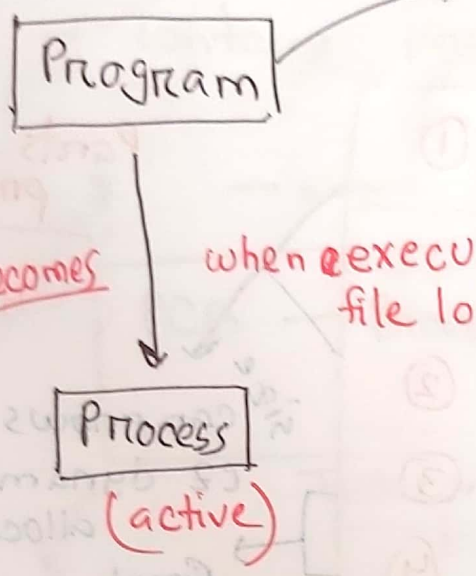
Chapter-3

Process



(when program starts execution in RAM it becomes process)

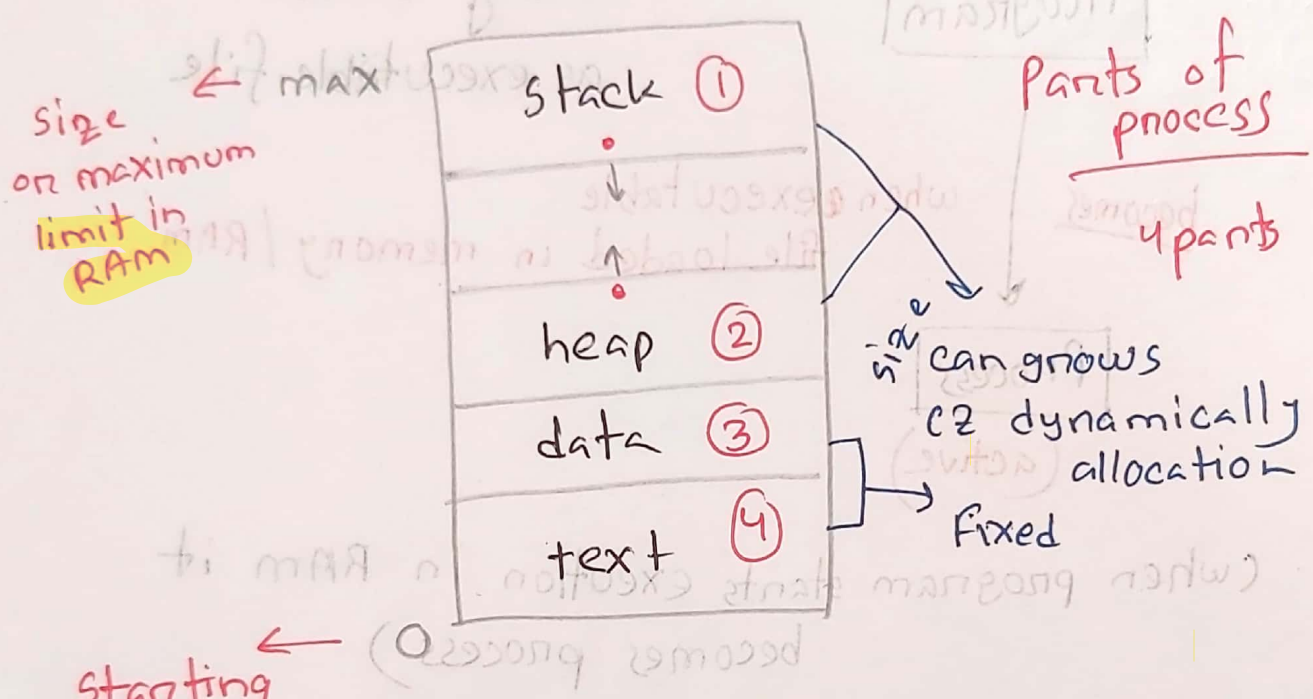
* To start a program you must bring it to ram

* program consists some instruction

* one program can have several process

Process

→ Process is a program in execution



(How does process look likes in memory / RAM)

① stack → ~~contains program code~~

→ contains tempory data

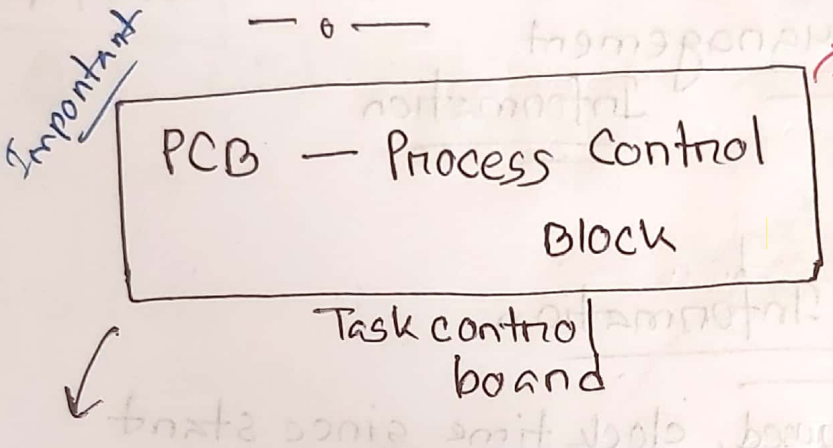
→ function parametens,
return addresses
Local variables

↓
inside in function

② Data Section → contain global vanibles

③ Heap → contain memory dynamically allocated during run time

④ Text → contains program code



information associated with each process in the OS

① Process state :

i — (new) ii — (running) iii — (waiting)

iv - (ready) v - (terminated)

② Program counter : Location of instruction to next execute (where the stants of execution)

(3) CPU Registers :- content of all process centric registers

(4) CPU scheduling Information

Priorities, scheduling pointers

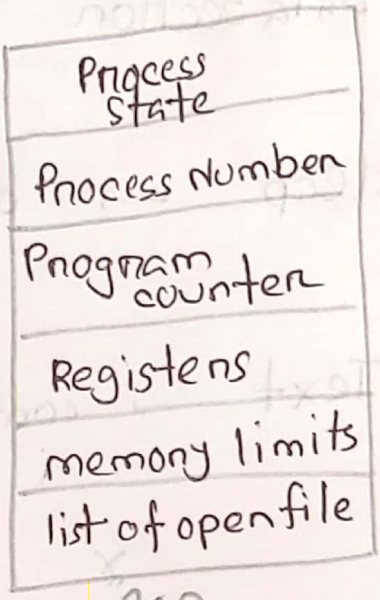
(5) Memory management Information

(6) Accounting Information

cpu used, clock time since start

(7) I/O status Information

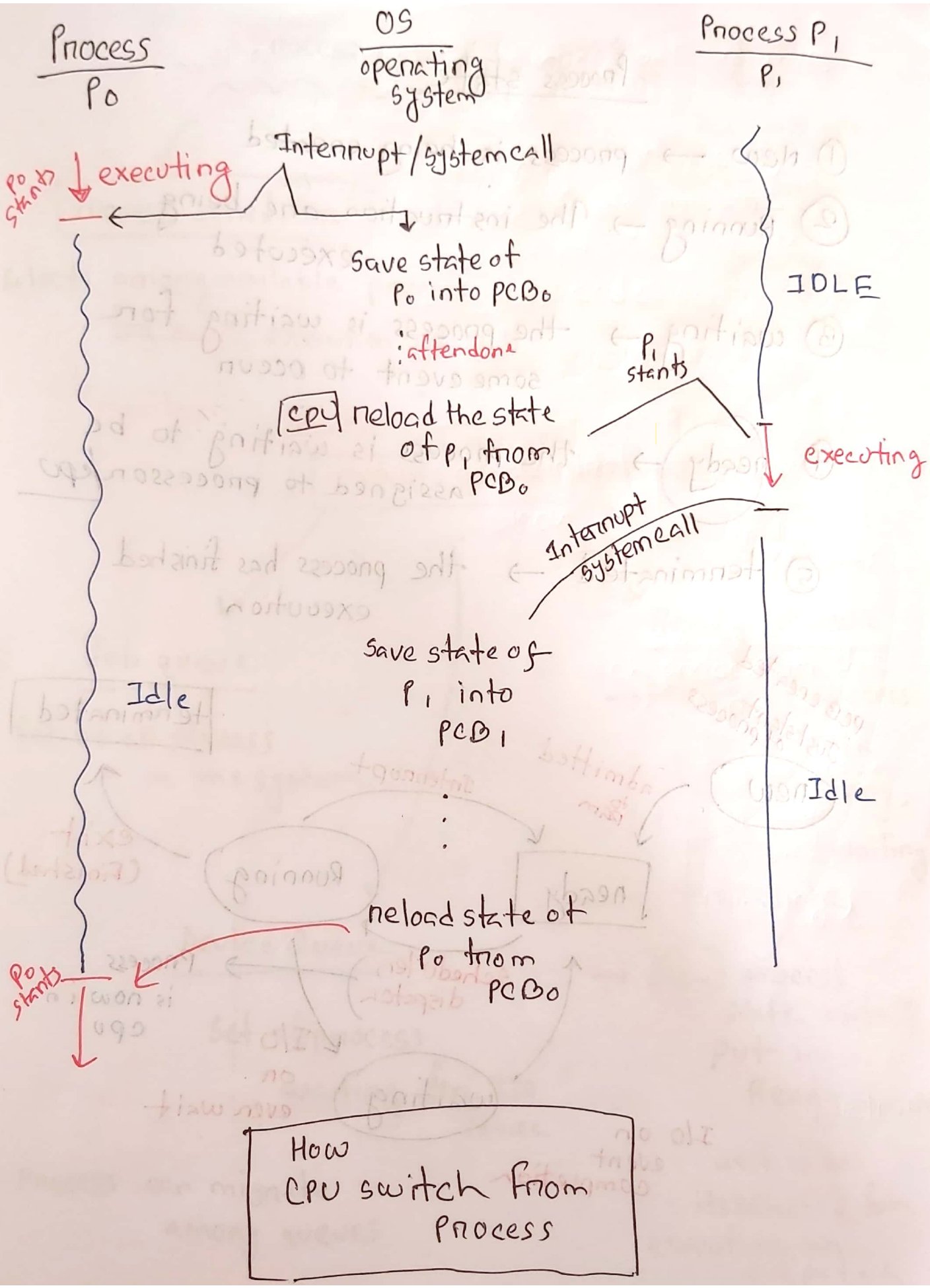
IO device allocated to process, list of open files



—o—

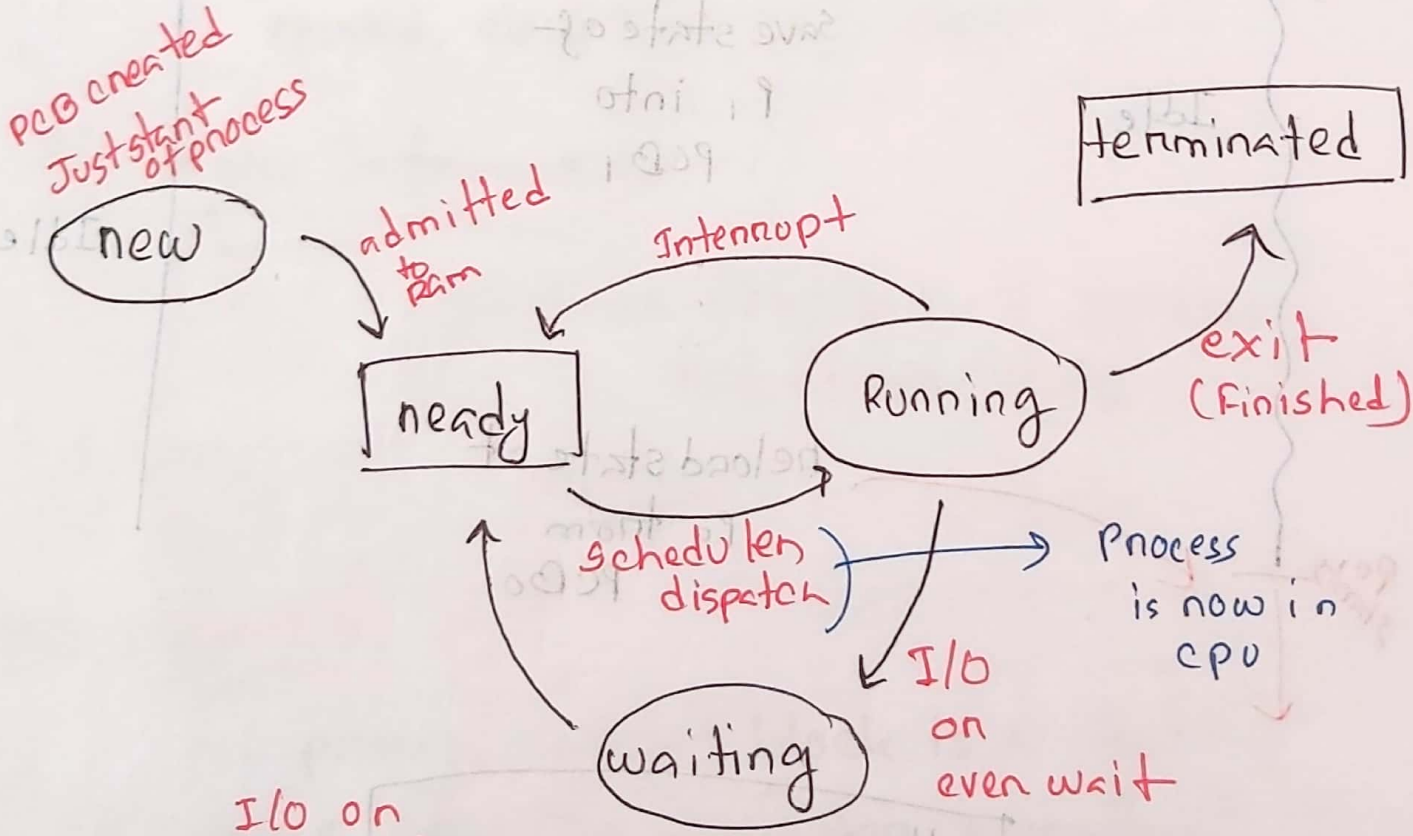
PCB definition

a process control block is a data structure used by computer operating system to store all the information about a process


Process
P0
OS
operating
system
Process P1
P1
Internupt/systemcall
P0 starts
executing
Save state of
P0 into PCB0
attendon
IDLE
P1
starts
CPU reload the state
of P1 from
PCB0
executing
Internupt
systemcall
Save state of
P1 into
PCB1
Idle
Idle
reload state of
P0 from
PCB0
P0
starts
How
CPU switch from
Process

Process state

1. New → process is being created
2. Running → The instruction are being executed
3. waiting → the process is waiting for some event to occur
4. ready → the process is waiting to be assigned to processor/cpu
5. terminated → the process has finished execution



Process scheduling

maximiz cpu use
quickly switch cpu for time shaning

Process scheduler

selects among available process for next execution on cpu

↳ maintains

scheduling queues of processes

Job queue

Set of all process in the system

Ready Queue

→ set of all process residing in main memory, ready and waiting to execute

Device Queue

Set of process waiting for I/O device

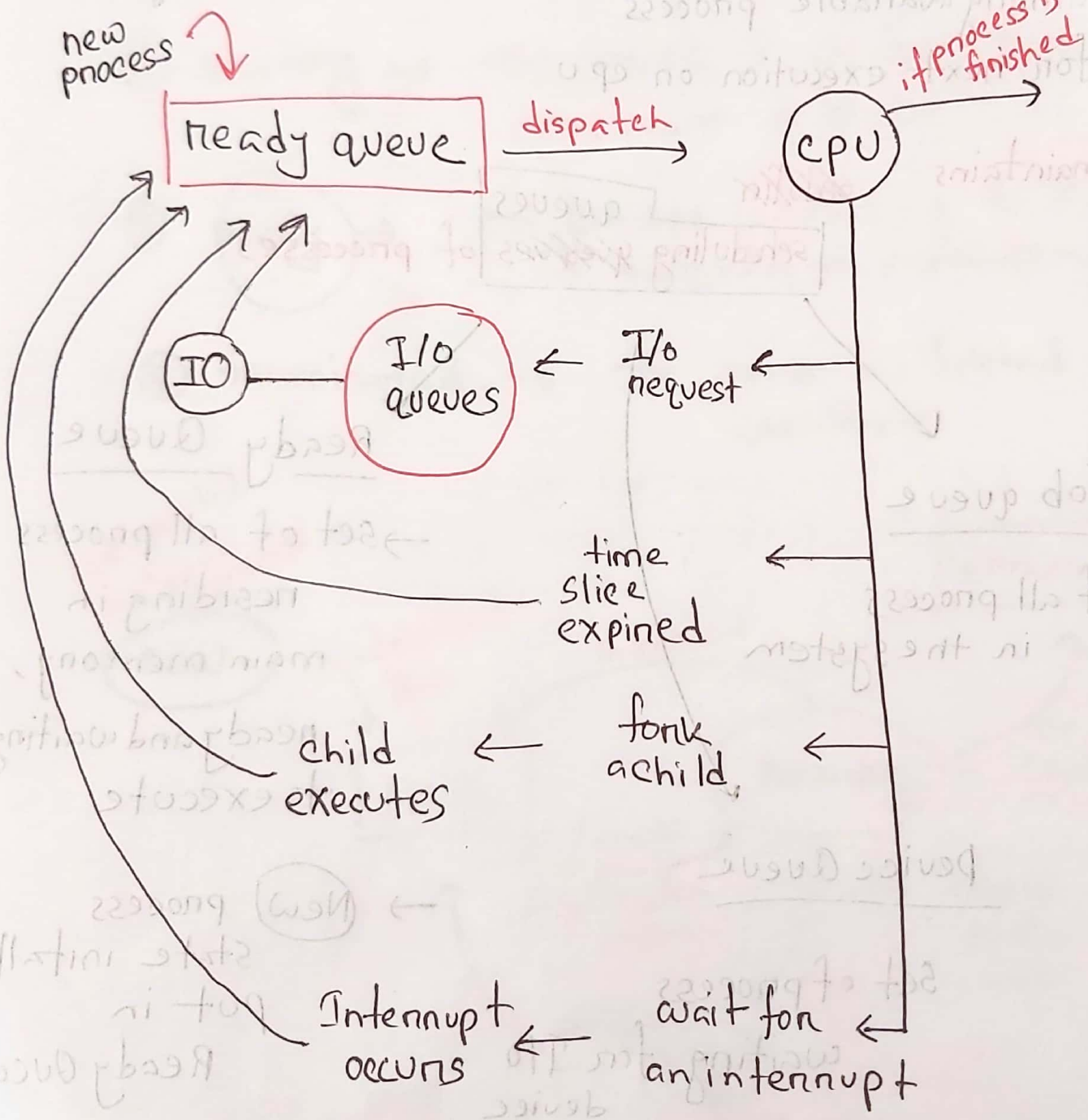
→ (New) process state init all put in Ready Queue

wait until its selected for execution on dispatch

* Process can mignate among queues

Representation of Process scheduling

Queueing diagram



Scheduler

Short term scheduler

→ CPU scheduler

→ which sends a process to CPU

→ select which process should be executed next and allocates to CPU

→ must be very fast
(is invoked / called frequently)
(miliseconds)

Long term scheduler

→ Job scheduler

→ selects which processes should be brought into the ready queue in main memory

→ strives for good process mix

→ may be slow
(is invoked / called in-frequently
(seconds, minutes)

→ controls the degree of multiprogramming

definition

how many program should allow at same time

- on admitted to main memory

Types of process

① I/O bound Processs :-

— spend more time doing I/O than computations

— many short cpu bursts

for how long you using the cpu

② cpu-bound process

— spend more time doing computations

— few very long cpu bursts

— o —

Mediom-term-scheduler :

— remove process from memory

(swaping) — store in disk

— bring back in RAM to continue execution

Context switch

definition important

② the system must save the state of old process and load the saved state for new process via context switch

① when a cpu switches to another process it is called context switch

(context of process represented in PCB)

while contex switch system does not do useful work

Definition :- ① + ②

—— o ——

Process Creation

Parent process create child process which in turn create other process, forming a tree of process

* Chrome Browser is a multiprocess with 3 different type of process

① Browser process : manage user interface, disk, network I/O

② Render process : render web pages, deals with HTML and javascript

③ Plug in process : for each type of plugin

— o —

Based no Interprocess communication

Process type

① Independent process
(not affected by other process)

② Cooperating process
(affected by another process)

Sockets

is defined as an end point for communication

(socket parts)
address
- Ipaddress: 161.25.19.8
- Port: 1625

∴ socket = 161.25.19.8.1625

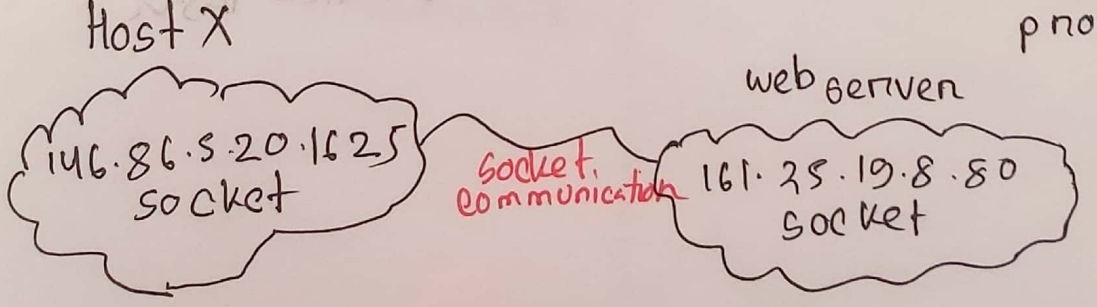
address of each computer

one device has same ip but diffenent port (port change based on tab)

communication consists between a pair of sockets

port < 1024 → well known ports (used for standard Internet service)

special Ip address: 127.0.0.1 (to refer to system on which process is running)



Process vs Job vs Task

① **Process** :- Process is a program under execution.

Process vs Program

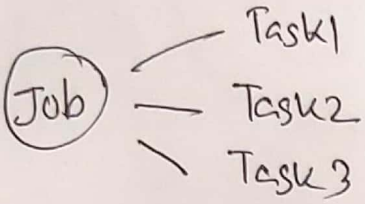
- ∴ a program can be defined as a set of instructions.
- Program is a passiven entity
  Process is a active entity
- Program became process when it comes to RAM
  - Program stays on Harddisk
  - process stay Harddisk, or CPU, main memony (RAM)
- 1 program can have multiple process

② **Task** :-
- Task is a unit of work being executed
- Task in os may be synonyous with process
- Task is a sub part of job
- Ex:- when we run a thread in Java - is called Task

(3) Job :-

→ a job is a complete unit of work under execution

- a job consists many task which in turn consists of many process



Synchronous	Asynchronous
→ Blocking is considered as synchronous	→ Nonblocking is considered as asynchronous
→ Blocking send has the senden block untill the message is received	→ Non blocking send has the senden send the message and continue
→ Blocking ~~recio~~ receive has the recieven block untill a message is available	→ Non blocking recive has the reciven receive a vaild message or null

(*) Process Altenate between
CPU bnusts and I/O bnusts

(*) CPU scheduler:-

when cpu becomes idle, it is the Job of cpu schedulen/shont tenm Schedulen to select anothen process from the neady queue to run next

(*) Schedulen dispatchen :- is a module that gives the control of CPU to the process selected by the schedulen

Types of socket

actually 3 types

1. Data gram socket (connectionless)
2. Stream socket (connection-oriented)
3. Raw socket
4. Sequenced packet socket

multi cast socket

access → Protocol

UDP (UPD)
User Datagram

(TCP)
Transfer control protocol

(ICMP)
Internet control message protocol

(IDP)
Internet data protocol

Degree of multiprogramming

Degree of multiprogramming describes the maximum number of process that a single processor system can accommodate efficiently.

Process	Thread
① Process mean any program is in execution	① Thread means a segment of a process
② Process takes more time to terminate	② Thread takes less time to terminate
③ takes more time for creation	③ Less time for creation
④ more time in context switching	④ Less time for context switching
⑤ Less efficient in communication	⑤ more efficient in communication
⑥ a system call is involved in it	⑥ No system call is involved
⑦ Process has Process control block	⑦ Thread has thread control block

Process scheduling

is the activity of process managen that handle the removal of the running process from cpu and the seleection of another process on the basis of a panticulan strategy.

Loop back addness

A loop back addness is distinct neserved ip addness 127.0.0.1 tonefen to system on which process is nonning

what is api?

Application program Intenface (API) is code that allows two softwane prognams to communicate with each other

The principal of Least privilege

the principal of least privilege states that a subject should be given only those privileges needed for it to complete its task.

* In UNIX, user is domain. True or false?

  - true

what is kernel

→ is a central component of OS that manages operation of computer and handware

→ acts a bridge between applications and data processing performed at handware Level using IPC and system calls

objectives

- To control disk, memory, task management
- To decide state of incoming process
- To establish communication between user level application and handware

(x) what is system call ? explain different type of system call.

→ **System Call** is a way for programs to interact with operating system.

5 Types system call

(1) Process call :-

- deals with process such as process creation & termination

Example :-

(Window)	(Linux)
CreateProcess()	fork()
Exit Process()	exit()
	wait()

(2) File management :-

- responsible for file manipulation creating, reading, writing into file

Example :-

(window)	(Linux)
• createFile()	• open()
ReadFile()	read()
writeFile()	write()
CloseHandle()	close()

③ Device management :-

- responsible for device manipulation
  reading / writing device buffer

Example :-

(windows)	(Linux)
SetConsolemode()	ioctl()
Readconsole()	read()

④ Information Maintenace

- handle information and its tranfer between os and user program

Example :-

(windows)	(Linux)
SetTimer()	alarm()
sleep()	sleep()

⑤ Communication

→ useful for interprocess communication

Example :-

(windows)	(Linux)
CreatePipe()	pipe()
MapViewOfFile()	mmap()

(x) How does communication take place in client server system ? Explain

→

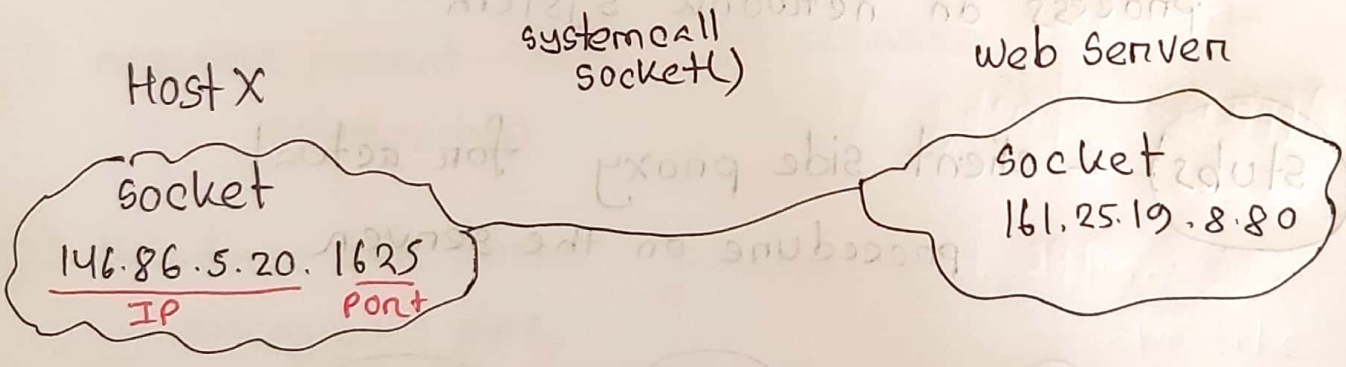
Client-server communication has two components client and server. The client send request to the server and the server responds to the client request

Three methods of Client-server communication

① socket → low level communication channel

→ a socket is an endpoint for communication

→ a socket is identified by an IP address and a port number



Two major form

① Connection Oriented / TCP (Transmission Control Protocol)

→ guranted that all packets will arrive in good condition at other end and in order recive

② Connectionless (UDP - user data protocol)

→ no guarantee that packets will delivered with undamaged and will receive in particular order

→ UDP ~~for~~ Transmission > TCP
faster

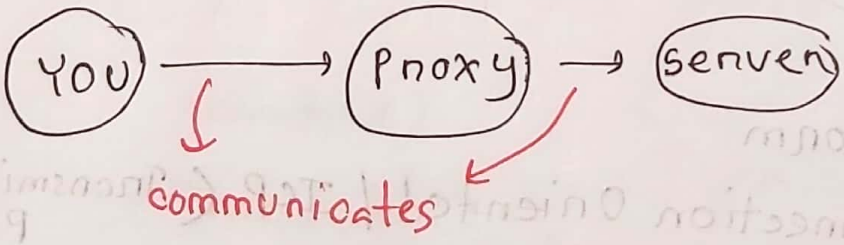
② RPC - Remote Procedure Calls

Function

* in socket we communicate with process
* In RPC " " " Functions / Procedure

RPC abstracts procedure calls between process on network system

stubs - client side proxy for actual procedure on the server



③ Pipe

- acts as a condult between two process to communiate

Ondinary Pipe

① ondinary pipe allows communication in standand producen consumen style

② unidinectional

③ Producens write to one end and consumen needs at othen end

④ nequine panent-child nelationship

⑤ windows calls these anonymous pipe

Named pipe

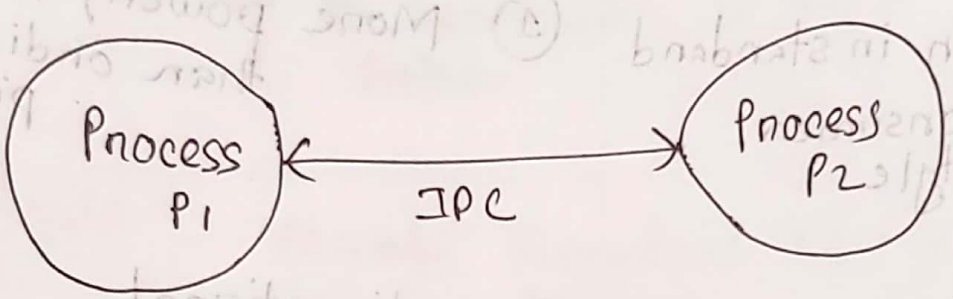
① Mone powenful than ondinary pipe

② bidinectional

④ Noneed panent chid nelationship

IPC – Interprocess Communication

– is the mechanism provided by OS that allows processes to communicate with each other



two types of process

1. Independent process
2. Co-operating process

why IPC?

IPC helps achieving

1. Computional speed up
2. Modulanity
3. Infonmation & data shaning
4. Pnivilege sepenation
5. Process can communicate with each other and synchronize their action

Approaches for Interprocess Communication