu to main memory.

3. The smallest size of os is MICRO KERNEL.

4. Micro kernel size is very small compared to kernel.

5. Micro kernel consists of two modes.

i. user mode

ii. kernel mode.

6. According to micro kernel, os functions are completed under kernel mode & remaining functions are done under user mode.

7. Firstly used as MACH

\* ~~PA1~~