UNIT: 01

BASICS OF OPERATING SYSTEM

Definition:-

An operating system is also known as OS. It is a software program that enables the computer hardware to communicate and operate with the computer system.

Functions of operating system:-

The most important functional components of the operating system are as follows:

1) Process Manager:- The process manager can be considered the most important component of operating system. It carries out several services such as creating, suspending (blocking), executing, terminating and destroying processes.

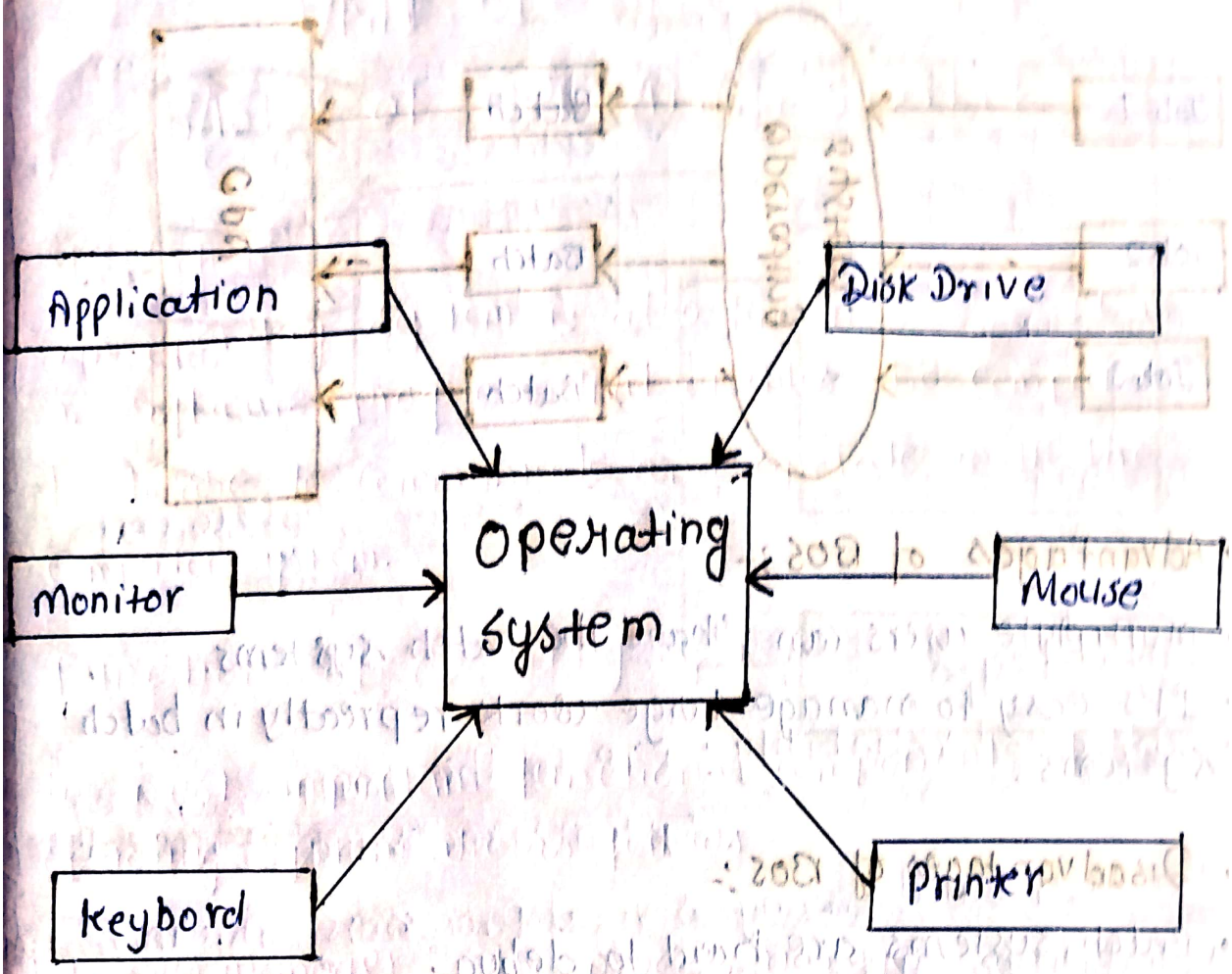
With multiprogramming several active processes are stored and kept in the memory at the same time. If there is only one CPU in the system then only one process can be in execution at a given time the other process is waiting for CPU service. The decision regarding which process to execute next is taken by the scheduler. The dispatcher allocates the CPU to a process and deallocates the CPU from other process. This switching of the CPU from one process to another process is called context switching and is considered overhedd time.

2) Memory Manager :- The memory manager controls the allocation and deallocation of of memory. It imposes certain policies and mechanisms for memory management. This component also includes Policies and mechanisms for memory protection.

3) Resource Manager :- The resource manager facilitates the allocation and deallocation of resource to the requesting processes. functions of this component include the maintenance of resource tables with information such as the number of resources of each type that are available and the number of resource of each type that have been allocated to specific Processes.

4) File Manager :- The file manager allows user to and processes to create and delete files and directories. In modern Operating System, files are associ-ated with mass storage devices such as magnatic tapes and disks.

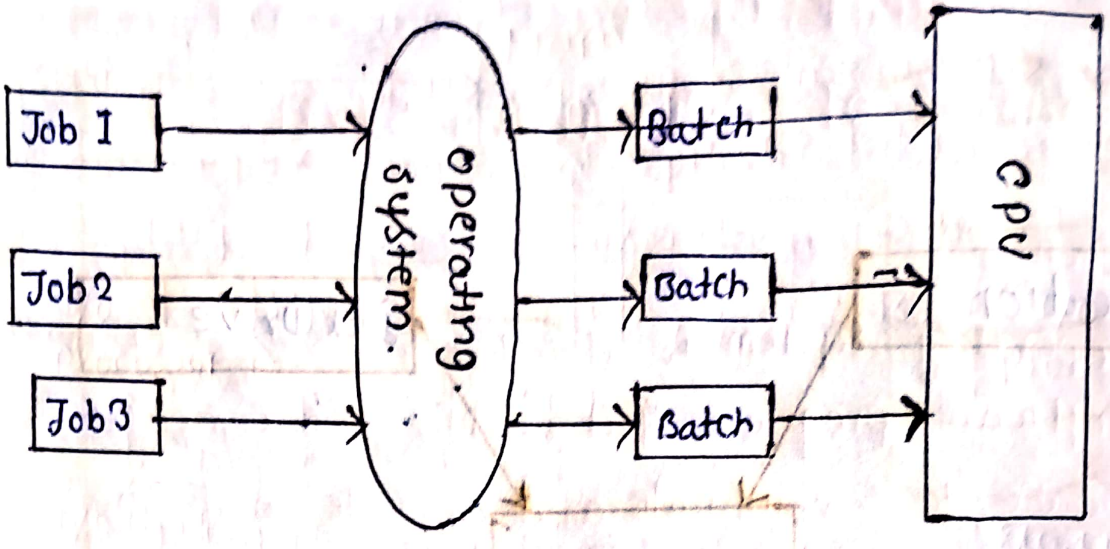
5) Device manager :- for the various devices in the computer system, the device manager provides an appropriate level of abstraction to handle system devices. For every device type a device driver program is included in the O.S. The devices manager operates in combination with the resource manager and file manager.



TYPES OF OPERATING SYSTEM

1) Batch operating System :-

This type of operating system does not interact with the computer directly. There is an operator which takes similar jobs having the same requirements and groups them into batches. It is the responsibility of the operator to sort jobs with similar needs. Batch Operating System is designed to manage and execute a large number of jobs efficiently by processing them in groups.



- Advantages of BOS :-
- multiplule users can share the batch systems.
- It's easy to manage large work repreatly in batch systems.

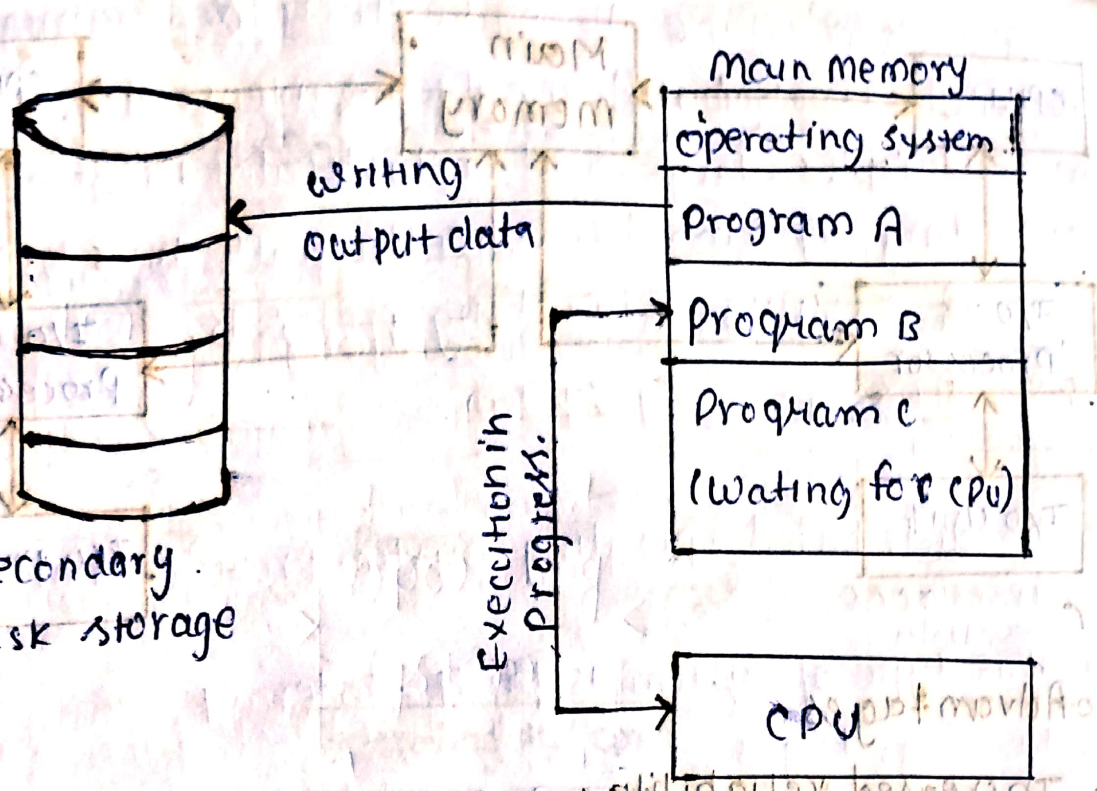
- Disadvantages of BOS :-
- Batch systems are hard to debug.
- It's sometimes costly.
- It's difficult to accurately predict the exact time required for a job to complete while it is in the queue.

2) Multi-programming operating system :-

Multiprogramming is an extension to batch processi-ng where the CPU is always kept busy. Each process needs two types of system time. CPU time, and I/O time.

In a multiprogramming environment, when a process does it's I/O, The CPU can start the execution of other processes. Therefore multiprogramming improves the efficiency of the system.

EXAMPLE :- THE

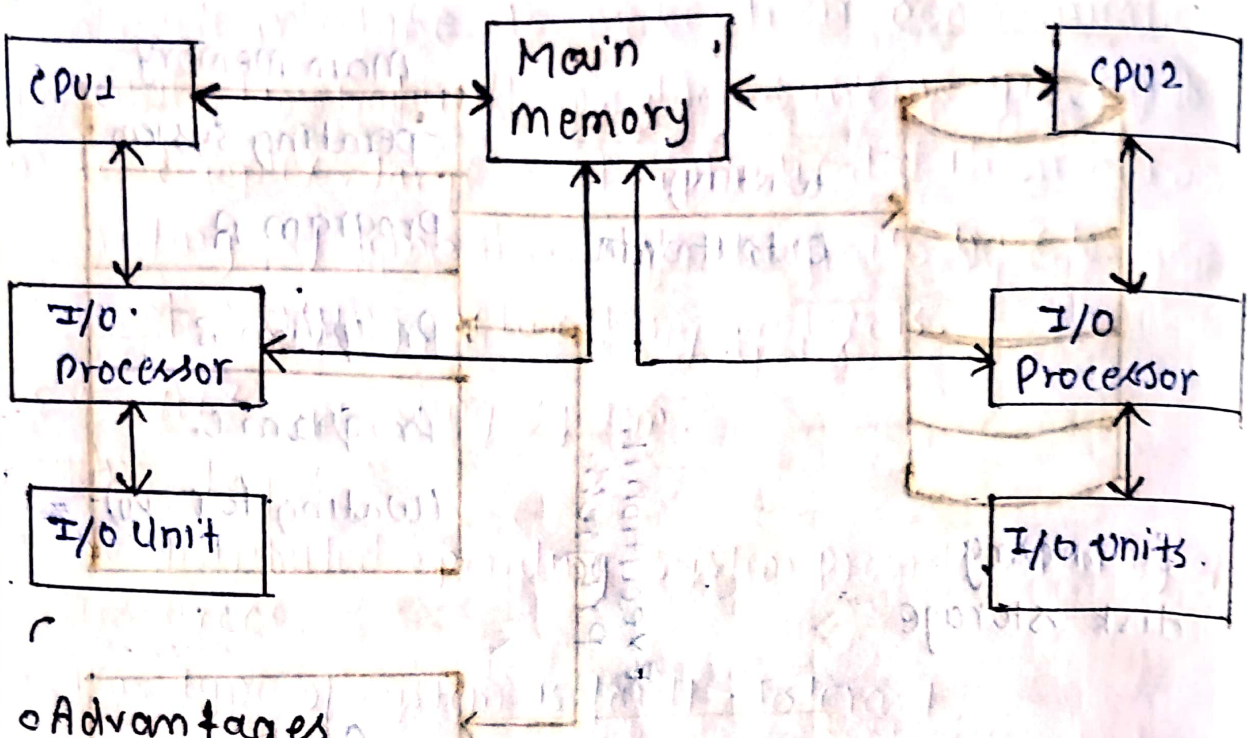


- Advantages
- Throughout the system, it increased as the CPU always had one program to execute.
- Response time can also be reduced.
- Disadvantages
- multiprogramming system provide an environment in which various systems resources are used efficiently, but they do not provide any user interaction with the computer system.

3) multiprocessing operating system.

In multiprocessing Parallel computing is achieved. There are more than one processors present in the system which can execute more than one process at the same time. This will increase the throughput of the system.

→ Ex:- CTSS.



- Advantages

- Increased reliability: Due to the multiprocessing system; processing tasks can be distributed among several processors. This increases reliability as if one processor fails, the task can be given to another processors for completion.
- Increased throughout:- As several processesors increase, more work can be done in less.

- Disadvantages :-

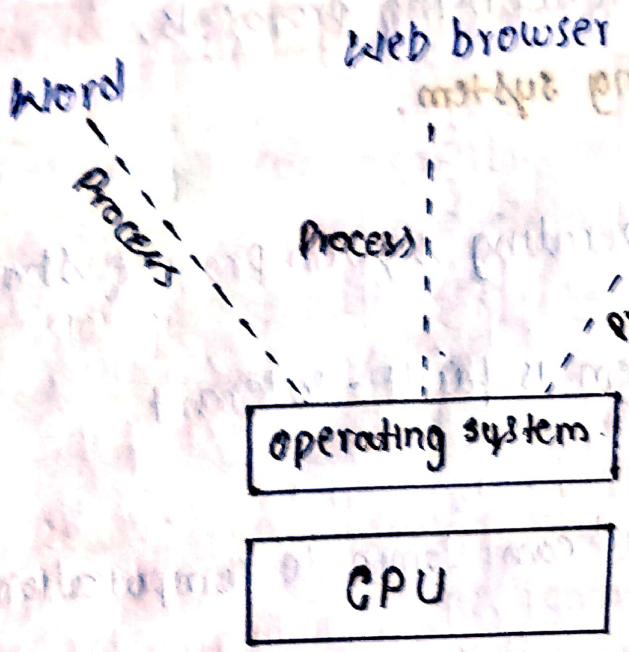
- Multiprocessing operating system is more complex and sophisticated as it takes care of multiple CPUs simultaneously.

4) Multitasking Operating system :-

The multitasking operating system is a logical extension of a multiprogramming system that enables multiple programs simultaneously. It allows user

to perform more than one computer task at the same time.

example:- WINDOW.



- Advantages.
- This operating system more suited to supporting multiple user simultaneously.
- The multitasking operating systems have well-defined memory management.

- Disadvantages
- The multiple processors are busier at the same time to complete any task in a multitasking environment, so the cpu generates more heat.

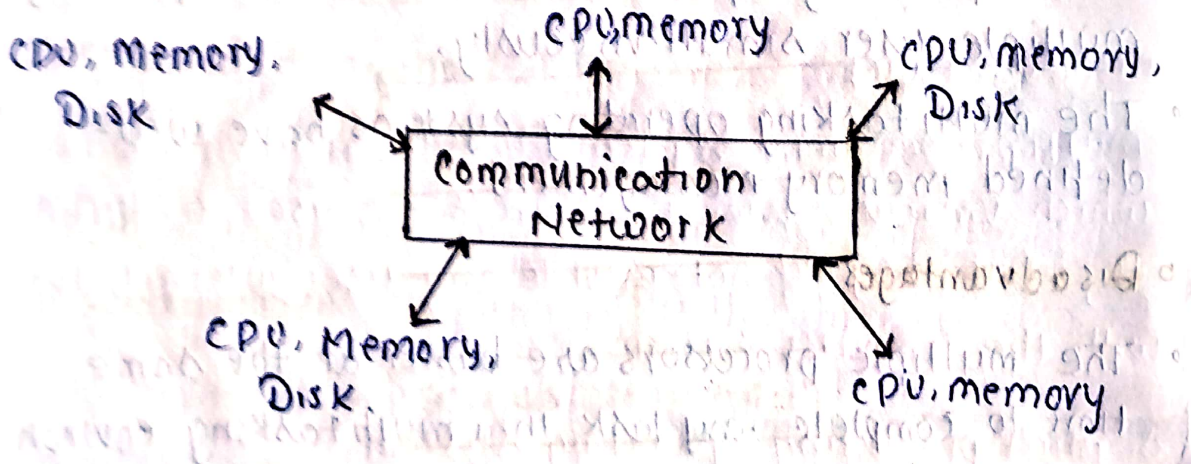
5) Distributed Operating System:-

The distributed operating system is not installed on a single machine, it is divided into parts, and these parts are loaded on different machines. A part of distributed operating system is installed

on each machine to make their communication feasible. Distributed operating systems are much more complex, large, and sophisticated than network operating systems because they also have to take care of varying networking protocols.

Ex:- AIX operating system.

- Advantages.
- The distributed operating system provide sharing of resources.
- This type of system is fault-tolerant.
- Disadvantages.
- Protocol overhead can dominate computation cost.



6) Real Time Operating system.

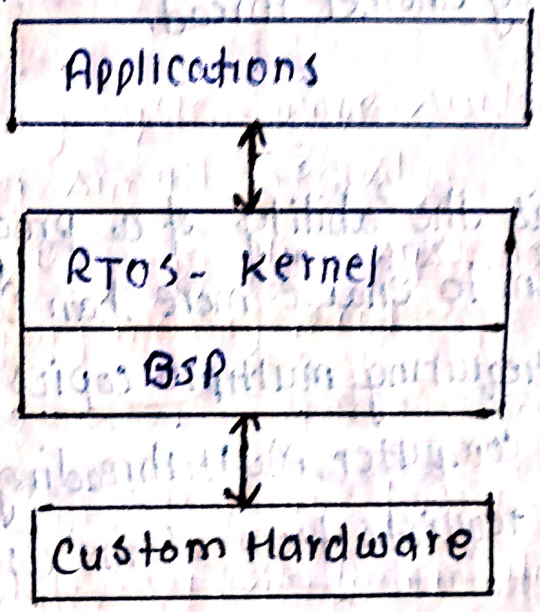
These types of operating systems serve real-time systems. The time interval required to process and respond to input is very small. This time interval is called response time. Real time operating are used when there are time requirements that

are very strict like missile systems, air traffic control systems, robots etc.

Types of Real-Time Operating systems.

(i) Hard Real-Time systems: Hard real time OSs are meant for applicat-on where time constraints are very strict and even the shortest possible delay is not acceptable. These systems are built for saving life like automatic parachutes or air bags, which are required to be readily available in case of an accident.

(ii) Soft Real-Time system: These OSs are for applications where time constraint is less strict.



- Advantages
- Easy to layout, develop and execute real-time applications under the real-time operating system.
- In a Real-time operating system, the maximum utilization of device and systems.

- Disadvantages:
- Real-time operating systems are very costly to develop.
- Real-time operating systems are very costly and can consume critical CPU cycles.

Threads

An Application consists of one or more processes. A Process, in the simplest term is an executing program. One or more threads run in the context of the process.

A thread is the basic unit to which the operating system allocates processor time. A thread can execute any part of the process code, including parts currently being executed by another thread.

Multithreading

Multithreading is the ability of a program or an operating system to enable more than one user at a time without requiring multiple copies of the program running on the computer. Multithreading can also handle multiple requests from same user.

Difference between Multitasking and multithreading.

The following table highlights the important differences between multitasking and multithreading.

Multitasking	Multithreading.
In the multitasking, the users are allowed to perform multiple tasks by CPU.	In multithreading multiple threads are created from a process.
Multitasking involves CPU switching between tasks.	Multithreading involves CPU switching between the threads. Due to power of this the power of Computer is increased.
In multitasking, the processes share separate memory locations.	In multithreading, the processes are allocated same memory.
Multitasking involves multiprocessing.	Multithreading doesn't involve the multiprocessing.
The CPU is provided to execute many tasks at a time.	The CPU is provided so that multiple threads can be executed at a specific time.
The process don't share its own resources.	The multiple threads share the same resources.

UNIT-02

OPERATING SYSTEM COMPONENTS.

An operating system is a large and complex system that can only be created by partitioning into small parts. These pieces should be a well-defined part of the system, carefully defining inputs, outputs, and functions.

The components of an operating system play a key role to make a variety of computer system parts work together. There are the following components of an operating system, such as:-

1. Process Management.
2. File Management.
3. Network Management.
4. Main memory management
5. Secondary storage management.
6. I/O Device Management
7. Security management.
8. Command interpreter system.

1. Process Management

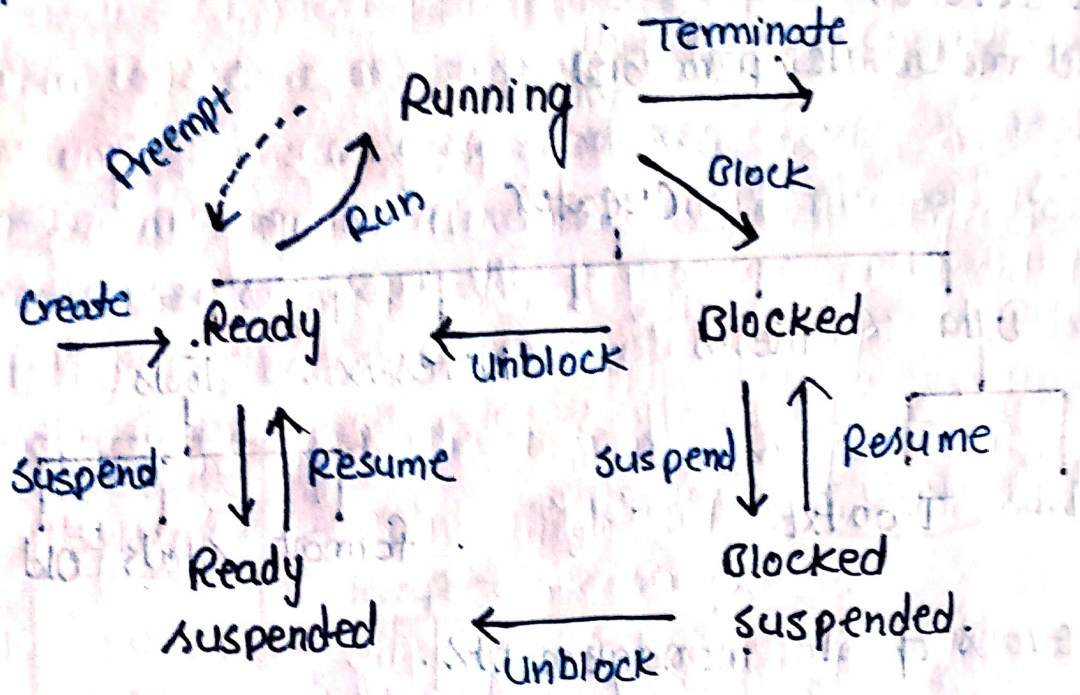
The Process management component is a procedure for managing many processes running simultaneously on the operating system. Every running software application program has one or more processes associated with them.

For example:- When we use a search engine like Chrome, there is a process running for the browser

Program.

Process Management keeps processes running efficiently. It also memory allocated to them and shutting them down when needed.

The execution of a process must be sequential so, at least one instruction should be executed on behalf of the process.



Functions of Process Management

Here are following functions of process management in the operating system. such as.

- Process creation and deletion.
- suspension and resumption.
- synchronization process.
- communication process.

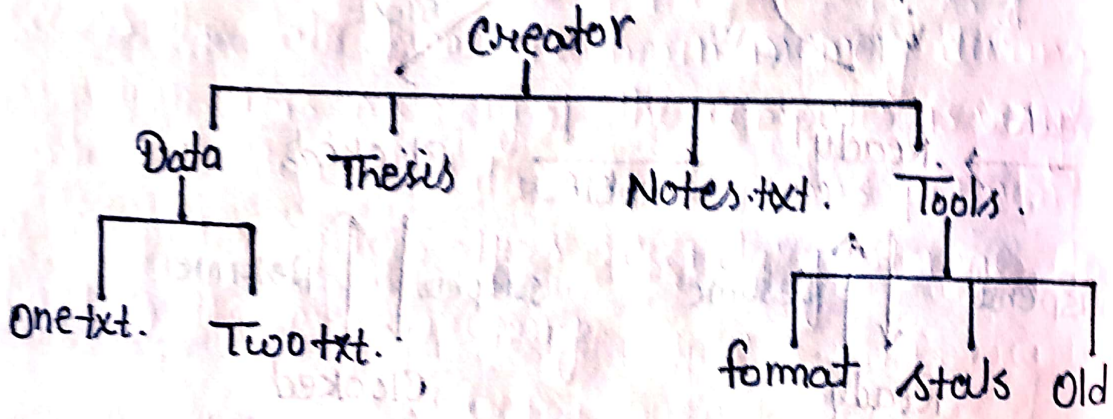
-: FILE MANAGEMENT :-

A file is a set of related information defined by its creator. It commonly represents programs (both sour

and object forms) and data. Data files can be alphabetic, numeric, or alphanumeric.

file are used for long-term storage. files are used for both input and output. every operating system Provides a file management service. The system call for file management includes.

- file creation
- file deletion
- Read and write operations.



functions of file managements.

The Operating system has the following imporatant activities in Connection with file management.

- file and directory creation and deletion.
- for manipulating files and directories.
- Mapping files onto secondary storage.
- Backup files on stable storage media.

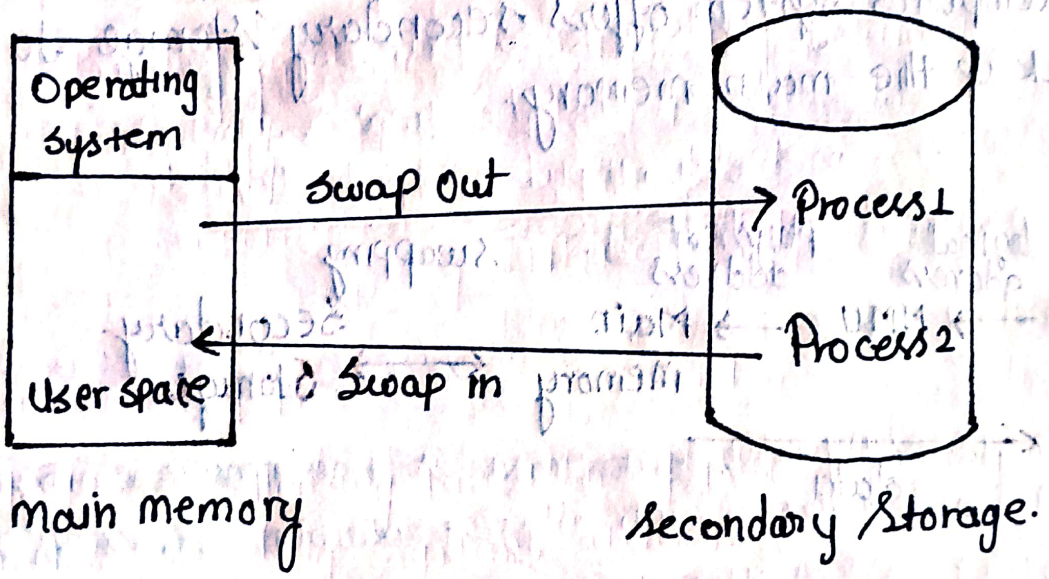
-. Main Memory Managements:-

Main memory is a large array of storage or bytes, which has an address. The memory management Process is conducted by using a sequence of reads

or ibrites of specific memory addresses.

It should be mapped to absolute addresses and loaded inside the memory to execute a programs. The selection of a memory management method depends on several factors.

However, It is mainly based on the hardware design of the system. Each algorithm requires corresponding hardware support. main memory offers fast storage that can be accessed directly by C.P.U. It is costly and hence has a lower storage capacity. However, for a program to be executed, it must be in the main memory.



functions of Memory Management.

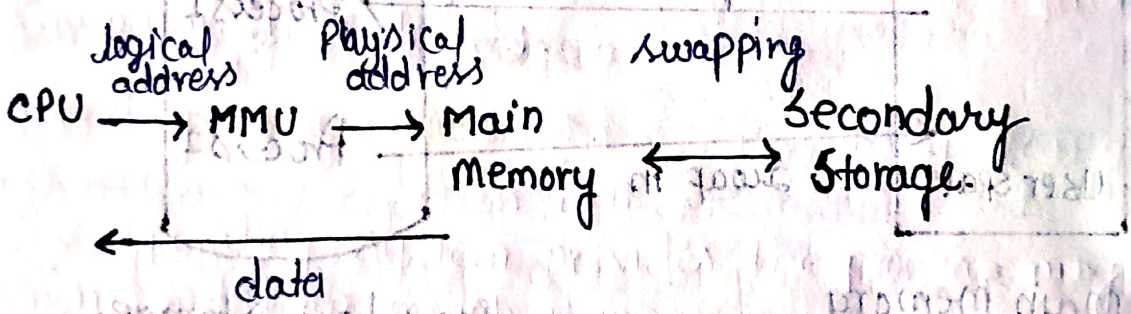
An operating system performs the following functions for memory management in the Operating system.

- It helps you to keep track of primary memory.
- Determine what part of it are in us by whom, what part is not to use.

- In a multiprograming system, the OS decides which process will get memory and how much.
- Allocates the memory when a process requests.
- It also de-allocates the memory when a process no longer requires or has been terminated.

Secondary-Storage Management

The most important task of a computer system is to execute programs. These programs help you to access the data from the main memory during execution. This memory of the computer is very small to store all data and programs permanently. The computer system offers secondary storage to back up the main memory.



Today modern Computers use hard drives/SSD as the primary storage of the both programs and data. However, the secondary storage management also works with storage devices, such as USB flash drives and CD/DVD drives. Programs like assemblers, and compilers are stored on the disk until it is located into memory, and then use the disk is used as a source and destination for processing.

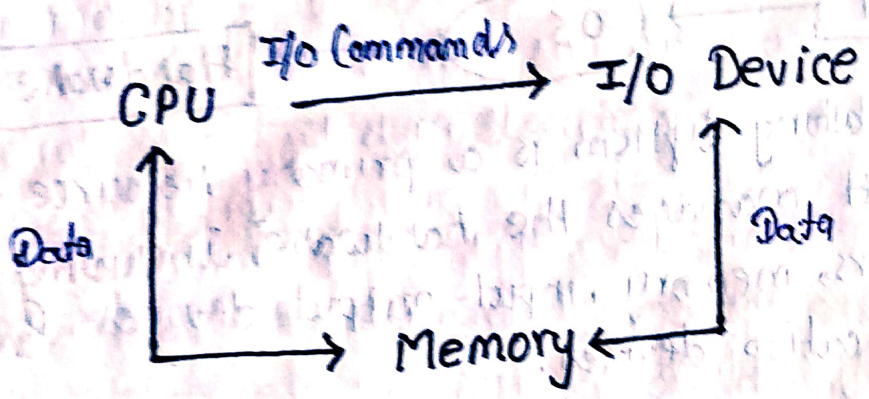
functions of Secondary storage management

Here are some major functions of secondary storage management in the operating system.

- Storage allocation.
- Free space management.
- Disk scheduling.

I/O Device Management.

One of the important use of an Operating System that helps to hide the variations of specific hardware devices from the user.



functions of I/O management.

The I/O management system offers the following functions, such as.

- It offers a buffer caching system.
- It provides general device driver code.
- It Provide drivers for particular hardware devices.
- I/O helps you to know the individualities of a specific device.

Operating system services

An operating system provides an interaction between the user and computer hardware. A user is a person sitting at the computer terminal concerned about the application rather than the architecture of the computer. The user never interacts with the hardware directly. To get the services of the hardware, he has to request through the services of hardware, or operating system.

The operating system is a primary resource manager. It manages the hardware, including processesors, memory, input-output devices and communication devices.

Operating system services.

The operating system provides the programming environment in which a programmer works on a computer system. The user program requests various resources through the operating system. The operating system gives several services to utility programmers and users. Application access these services through application programming interfaces or system calls. By invoking those interfaces, the application can request a service from the operating system, pass parameters, and aquire the operation outcomes.

- Following are the services provided by an operating system.
- Program execution.
- Control Input/output devices.
- Program creation.
- Error detection and Response.
- Accounting.
- Security and protection.
- File management.
- Communication.

-: Program execution :-

To execute a program, several tasks need to be performed. Both the instructions and data must be loaded into the main memory. In addition, input-output devices and files should be initialized, and other resources must be prepared. The Operating structure handle these kinds of tasks. The user now no longer should fear the reminisence allocation or multitasking or anything.

-: Control Input/output devices :-

As there are numerous types of I/O deveses within the computer system, and each I/O device calls for its own precise set of instructions for the operation. The operating system hides that info with the aid of presenting a uniform interface.

Thus, it is convenient for programmers to access such devices easily.

-: Program Creation :-

The operating system offers the structure and tools, including editors and debuggers, to help the Programmer create, modify, and debugging programs

-: Error Detection and Response :-

An Error in a device may also cause malfunctioning of the entire device. These include hardware and software errors such as devices failure, memory error, division by zero, attempts to access forbidden memory locations, etc. To avoid error, the operating system monitors the system for detecting errors and takes suitable action with at least impact on running applications.

While working with computers, errors may occur quite often. Errors may occur in the :-

(i) Input/output devices :- for example, connection failure in the network, lack of of paper in the printer etc.

(ii) User programs :- For example, attempt to access illegal memory locations, divide by zero, use too much CPU time etc.

(iii) Memory hardware :- for example, memory error, the memory becomes full.

To handle these errors and other types of possible errors, the operating systems takes appropriate action and generates messages to ensure correct and consistent computing.

-: Accounting :-

An Operating device collects utilization records for numerous assets and tracks the overall performan parameters and responsive time to enhance overall performance. These personal records are beneficia for additional upgrades and tunning the devices to enhance overall performance.

-: Security and Protection :-

operating device affords safety to the statistics and packages of a person and protects any inter--ference from unauthorized users. The safety featur counters threats, which are published via way of individuals out of doors the manage of the running device.

for example:- When a user downloads something from the internet, that program may contain malicious code that may harm the already existing programs. The operating system ensures that proper checks are applied while dowloading such programs.

If one computer system is shared (almost) amongst a couple of users, then the various processes must be protected from another instru-

-ction. For this the operating system provides various (instruction) mechanisms that allows only those processes to use resource that have gained proper authorization from the operating system. The mechanism may include providing unique users ids and passwords to each user.

-: File Management :-

Computers keep data and information on secondary storage device like magnetic tape, magnetic disk, optical disk etc. Each storage media has its capabilities like speed, capacity, data transfer rate, and data access methods.

For file management, the operating system must know the types of different files and the characteristics of different storage devices. It has to offer the proportion and safety mechanism of documents additionally.

-: Communication :-

The operating system manages the exchange of data and programs among different computers connected over a network. This communication is accomplished using message passing and shared memory.

Communication may be implemented by two methods, either by shared memory or by message passing.

-. Resource Management :-

In case of multi-user or multi-tasking environment resources such as main memory, CPU cycles and files storage are to be allocated to each user or job. following are the major activities of an operating system with respect to resource management.

- The OS manages all kinds of resources using schedulers.
- CPU scheduling algorithms are used for better utilization of CPU.

-. Accounting :-

In an operating system, accounting refers to the Process of tracking and recording the usage of system resources by users or processes. The goal is to monitor system performance, resource allocation and user activities. It helps administrators understand how resources like CPU time, memory, disk space, and network bandwidth are used.

Key functions of accounting in an OS include :-

1. Resource usage tracking - The OS tracks how much of various resources (CPU, memory, I/O, etc.) are used by individual users or processes.

3 PROCESS

Basically, a process is a simple program.

An active program which running now on the operating system is known as Process. The process is the base of all computing things. Although Process is relatively similar to the computer code but, the method is not the same as computer code. A process is a "active" entity.

Process in an operating system.

A process is actively running software or a computer code. Any procedure must be carried out in a precise order. An entity that helps in describing the fundamental work unit that must be implimented in any is system is referred to as a process.

In other words, we create computer programs as text files that, when executed, create processes that carry out all of the tasks listed in the program.

When a program is loaded into memory, it may be divided into the four components stack, heap, text, and data to from a process. The simplified depiction of a process in the main memory is shown in the digram below.

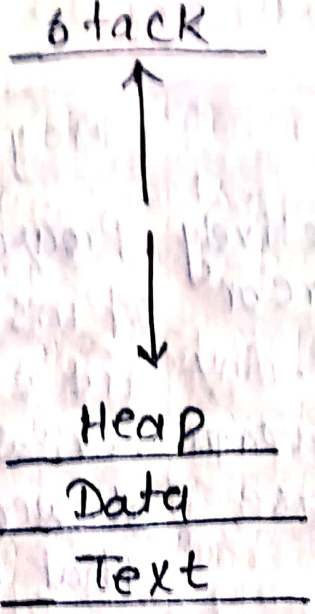


Fig:- Process Diagram

- Stack

The process stack stores temporary information such as method or function arguments, the return address, and local variables.

- Heap

This is the memory where a process is dynamically alloted while it is running.

- Text

This consists of the information stored in the processor's registers as well as the most recent activity indicated by the program counter's value.

- Data

In this section, both global and static variables are discussed.

Program :-

Program is a set of instructions which are executed when the certain task is allowed to complete that certain task. It is usually written in a programming language like, C, C++, Java etc.

Difference between process and program.

S.NO	Process	Program
1.	A process is actively running software or a computer code. Any Procedure must ba carri-ed out in a precise order.	Program is a set of instructions which are executed whe the cer-tain task is allowed to complete that cer-tain task.
2.	Process is Dynamic in nature.	Program is static in nature.
3.	Process is an active in nature.	Program is (static in) Passive in nature.
4.	process is created during the execution and it is loaded direc-tly into the main memory.	Program is already existed in the memory and it is prerent in the secondary memory.
5.	Process has own control system known as Process Control board/Block.	Program does not have any control system
6.	Process changes from time to time by itself.	Program cannot be changed on it's own. It must be changed by the Programmer.
7.	A process needs extra data in addition to the program, data needs for management and execution.	Program is basically divided into two parts one is code part and other is data part.

PROCESS STATE

As a process executes, it changes state. The state of a process is defined in part by current activity of that process. Each process may be in one of the following state.

- New :- The process is being created.
- Running :- Instructions are being executed.
- Waiting :- The process is waiting for some event to occur (such as an I/O completation or reception of a signal)
- Ready :- The process is waiting to be assigned to a processor.
- Terminated :- The process has finished execution

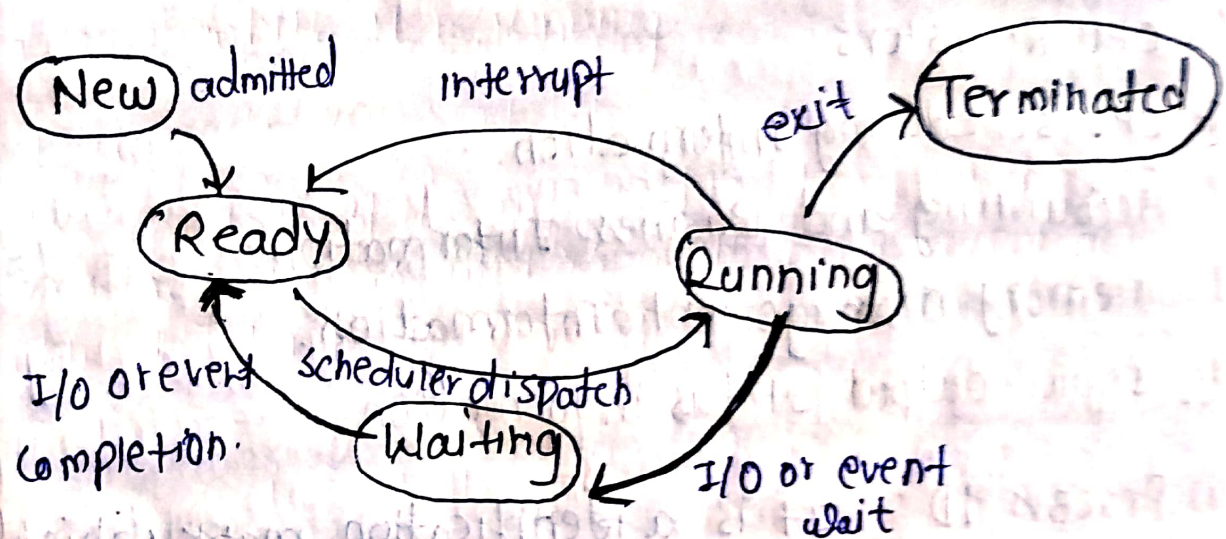


fig :- Process state

PROCESS CONTROL BLOCK (PCB)

An Operating system helps in process creation scheduling and termination with the help of Process Control Block. The process control Block (PCB) which is part of the operating system, aids in managing how processes operate. Every os Process has a process control Block related to it. By keeping data on different things including their state, I/O status, and CPU scheduling, a PCB maintains track of processes.

The Components of process Control Block.

A process Control Block consists of;

1. Process ID
2. Process State
3. Program Counter
4. CPU Registers
5. CPU scheduling Information.
6. Accounting and Business Information.
7. memory management Information.
8. Input output status information.

(i) Process ID :- It is a identification mark which is present for the process. This is very useful for finding the process. It is also very Useful for identifying the process also.

(2) Process ... In a process state many factors work like new state, Ready state, waiting state, Running state, terminated state.

3) Program Counter :- The address of the following instruction to be executed from memory is stored in a CPU register called a program Counter (PC) in the computer processor. It is a digital counter required for both task execution speed and for monitoring the present stage of execution.

An instruction counter, instruction pointer, instruction addresses register, or sequence control register are other names for a program counter.

4) CPU Registers :- When the process is in a running state, here is where the contents of the processor registers are kept. Accumulators, index and general purpose registers, instruction registers and condition code registors are the many categories of CPU registers.

5) CPU Scheduling information :- It is necessary to arrange a procedure for execution. This schedule determines when it transitions from ready to running. Process priority scheduling queue pointers (to indicate the order of execution) and several other scheduling parameters are all included in CPU Scheduling information.

6) Accounting and Business information :- The state of Bussiness addressing and information includes information such as CPU

use the ammount of time a process uses in real time, the number of Jobs or processes etc.

4) Memory Management Information :- The state of the memory Management information section contains information on the page, segment tables, and the value of the base and limit registers ; It relies on the operating system's memory system.

5) Input output status information :- This input output status information section consists of input and output related information which includes about the Process statuses etc.

Operations On the process :-

1. Creation :- Once the process is created, it will be ready and come into the ready queue (main memory) and will be ready for the execution.

2. Execution :-

2. Scheduling :- Out of the many processes present in the ready queue, the operating system chosses one process and start executing it. Selecting the process which is to be executed next ; is know as scheduling.

3 :- Execution :- Once the process is scheduled for the execution, the processor starts executing it. Process may come to the blocked or wait state during the execution then in that case the processor starts executing the other Processes.

4) Deletion/Killing :- Once the purpose of the process gets over then the OS will kill the process. The context of the process (PCB) will be deleted and the process gets terminated by the operating system.

CONTEXT SWITCHING :-

Context switching is a technique or method used by the operating system to switch a process from one state to another to execute its function using CPU in the system. When switching perform in the system, it stores the old running process's status in the form of registers and assings the CPU to a new process to execute its task. While a new process is running in the system, the previous process must wait in a ready queue. The execution of the old process starts at that point where another process stopped it. It defines the characteristics of a multitasking operating system in which multiple processes shared the same CPU to perform multiple tasks without the need for additional processors in the system.

Example of context switching :-

Suppose that Multiple Processes are stored in a Process Control Block (PCB). One process is running state to execute its task with the use of CPUs. As the process is running, another process arrives in the ready queue, which has a high priority of completing its task using CPU. Here we need to use context switching that switches the current process with

the new process requiring the CPU to finish its tasks. While switching the process, a context switch saves the status of the old process in registers. When the process reloads into the CPU, it starts the execution of the process when the new process stops the old process. If we do not save the state of the process, we have to start its execution at the initial level. In this way, context switching helps the operating system to switch between the processes, store or reload the process when it requires executing its task.

Context switching triggers :-

Following are the three types of context switching triggers as follows.

1) Interrupts :- A CPU requests for the data to read from a disk, and if there are any interrupts, the context switching automatic switches a part of the hardware that requires less time to handle the interrupts.

2) Multitasking :- A context switching is the characteristic of multitasking that allows the process to be switched from the CPU so that another process can be run. When switching the process, the old state is saved to resume the process execution at the same point in the system.

3) Kernel/User switch :- It is used in the operating systems when between the user mode, and the kernel/user mode is performed.

4 PROCESS SCHEDULING

Process scheduling is the method by which an operating system manages the execution of processes. It determines which process runs at any given time, aiming to optimize CPU usage and system responsiveness.

Operating system uses various schedulers for the Process scheduling described below.

1. Long term scheduler :- Long term scheduler is also known as job schedule

It chooses the processes from the pool (secondary memory) and keeps them in the ready queue maintained in the primary memory.

Long term scheduler mainly controls the degree of multiprogramming. The purpose of long term scheduler is to choose a perfect mix of IO bound and CPU bound processes among the jobs present in the pool.

2. Short term scheduler :- Short term schedular is also known as CPU scheduler, It selects one of the jobs from the ready queue and dispatch to the CPU for the execution.

A scheduling algorithm is used to select which job is going to be dispatched for the execution. The job of the short term scheduler can be very critical in the sense that if it selects job whose CPU burst time is very high then all the jobs after that, will have to wait in the ready queue for a very long time.

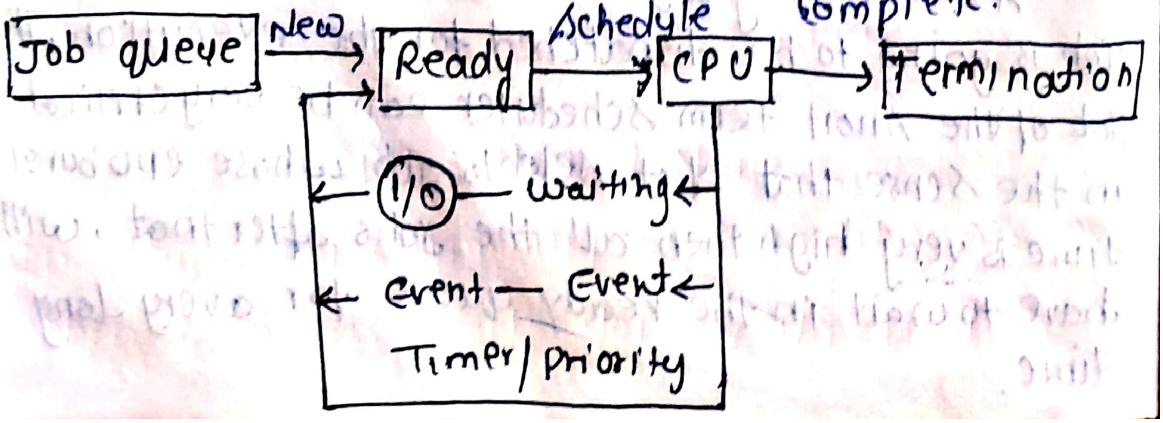
The problem is called starvation which may arise if the short term scheduler makes some mistakes while selecting the job.

3) Medium term scheduler :- Medium term scheduler takes care of the swapped out processes. If the running state processes needs some IO time for the completion then there is a need to change is state from running and to waiting.

Medium term scheduler is used for this purpose. It removes the process from the running state to make room for the other processes. Such processes are the swapped out processes and this procedure is called swapping. The medium term scheduler is responsible for suspending and resuming the process.

PROCESS QUEUES :-

The operating system manages various types of queues for each of the process states. The PCB related to the process is also known stored in the queue of the same state. If the process is moved from the corresponding queue and added to the other state queue in which the transition is made.



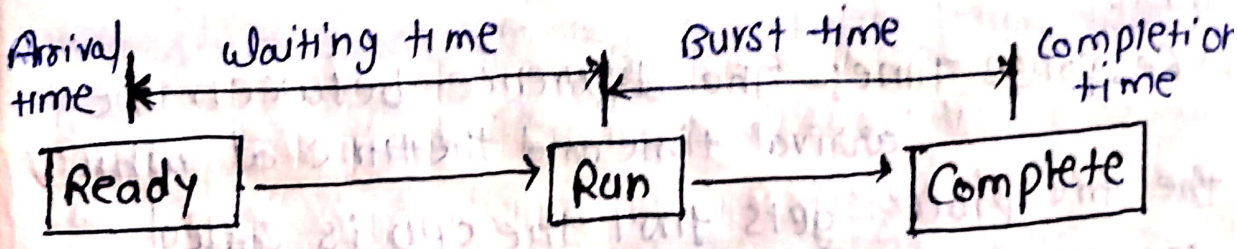
There are the following queues maintained by the Operating system.

1. Job Queue :- In starting, all the process get stored in the job queue. It is maintained in the secondary memory. The long term scheduler (Job scheduler) picks some of the jobs and put them in the primary memory.

2. Ready Queue :- Ready queue is maintained in primary memory. The short term shedul picks the job from the ready queue and dispatch to the CPU for the execution.

3. Waiting Queue :- When the process needs some IO operation in order to complete its execution, OS changes the state of the process from running to waiting. The context (PCB) associated with the process gets stored on the waiting queue which will be used by the process-or when the process finishes the IO.

Various times related to the process:



CT - AT = WT + BT

TAT = CT - AT

Waiting time = TAT - BT

where

TAT → Turn around time

BT → Burst time

AT → Arrival time.

1) Arrival time :- The time at which the process enters into the ready queue is called the arrival time.

2) Burst Time :- The total amount of times required by the CPU to execute the whole process is called the Burst time. This does not include the waiting time.

3) Completion Time :- The time at which the process enters into the completion state or the time at which the process complete its execution, is called completion time.

4) Waiting time :- The total amount of time for which the process waits for the CPU to be assigned is called waiting time.

5) Turnaround Time :- The total amount of time spent by the process from its arrival to its completion, is called turnaround time.

6) Response time :- The difference between the arrival time and the time at which the the process gets first the CPU is called Response Time.

CPU SCHEDULING

In the Uniprogramming systems like ms Dos, when a process waits for any I/O operation

to be done. the CPU remains idol. This is an overhead since it wastes the time and causes the problem of starvation. However in Multiprogramming systems, the CPU doesn't remain idle during the waiting time of the process and it starts executing the processes. Operating system has to define which process the CPU will be given.

In multiprogramming systems, the operating systems scheduling the process on the CPU to have the maximum utilization of it and this procedure is called CPU scheduling. The operating system uses various scheduling algorithm to schedule the processes.

Why do we need scheduling :-

In Multiprogramming, if the long term scheduler picks more I/O bound processes than most of the time the CPU remains idol. The task of operating system is to optimize the utilization of resources.

If the most of the running processes change their state from running to waiting then there may always be a possibility of deadlock in the system. Hence to reduce this overhead, the OS needs to schedule the jobs to get the optimal utilization of CPU and to avoid the possibility to deadlock.

" CPU scheduling is a process which allows one process to use the CPU while the execution of another process is on hold (waiting state) due

to unavailability of any resource like I/O etc. Thereby making full use of CPU. The aim of scheduling is to make the system efficient fast and fair.

Non-Preemptive Scheduling :-

Under non-preemtive scheduling, once the CPU has been allocated to a process the process keeps the CPU untill it releases the CPU either by terminating or by switching to the waiting state. This scheduling method is used by the microsoft Window 3.1.

Preemtive scheduling :-

In this type of scheduling the tasks are usually assigned with priorities. At times it is necessary to run a certain task that has a higher priority before another task although it is running. Therefore, the running task is interrupted for some time and resumed later when the priority task has finished its execution.

Scheduling Algorithms in OS :-

There are various algorithms which are used by the operating system to schedule the processes on the processor in an efficient way.

The purpose of a scheduling algorithm :-

1. Maximum CPU Utilization.
2. Fare allocation of CPU.

3. Maximum throughput.
4. Minimum turnaround time.
5. Minimum waiting time.
6. Minimum response time.

There are the following algorithms which can be used to schedules the jobs.

1) First come first serve:- It is the simplest algorithm to implement. The process with the minimal arrival time will get the CPU first. The lesser the arrival time, the sooner will the process gets the CPU. It is the non-preemptive type of scheduling.

(2) Round Robin :- In the Round Robin scheduling algorithm, the OS defines a time quantum (slice). All the processes will get executed in the cyclic way. Each of the process will get (back to the ready queue) the CPU for a small amount of time (time quantum) and then get back to the ready queue to wait for it's next turn. It is a preemptive type of scheduling.

r3) Shortest Job first :- The job with the shortest Burst time will get the CPU first. The lesser the burst time, the sooner will the process get the CPU. It is the non-preemptive type of scheduling.

(4) Shortest remaining time first :- It is the preemptive form of SJF. In this algorithm the OS schedules the job according to the remaining time of execution.

(5.) Priority based Scheduling :-

In this algorithm the priority will be assigned to each of the processes. The higher priority, the sooner will the process get the CPU. If the Priority of the two processes is same then they will be scheduled according to their arrival time.

(6) Highest Response Ratio Next :-

In this scheduling algorithm, the process with highest response ratio will be scheduled next. This reduces the starvation in the system.

FIRST COME FIRST SERVE CPU PROCESS SCHEDULING IN OPERATING SYSTEM.

First come first serve paves the way for understanding of other algorithms. This algorithm may have many disadvantages. But these disadvantages created very new and efficient algorithms! So, it is our responsibility to learn about First come first serve CPU Process scheduling Algorithms.

Important Abbreviations

1. CPU - Central processing Unit
2. FCFS - First come first serve
3. AT - Arrival Time
4. BT - Burst Time
5. WT - Waiting Time
6. TAT - Turn Around Time.
7. CT - Completion Time
8. FIFO - First come first out.

First come first serve:-

First come first serve CPU scheduling Algorithm shortly known as FCFS is the first algorithm of CPU Process Scheduling Algorithm. In First come first serve Algorithm In what we do is to allow the Process to execute in linear manner.

This means that whichever process enters process enters the ready queue first is executed first. This shows that first come first serve Algorithm follows first in first out (FIFO) Principle.

The first come first serve Algorithm can be executed in pre emptive and non pre Emptive manner.

- Characteristics of FCFS CPU process Scheduling

1. Implementation is simple.
2. Does not cause any causalities while using.
3. It adopts a non-preemptive and pre emptive strategy.
4. Arrival time is used as a selection criterion for Procedures.

- Advantages :-

- In allocate order to allocate processes, it uses the first come first out queue.
- FCFS CPU scheduling process is straight forward and easy to implement.
- There is no chance of process starving.
- As there is no consideration of process priority, It is an equitable algorithm.

- Disadvantages :-
- FCFS CPU scheduling Algorithm has long waiting time.
- FCFS CPU scheduling favors CPU over Input or output operations.
- In FCFS there is a chance of occurrence of convoy Effect.

5. Interprocess communication And synchronization.

In General Inter process Communication is a type of mechanism usually provided by the operating system (or OS). The main aim or goal of this mechanism is to provide communications in between several processes. In short, the intercommunication allows a process letting another process know that some event has occurred.

Definition:- "Interprocess Communication is used for exchanging useful information between numerous threads in one or more processes (or programs)

Types of Processes:-

- Independent process.
- co-operating process.
- An indipendent process is not affected by the execution of other processes.
- while a co-operating process can be affected by other executing processes.

methods of IPC

(i) Shared memory

(ii) message passing

(ii) Shared memory method

There are two processes : Producer and Consumer The producer produces some items and consumer consumes that item. The two processes share a common space or memory location known as buffer where the item produced by the producer is stored and from which the consumer consumes the item if needed. There are two versions of this problem

- The first one is known as unbounded buffer problem in which producer can keep producing items and there is no limit on the size of the buffer, the second one is know as,

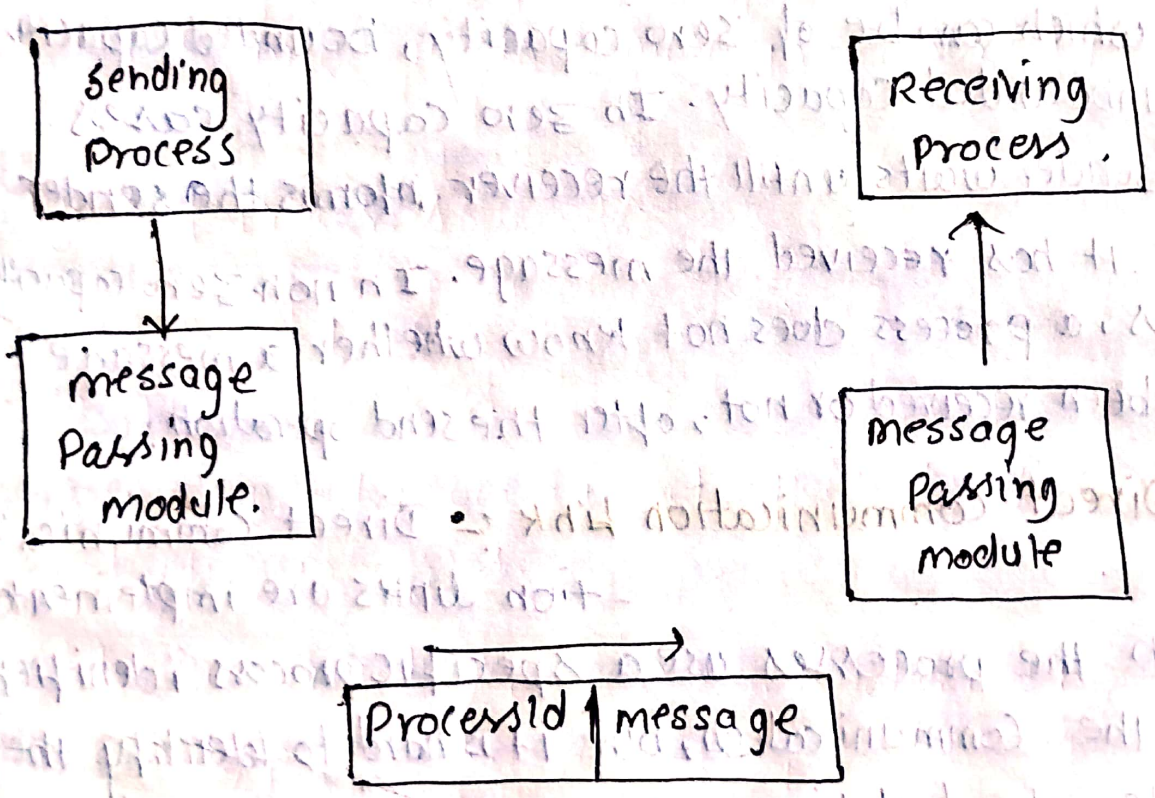
Bounded buffer problem. In which the producer can produce upto a certain number of items before it starts waiting for consumer to consume it. ~~wh~~

first the producer and the consumer will share a common memory location, then the producer will start producing items. If the total produced item is equal to the size of the buffer, the producer will wait to get it consumed by the consumer. Similarly, the consumer will first check for the availability of the item. If no item is available, the consumer will wait for the producer to produce it. If there are items available, consumer will consume them.

(iii) message passing method:

In this method, processes communicate with each other without using any kind of shared memory. If two processes P_1 and P_2 want to communicate with each other, they proceed as follows.

- establish a communication link (If a link already exists, no need to establish it again)
- start exchanging messages using basic primitives.
  We need at least two primitives:
  - send (message, destination) or send (message)
  - Receive (message, host) or receive (message)



The message size can be fixed size or of variable size. If it is of fixed size, it is easy for an os designe but complicated for a programmer and if it's of variable size then it's easy for a programmer but complicated for the os designer.

A standard message can have two parts : header and body. The header part is used for storing message type, destination id, source id, message length, and control information, The control information contains information like what to do if runs out buffer space, sequence number, priority, Generally, message is sent in FIFO style.

message passing through communication link.

A link has some capacity that determines the number of messages that can reside in it temporarily for which every link has a queue associated with it which can be of zero capacity, bounded capacity or unbounded capacity. In zero capacity cases, the sender waits until the receiver informs the sender that it has received the message. In non-zero capacity cases, a process does not know whether a message has been received or not, after the send operation.

(i) Direct Communication Link :- Direct communication links are implemented when the processes use a specific process identifier for the communication, but it is hard to identify the sender ahead of time. ex: print server.

(ii) Indirect Communication Link :- In-direct communication is done via shared mailbox (port), which consists of a queue of messages. The sender keeps the message in mail box and receiver picks them up.

Role of Synchronization in IPC.

It is one of the essential parts of their process Communication. Typically, this is provided by interprocess Communication control machanisms, but sometimes it can also be controlled by Communication process.

These are the following methods that used to provide the synchronization.

(i) mutual execution :- It is generally required that only one process thread can enter the critical section at a time. This also helps in synchronization and creates stable state to avoid the race condition.

(ii) Semaphore :- semaphore is a type of variable that usually controls the access to the shared resources by several processes. Semaphore is further divided into two types which are as follows.

(i) Binary semaphore.

(ii) Counting semaphore.

(iii) Barrier :- A barrier typically not allows an individual process to proceed unless all the processes does not reach it. It is used by many Parallel languages, and Collective routines impose barriers

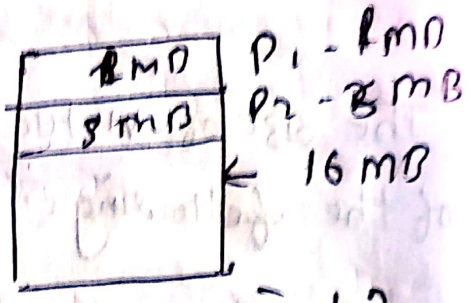
(iv) Spinlock :- Spinlock is the type as its name implies.

The processes are trying to acquire the spinlock waits or stays in a loop while checking that the lock is available or not. It is known as busy waiting because even though the preess active, the process

Memory management

* Memory management is a form of resource management applied to computer memory. The essential requirement of memory management is to provide the ways to dynamically portions of memory to programs at their request, and free it for reuse when no longer needed.

→ sweap in and swap out concept:-



* why memory management is required?

→ Allocate and deallocate memory space before and after the process execution.

→ To keep track of used memory space by Processes.

→ To minimize fragmentation issue.

→ To proper utilization of main memory.

→ To maintain data integrity while executing of process.

memory shouldn't be useless

Logical And Physical Address..

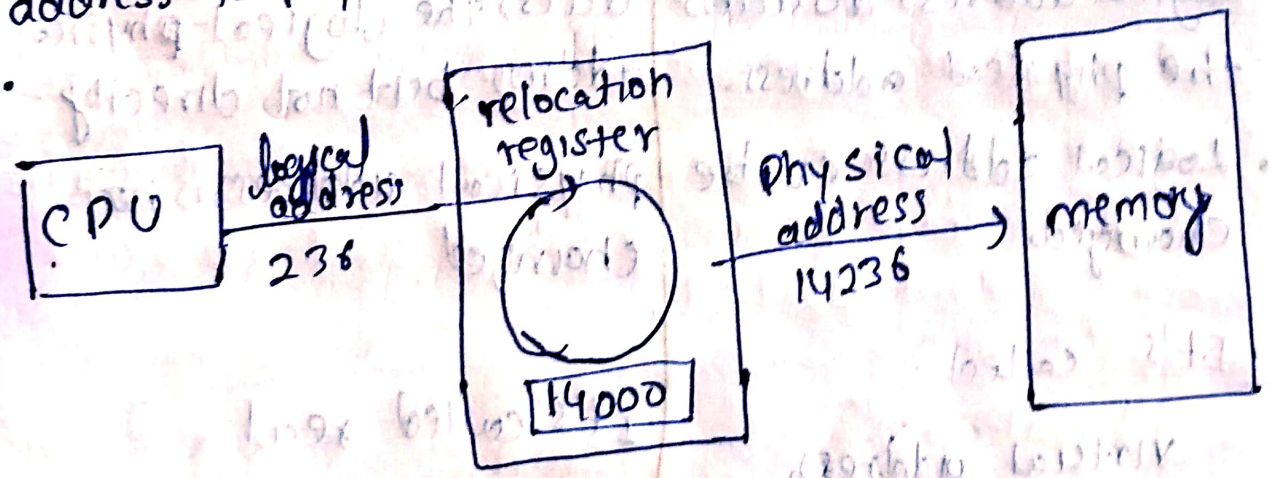
Logical Address:- logical address is known as virtual address.

- It's a address generated by CPU during program execution.
- This address is used as a reference to access the physical memory located by CPU.

Physical Address:-

It is the actual address in main memory where data is stored.

- Physical address are used by the memory management unit (mmu) to translate logical address to physical address.

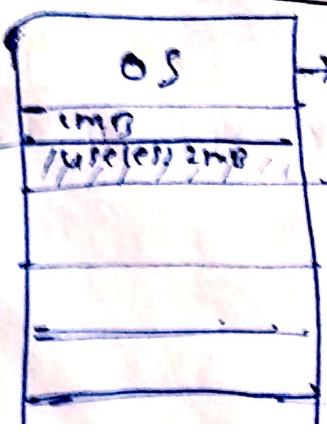


Explaination —?

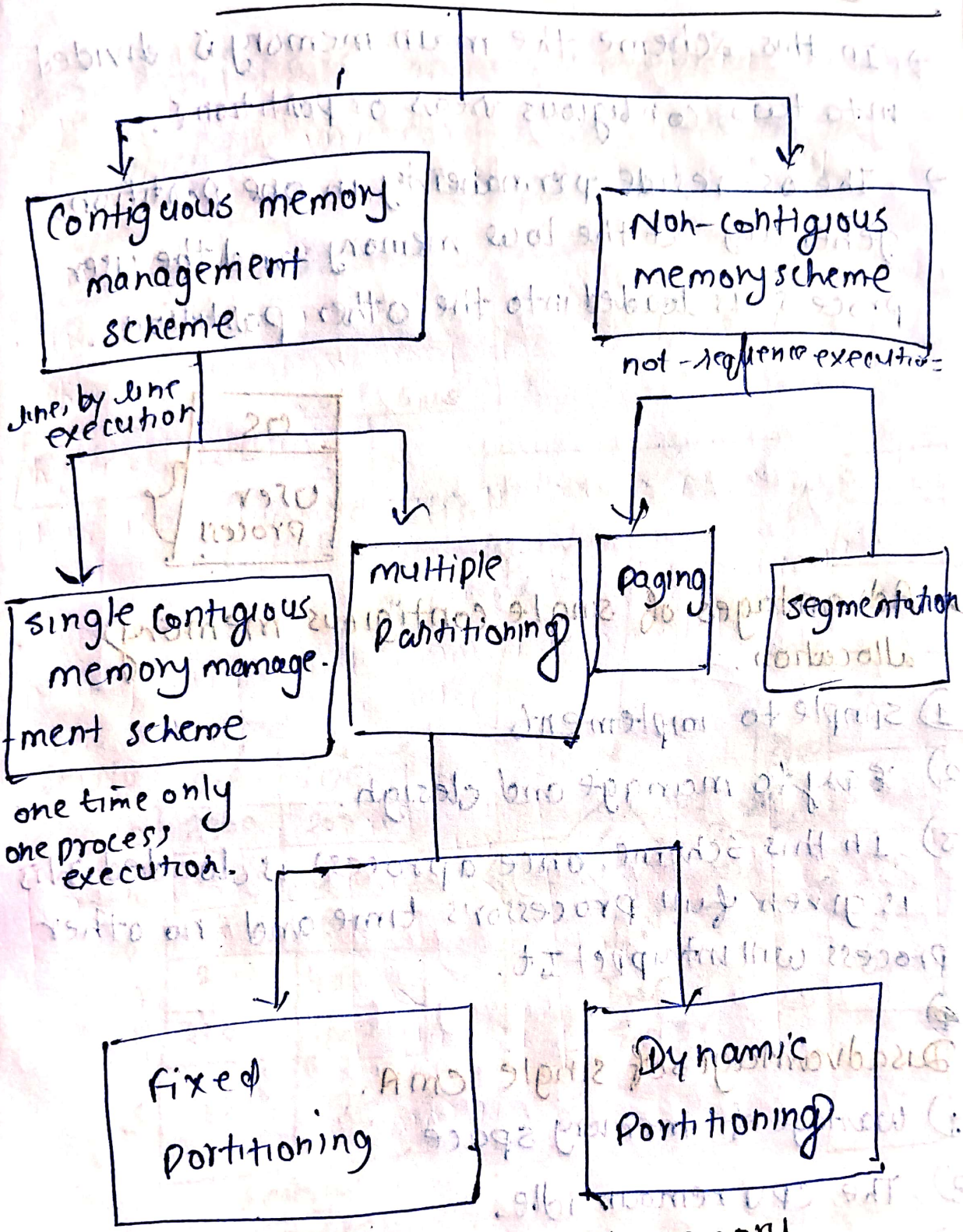
Physical address V/S logical address

Logical add.	Physical Add.
• It's generated by CPU	• It's acces the physical memory location by CPU.
• It is set of all logical address generated by CPU in reference of a program.	• Address mapped to the corresponding logical add-ress.
• User can view the logical address of a Program.	• User can never view physical address of a Program.
• User can use the logical address to access the physical address.	• The user can indirectly access the physical addres but not directly
• Logical address can be changed.	physical address is not changed.
• It's called virtual address.	It's called real address.

Ex: P_1, P_2, P_3, P_4
1MB, 2MB, 3MB, 2MB



Memory management techniques:-



(i) Contigious memory allocation

→ In a contigious memory allocation management scheme, each program occupies a single block of storage location such as a set of memory locations with consecutive address.

(1) single contigious memory allocation.

→ In this scheme, the main memory is divided into two contiguous areas or partitions.

→ The os reside permanently in one partition generally at the low memory and the user process is loaded into the other partition.

- Advantages of single contigious memory allocation.

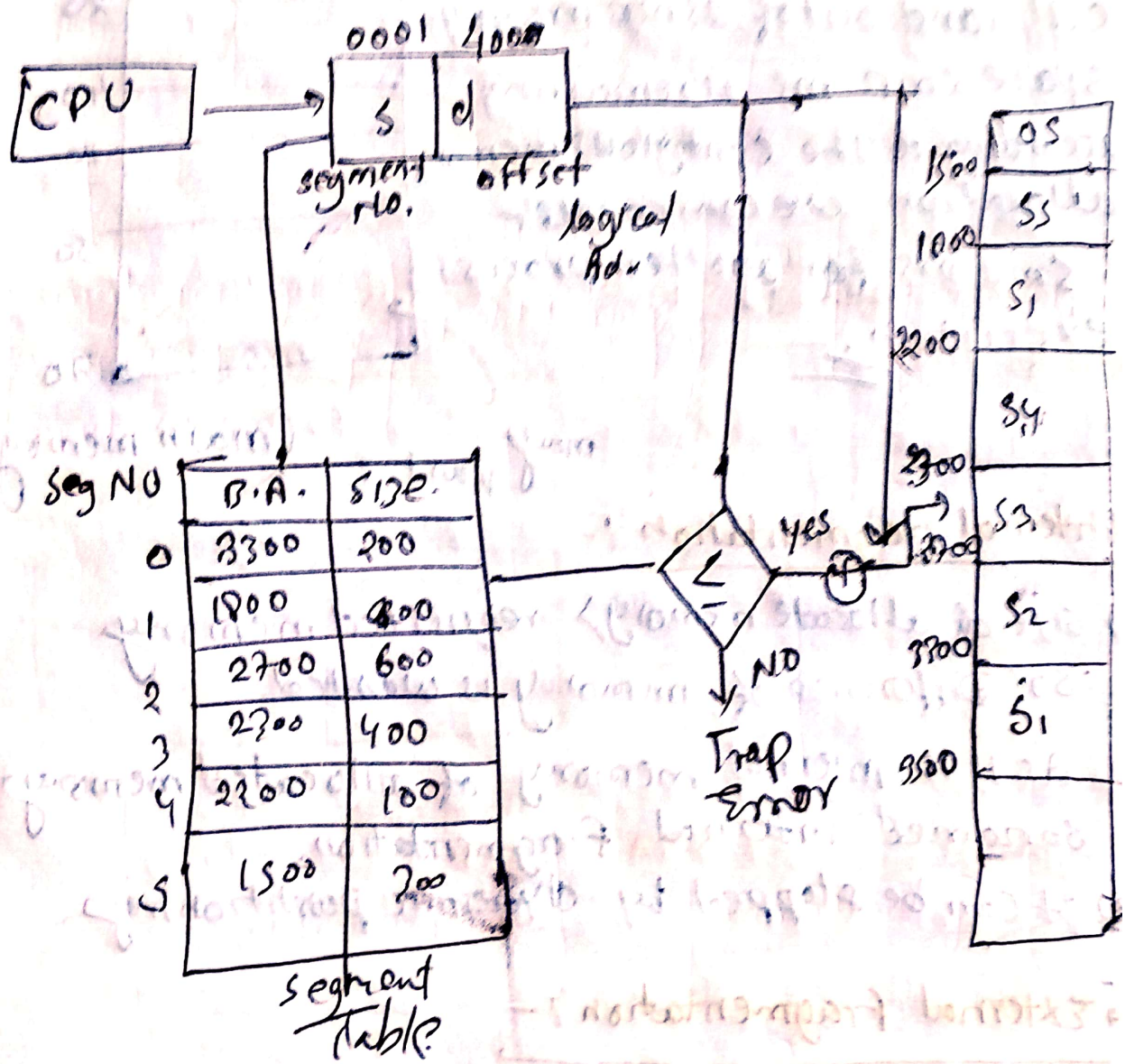
1) simple to implement.
2) easy to manage and design.
3) In this scheme, once a process is loaded, it is given full processor's time and no other process will interrupt it.

4) Disadvantages of single CMA.

1) Wastage of memory space.
2) The CPU remain idle.
3) It can not be executed if the program is too large.
4) It does not support multiprogramming.

Segmentation:

Segmentation is a memory management technique in which the memory is divided into the variable size parts. Each part is known as a segment which can be allocated to a process.

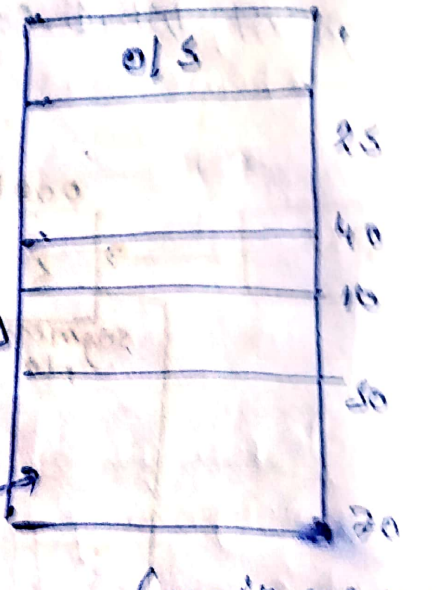


B.A = Base Add.

Fragmentation :-

It's is wastage or loss of memory due to may be of free memory is not being executed or not able to allocate it.

if we have a process of 20MB we can fill/execute it in first cell, and out of 25MB memory space 5MB ms is remaining according to the contyious memory allocation we can not use 5MB ms for another process execution.



Internal fragmentation :-

→ size of allocate memory > requested memory
: so Difference of memory is wasted.

→ As it is internal memory of allocated memory so named internal fragmentation.

→ It can be stopped by dynamic partitioning.

External fragmentation :-

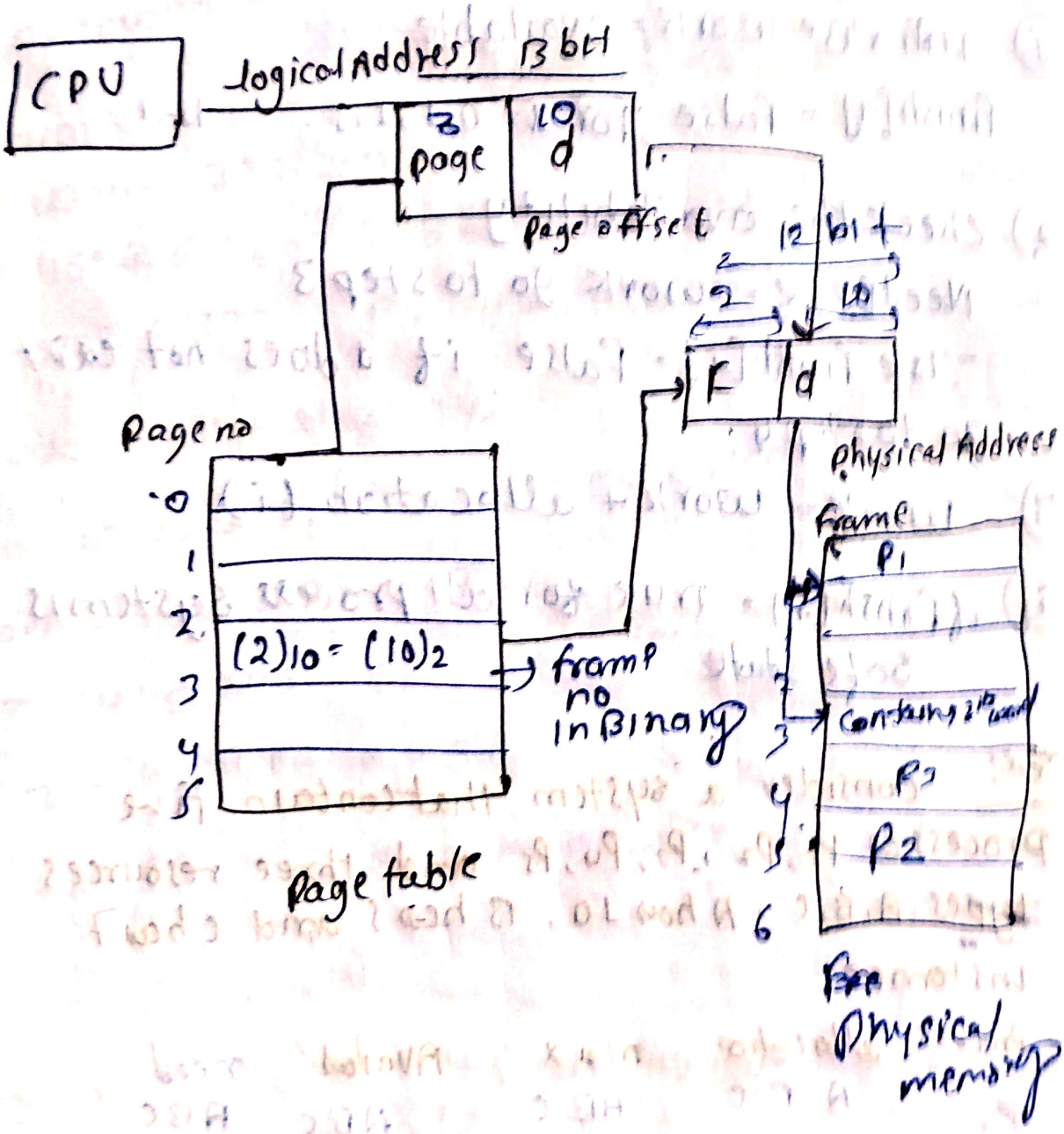
→ Total memory space exists to satisfy a request but it is contyious

→ As it is unallocated memory part so named External memory location.

→ It can be stopped by Paging & segmentation.

Paging

Paging is the storage machanism used in OS to retrieve processes from secondary storage to the main memory or pages.



Deadlock Avoidance

Banker's Algorithm:-

Bankers Algo is the combination of safety Algorithm and the resource request Algorithm

Safety Algo:-

1) Initialize work = available
   finish[i] = false ; for i = 0, 1, 2, 3 ---- n-1
2) check the availability
   Need[i] <= work go to step 3
   else finish[i] = false if it does not exist go to step 4.
3) work = work + allocation (i)
4) if finish[i] = true for all process systems is safe state.

Ex: Consider a system that contain five processes P_1, P_2, P_3, P_4, P_5 and three resources types A, B, C, A has 10, B has 5 and C has 7 instances:

Process	allocation	max	Availabl	need
	A B C	A B C	A B C	A B C
P_0	0 1 0	7 5 3	3 3 2	7 4 3
P_1				1 2 2
P_2	2 0 0	3 2 2		6 0 0
P_3	3 0 2	9 0 2		
P_4	2 1 1	2 1 1		0 1 1
P_5	0 0 2	4 3 3		4 3 1

need = max - allocation

1) Available = work

Need of P1 ≤ work

743 ≤ 332 × P_1 is not executed

122 ≤ 332

P_2 executed.

2) work = work + allocation of P_2

work = 332 + 200

~~532~~ 600 ≤ 532 × P_3 is not executed.

000 ≤ 532 P_4 is executed

work = 532 + 211 = 743

743 ≤ 431 P_5 is executed.

work = 743 + 002 = 745

743 ≤ 745 P_1 is executed

work = 745 + 010 = 750

600 ≤ 750 P_3 is executed.