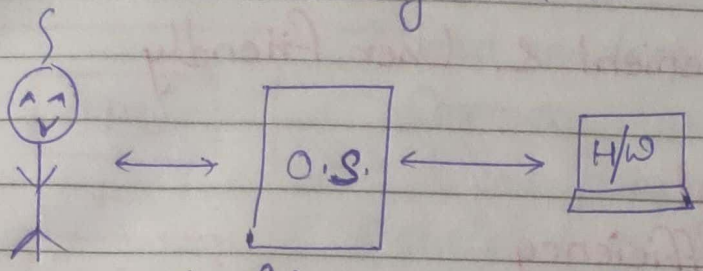
Operating System

It is a system software.

→ Acts as intermediatory between h/w & user



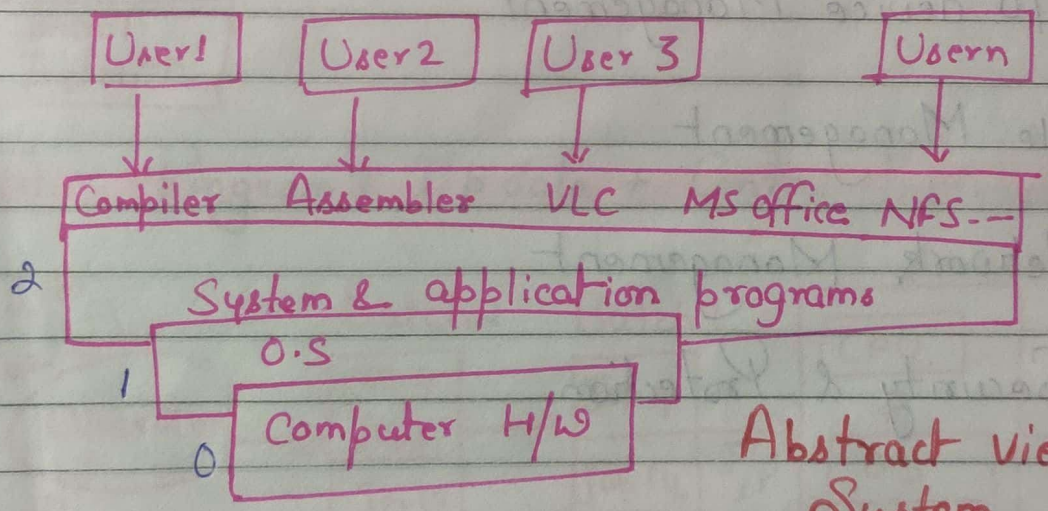
Direct access to h/w is not efficient & convenient.

→ Resource Manager

Which application is provided which resource

→ Provide a platform on which other application programs are installed

Single window operation — All applications request O.S. & can't access hardware directly.



Abstract view of System

* System Software —: That Works directly in system management

Ex —: Compiler, ~~Disk~~ Device driver, Assembler etc.

* Application Software —: MS-office, VLC, Game etc.

* Goals of O.S. :-

→ Primary Goal
⇒ Convenient & User Friendly

→ Secondary Goal
⇒ Efficiency

* functions of O.S. :-

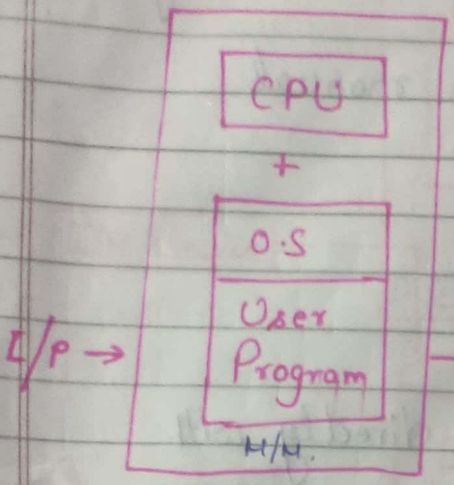
1. Process Management
2. Memory Management
3. I/O device Management
4. File Management
5. Network Management
6. Security & Protection

Evolution of O.S.

APPU DATE ------ PAGE ------

(1) Batch Processing

In starting Mainframe Computer

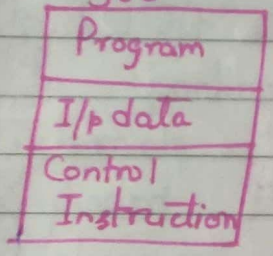


→ Card readers & tape drivers were Common I/p & o/p device

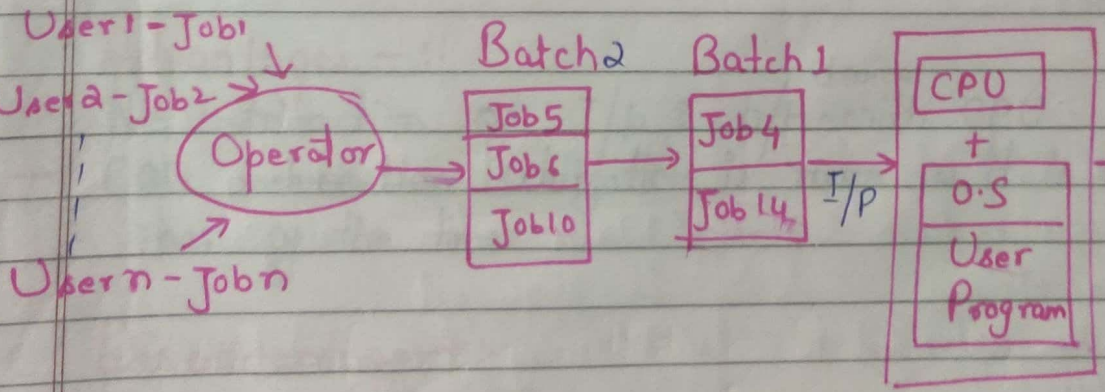
→ User prepared a Job which consisted of the program I/p data & control instruction

⟶ I/p Job was given in the form of punch cards & Results appeared in the form of punch cards. After processing

⟶ OS was simple, always present in memory & major task was to transfer one Job to other.



Job = Program + I/p data + Control Instruction.



— Jobs with similar needs are batched & executed through the processor as a group.

— Operator sorts Jobs as a deck of punch cards into batch with similar needs.

Eg FORTRAN, COBOL

APPU DATE
PAGE

Advantage -:

- In a batch Job execute one after other saving time from activities like loading compiler.
- During a batch execution no manual intervention is needed

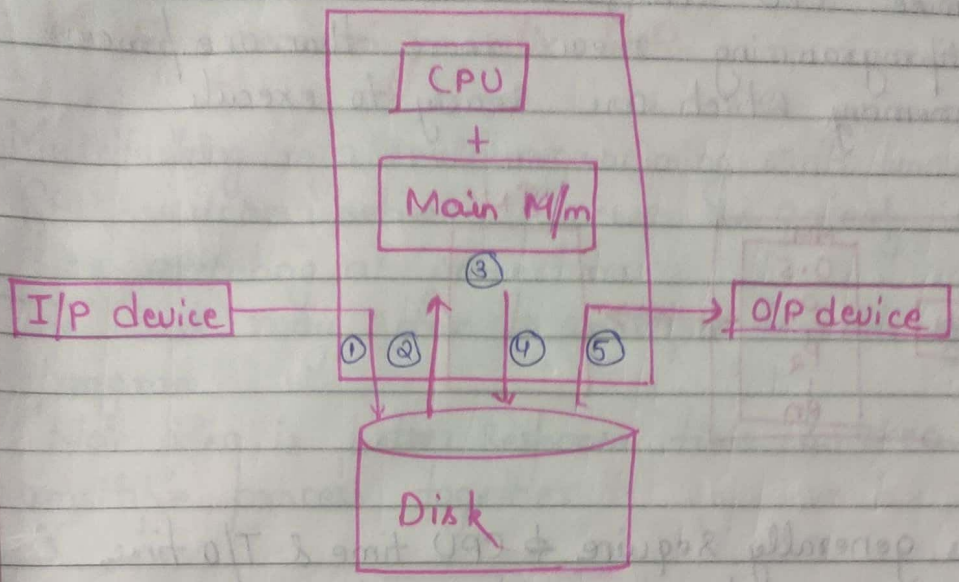
Disadvantage -:

- M/m limitation
- Interaction of I/p & o/p devices directly with CPU.

(2)

APPU DATE _____
PAGE _____

* SPOOLING ⇒ :· Simultaneous peripheral operation online



— I/p & O/p devices are slow than Computer
— In spooling, data is stored into the disk & than interact with disk (digital) via main m/m.
— Keyboard, mouse etc
— Spooling is capable of overlapping I/O operation for one job with CPU operation of other jobs.

Advantages – :·

— No interaction of I/p & o/p device with CPU
— CPU ~~interaction~~ utilization is more as it is busy most of the time

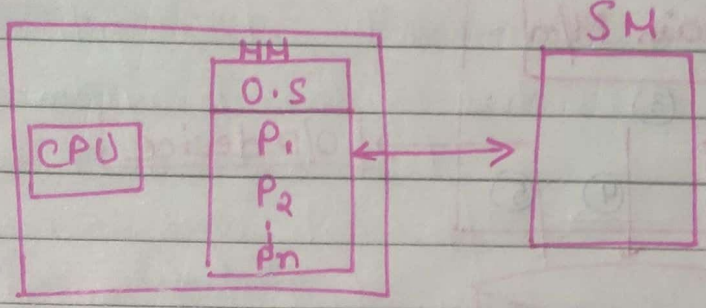
Disadvantage – :·

— Spooling was uniprogramming.

(3)

Multiprogramming O.S. :-

- Maximize CPU utilization
- Multiprogramming mean more than one process in memory which are ready to execute



- Process generally require both CPU time & I/O time, so if running process perform I/O or some other event which do not require CPU then instead of sitting idle CPU make a context switch & pick some other process & this idea will continue.
- CPU is never idle unless there is not process ready to execute or at the time of context switch.

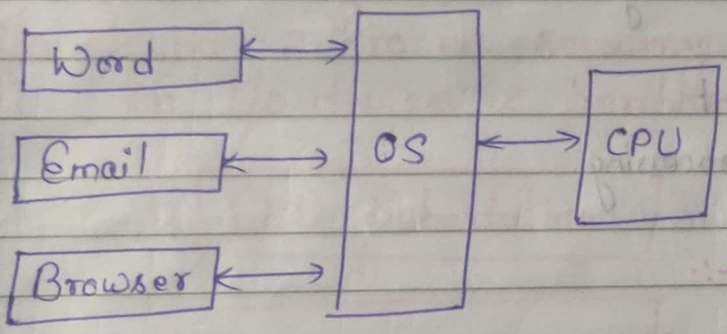
Advantage	Disadvantage
✓ High CPU utilization	✓ Difficult scheduling
✓ Less waiting time, response time etc	✓ Main m/m management is required
✓ Maybe extended to multiple users	✓ Memory fragmentation
✓ Nowadays useful when load is more.	✓ Paging (Non-contiguous m/m allocation).

(4)

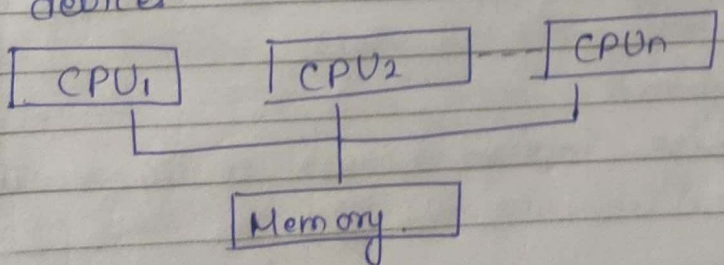
APPU DATE _____
PAGE _____

Multitasking / Time Sharing / Fair Share / Multiprogramming with R.R.

- Multitasking is a multiprogramming with time sharing.
- CPU switches between process so quickly it seems like executing at same time
- The task may refer to multiple threads of the same program
- Main idea is better response time & executing multiple process together



(5) Multiprocessing O.S.

- 2 or more CPU within a single computer in close communication sharing the system bus, m/m & other I/O devices
- 

- Different process may run on different CPU, true parallel execution.

- Symmetric ⇒ One O.S. control all CPU, each CPU has equal rights
- Asymmetric ⇒ Master slave architecture, system task on one processor & applications on other as one CPU will handle all h/w interrupt or I/O device, they are easy to design but less efficient.

Advantage –:

- ✓ Increased throughput
- ✓ Increased reliability
- ✓ Cost saving
- ✓ Battery efficient
- ✓ True parallel processing.

Disadvantage –:

- ✓ More Complex
- ✓ Overhead or coupling reduce throughput
- ✓ Large main m/m.

UNIT - I

APPU DATE _____ PAGE _____

Q) O.S. →

It is a program that after being initially loaded into computer manages all other programs. It is the management of activities & sharing limited resources of the computer.

Q) Different views of O.S.

(i) User View

- Single user optimization
  Resource utilization is taken lesser care than that of the performance of system
- A user access system through other terminals.
- User sharing resources such as networking & servers to access & print file.
- Handheld systems.
- Embedded Systems

(ii) System View

- O.S. as resource manager
- O.S. as I/O manager.

Q) Real time System

When rigid time requirements have been placed on the operation of a processor or flow of data

Eg :- Systems controlling Scientific experiments
Medical image system.

Types -:

(i) Hard - Real time system

- Action must occur at certain moment.
- No Secondary storage — Conflicts with time sharing
- Data stored in ROM →

Eg :- Car engine
Heart Pacemaker

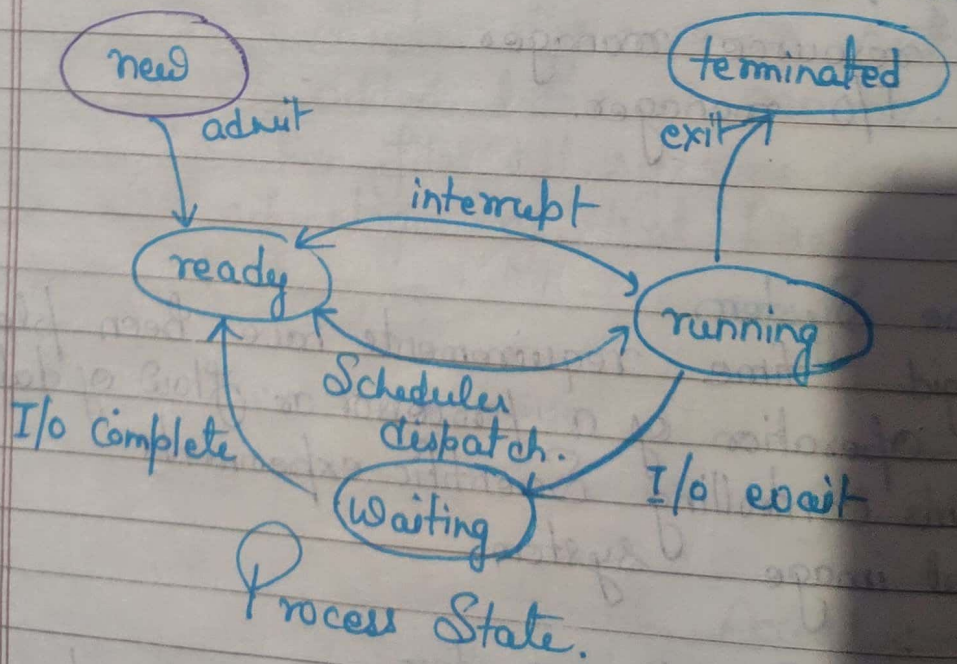
(ii) Soft Real time system.
Limited utility in industrial control
Useful in apps like multimedia,

Eg - Live audio - video system

Unit II

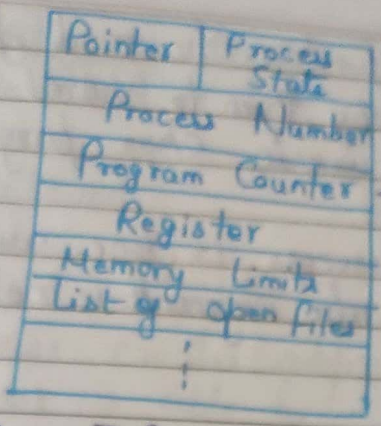
Process.

A program in execution. It include a process stack, having temporary data & data section which contain global variable.



Process State.

* PCB -:



- Address of next instruction

- CPU-scheduling Information.
- M/m management info
- Accounting info
- I/O status info.

2) Concurrency —:
A process is cooperating if it gets effected or can effect execution. It is done by IPC types-:
Shared m/m or Msg Passing.
2 or more processes should be synchronized that they run concurrently & should n't have deadlock.
Synchronization is needed when one process is waiting for another to complete.
Eg -: Read/Write Method.

3) IPC —: It is not generally supported by single user system. (DOS).

Shared Memory —: Sharing of address space.

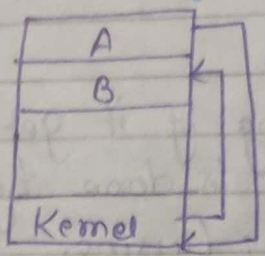
1) Initialize shared m/m area. (Generate identifier)
2) Attach shared m/m to some address.

(iii) We invoke a function to read from keyboard & copy them to shared m/m. In case of α numeric char program terminate.

(iv) Detaching of process is done automatically but the shared m/m segment is not destroyed.

Ex-: Producer Consumer problem.
— Using Buffer.

(ii) Message Passing.



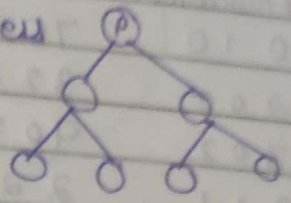
(a) Direct or Indirect Communication
Direct —: Each process
send (p, message)
receive (q, message)

Indirect — Mailbox.
send (a, msg)
Sorry Receive (a, msg).

(b) Synchronous & Asynchronous
↓ ↓
[Blocking → Send & receive
[Unblocking → Send & receive.

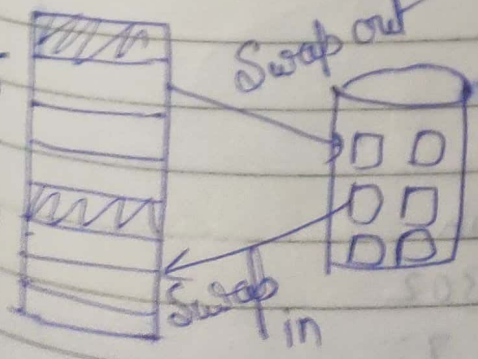
(c) Automatic & Explicit Buffering
— Zero capacity buffer
— Bound "
— Unbound "

(d) Process Generation.
When a process creates subprocess that subprocess is able to obtain its resources directly from O.S.



Resources are shared by process to its children. Restricting a child ~~from~~ to a subset of a parent prevent any process from creating too many subprocesses & overloading.
When a process is created it obtains various physical & logical resources, input that may be passed from parent to child
- Parent executes concurrently with children.
- Parent waits for child execution.
- Child is duplicate of parent (address space).
- Child process has a program loaded into it.

(e) Demand Paging
Page is brought into m/m when it is demanded. Combination of paging & swapping. Complete process should be in sec. storage as pages.



A-:① Reduce m/m req.
② Swap time reduces
③ Degree of multiprog. increase

D-:① Page fault may occur
② M/m is complex.

① 5 processes (P_0, P_1, P_2, P_3, P_4)

3 Resources

A = 10 B = 5 C = 7.

P.	Allocation	Max	Available / ~~Needed~~	Need
P_0	0 1 0	7 5 3	3 3 2	7 4 3
P_1	2 0 0	3 2 2		1 2 2
P_2	3 0 2	9 0 2		6 0 0
P_3	2 1 1	2 2 2		0 1 1
P_4	0 0 2	4 3 3		4 3 1

int

Need = Max − Allocation.

Work = 332.

P_0 Need = 743 —— No.

P_1 Need = 122 —— Yes. finish

Work = 332 + 200

= 532

P_2 Need = 600 —— No

P_3 Need = 011. finish.

Work = 532 + 211

= 743

P_4 Need = 431. finish

Work = 743 + 002

= 745.

P_0 Need = 743 finish.

Work = 745 + 010

= 755

P_2 Need = 600 finish

Work = 755 + 302

= 1057

Safe Sequence

$P_1 \rightarrow P_3 \rightarrow P_4 \rightarrow P_0 \rightarrow P_2$

APPU DATE _____ PAGE _____

∴ If P_1 request $(1,0,2)$ (A B C) check if it is a safe state.

$P_1 \longrightarrow R(1,0,2)$

Need $P_1 \rightarrow 122$.

Available = 332 − 102

= 230..

P_1 = 200 + 102

= 302.

Need P_1 = 122 − 102

= 020

Need	Available
743 — x	230.
020 — ✓	230 + 200 = 430.
600 — x	
011 — ✓	430 + 211 = 641.
431 — ✓	641 + 002 = 643.
P_0 — 743 x	
P_3 = 600 ✓	643 + 302 = 945
P_0 - 743 ✓	945 + 010 = 955. ✓✓