

AGENDA

- ✓ = 1> lifecycle of Process
 - ✓ = 2> Thread
 - ✓ = 3> Multicore / single core
 - ** 4> concurrency v/s Parallelism
 - = 5> create own thread
- }

start by 9:05 PM

*) PROCESS

Lifecycle of Process:

developer → coded → uploaded (Playstore)

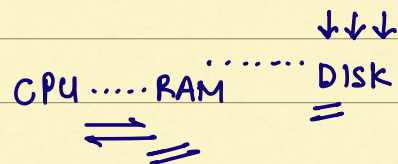
PROGRAM:

(Disk)

set of instructions that we
want to run.

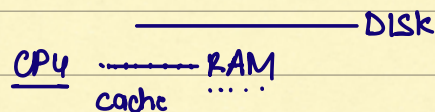
PROCESS:

Program in Execution.
(RAM)



12GB
8GB
4GB

128GB/256GB



PROGRAM → install



(RUN) → PROCESS.

*) How PROCESS RUNS IN CPU:

Google drive docs.

#

- 1.) Auto save
- 2.) Grammar check
- 3.) suggestions
- 4.) history
- 5.) word count

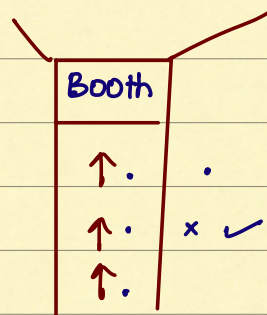
Main() — wordcount() — ys....
grammarcheck()

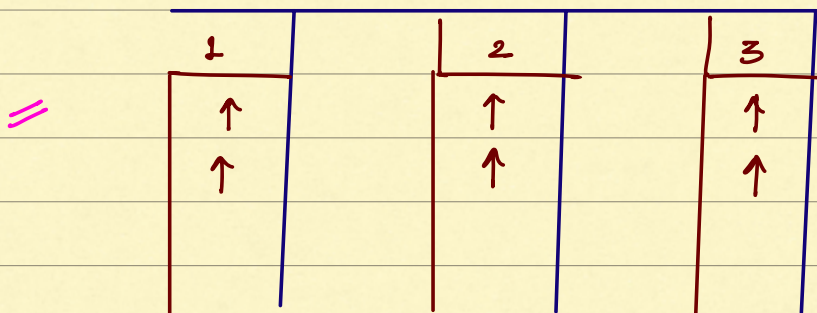
→ Threads.

lightweight process.

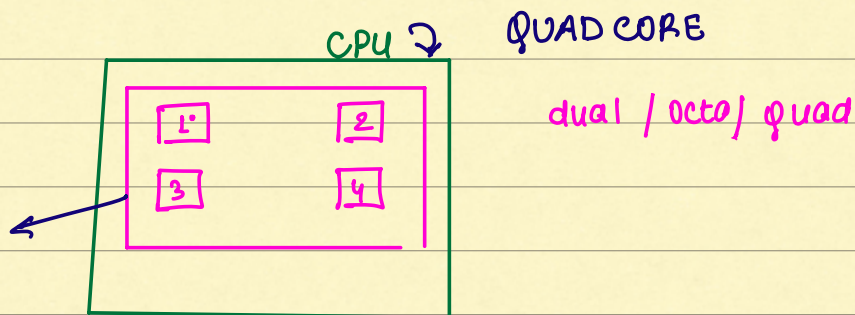
" Thread is smallest unit
of CPU execution "

Eg: Toll tax system



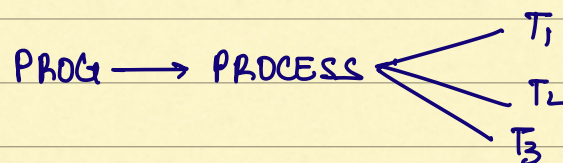


→ TOU booth: CORE.



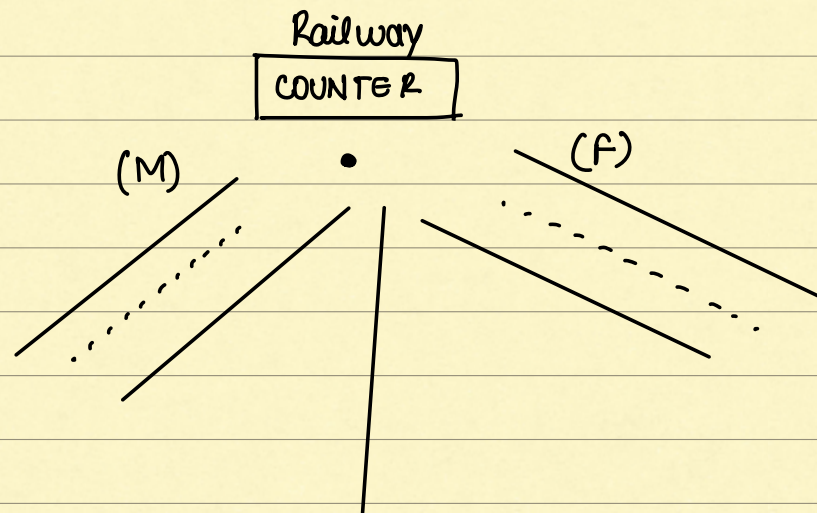
Each core takes a thread at one time

1 core executes → 1 thread at a time
↓
(task)



TOU booth: core
lane: thread
& core: task



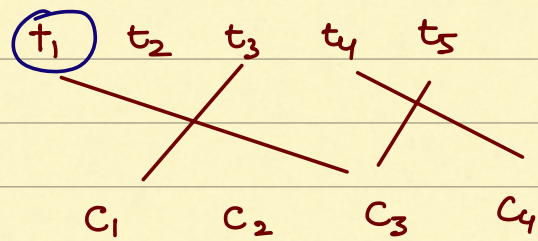


Thread - smallest unit of execution.

$\begin{matrix} \text{---} \\ \text{---} \\ \text{---} \end{matrix}$
 (t_1, t_2, t_3)

"Hello world" — at least 1 thread.

O.S. →

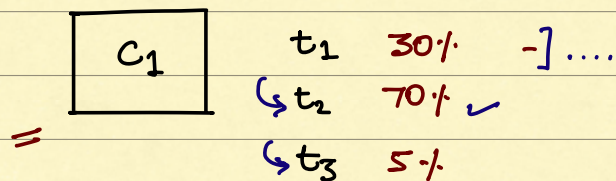


→ scheduling Algo.

*2) CONCURRENCY v/s PARALLELISM:

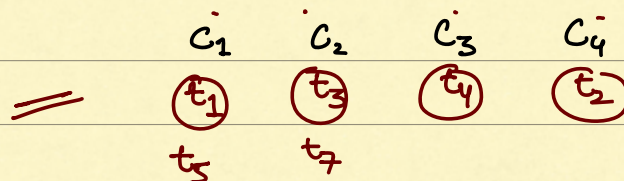
1.) Concurrency:

Multiple things executing in
system, might Not
at once.



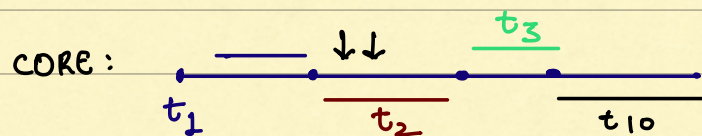
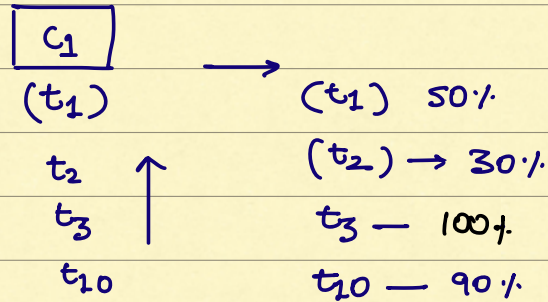
2.) parallelism:

concurrency ⊕ Running at
same time



10:51 PM IST

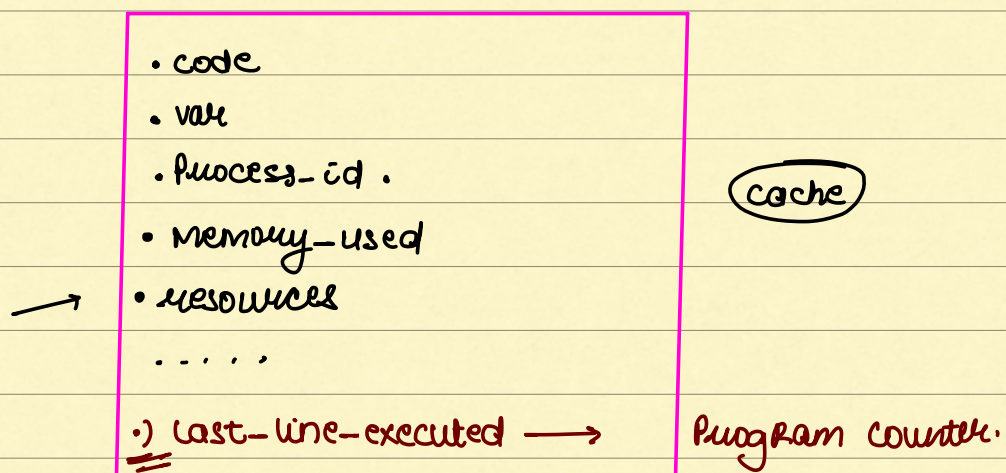
*) CONTEXT SWITCHING:

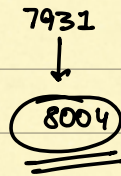


→ Problem ??

context switching - switching b/w multiple threads.

P.C.B — Process Control block





C.S. → time taken by CPU to save
= Get data FROM PCB.

*} HOW TO CREATE THREADS:

S1.) In multithreaded code-

Never think in terms of creating
thread

→ Instead think in terms of
task.

S2.)

"Print Hello" — from a diff thread.

```
class HelloWorldPrinter
```

S3.) Implement Runnable interface in your class

S4.) Provide definition

```
run() {
```

```
    s.o.p("Hello world")
```

```
}
```


ss.) create threads & execute