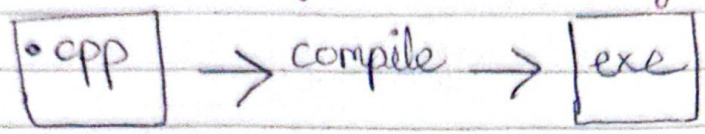
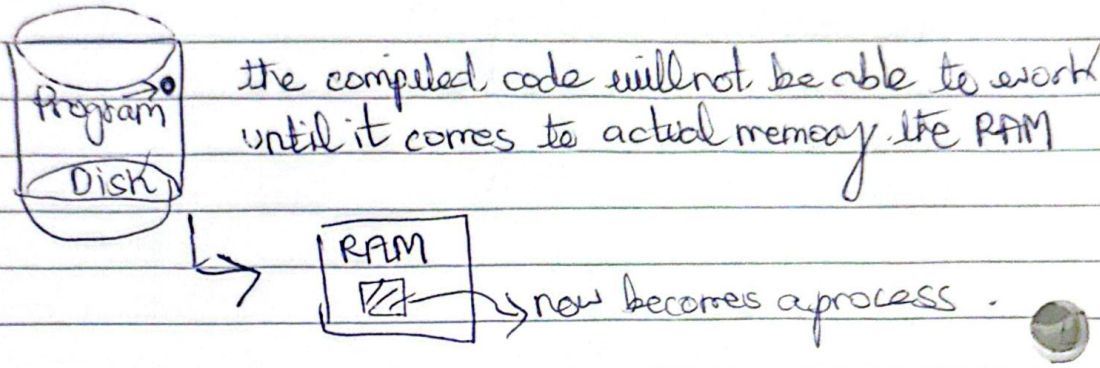
Multitasking vs Multithreading.



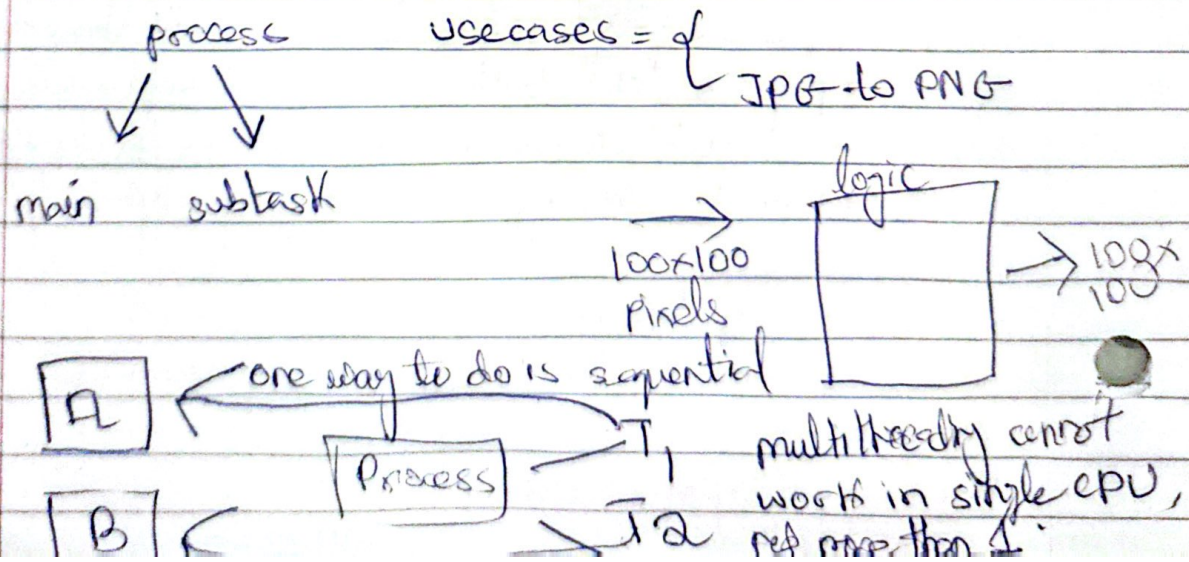
when we double click an exe it becomes a process and process is a program under execution

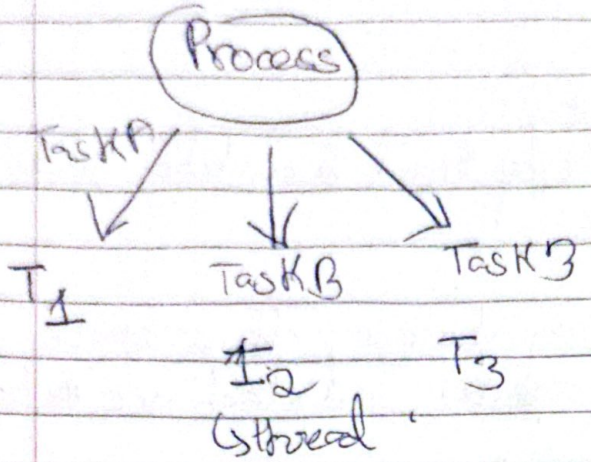


the compiled code will not be able to work until it comes to actual memory the RAM

Thread

thread is a light weight process, which can be executed independently, is a subprocess inside a process, it writes data to cloud which is an asychous work.



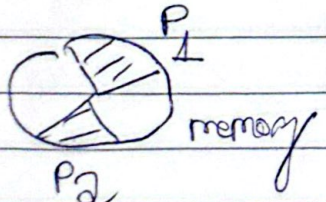


there will be no isolation between two threads, part of same process

Multitasking

(1) there are more than 1 process concept

(2) isolation and memory protection



(3) process schedule.

Processes
↓
threads are scheduled based on priority

resources are shared between threads of that process

(3) threads scheduled.

- Multi threading

(1) more than one thread

(2) not isolation and memory protection

Difference

- Thread context switching
OS saves current state of thread & switches to another thread of same process.

doesn't include switching of memory address space.
(But program counter, registers and stack are included)

CPU's cache is preserved

fast switching

- Process context switching

OS saves current state of process & switches to another process by restoring its state

slow switching

includes switching of memory address space
slow switching
CPU's cache state is flushed.

each process has one cache