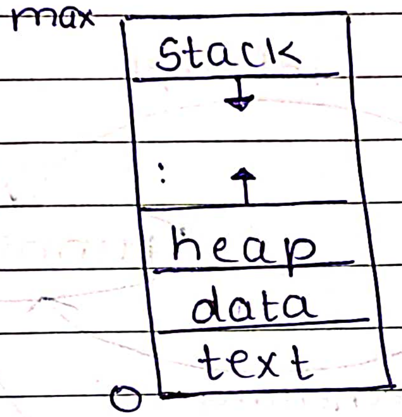
Process And Threads

- ☆ A program under execution is called as Process.
- Process execution must progress in sequential order.
- Process can be divided into four section namely, stack, heap, data, text.



Process in memory

- Sections :-

text - contains program code

data - contains global and static variables.

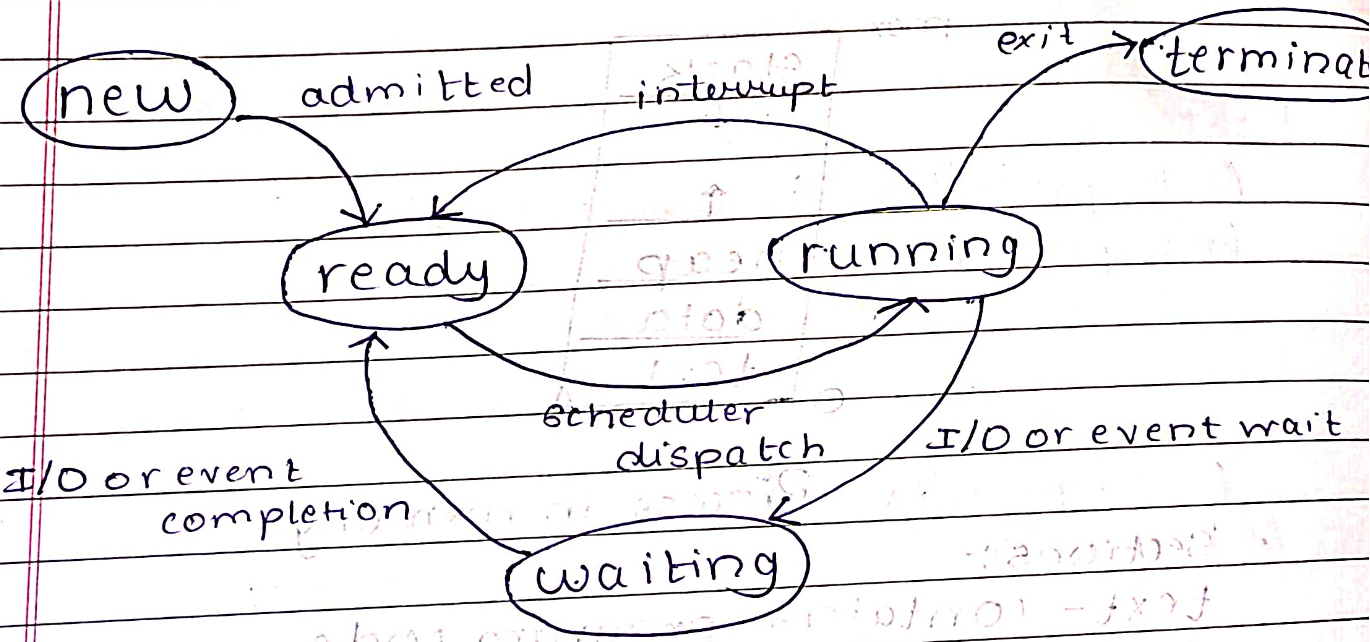
heap - Used for dynamically allocated memory to process during its run time.

Stack - contains temporary data.

☆ Process	Program
① Dynamic object	① Static object
② Active Entity	② Passive Entity
③ loaded in main memory	③ loaded in secondary memory
④ Time Span is Limited	④ Time Span is unlimited

★ Process States

- Current activity of a process is known as its state~~s~~.
- As process executes, it changes state.



- new : The process is being created.
- ready: The process is waiting to be assigned to a processor.
- running: Process is being executed
- waiting:- process is waiting for some event to occur.
- terminated :- Process has finished execution

* Process Control Block (PCB)

Process State
process Number
Program counter
registers
memory limits
list of open files
. . .

Process state :- Specifies current state of process

Process Number:- Each process has some unique identification number called as Process Indentification number (PID)

Program counter :- Pointer indicates the address of next instruction to be executed for process.

Registers :- Registers like stack pointer, index registers, general-purpose register etc where process need to be stored for execution for running state.

memory limits :- information like Memory used by the operating system.

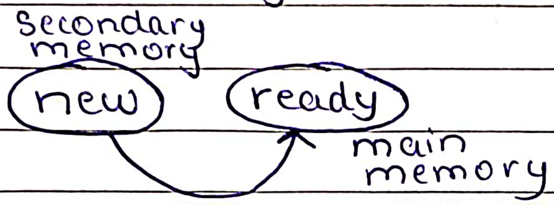
File management :- Handles lists of open files.

ST>MT>LT

★ Types of Schedulers

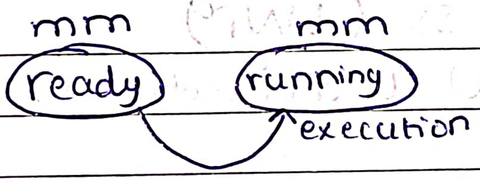
① Longterm Scheduler

- Selects a job or process from jobpool from secondary memory and loads into main memory.
- Executes less frequently
- controls degree of multiprogramming.



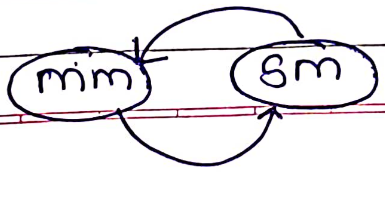
② Short-term Scheduler

- Selects a job from ready queue and submit to CPU.
- As it selects only one job at atime, it is invoked very frequently.
- Main objective is to increase System performance.



③ Mid-term Scheduler.

- Mid-term Scheduler is part of swapping so it is known as swapper/swap scheduler.
- mid-term Scheduler ~~process~~, ~~it~~ swaps the process from main-memory to secondary-memory and vice-versa.
- As it removes the process from memory it reduces the degree of multiprogramming



Page No.
Date

☆ Context Switch

- Switching CPU to another process requires saving the state of the old process and loading the saved state of the new process. This task is known as context Switch.
- Enables multiple process to use single CPU.

☆ Threads

- A thread is a flow of execution through the process code, with its own program counter that keeps track of instructions to execute next, system registers which hold its current working variables and stacks which hold execution history.
- Light-weight process (LWP)
- Basic unit of CPU utilization.
- Subset of process.
- Run in shared memory spaces.
- Shares file description.
- Shares file system context.
- Shares signal handling
- Threads are dependent.

☆ User Level Threads	Kernel Level Threads
① ULT are faster to create and manage	① KLT are slower to creater andmanage
② Threads implemented at user level known as ULT	② Threads implemented at Kernel level known as KLT.
③ Can run on any O·S	③ Specific to O·S

☆ Benefits

① Efficient Communication:- Communication between multiple threads is easier as the threads shares common address space.

② Computational Speedup:- On single processor ~~setup~~, System, a process can be executed speedily by creating multiple threads in the process.

→ contd [next pg]

Page No.
Date

★ Benefits

(3) Responsiveness :- Allows a program to run even if part of. it is blocked using multithreading

(4) Resoure sharing :- All threads of a process shares resources such as memory, data, files etc.