

### LEC-3: Multi-Tasking vs Multi-Threading



**Program:** A Program is an executable file which contains a certain set of instructions written to complete the specific job or operation on your computer.

- **It's a compiled code.** Ready to be executed.
- Stored in Disk

**Process:** Program under execution. Resides in Computer's primary memory (RAM).

**Thread:**

- Single sequence stream within a process.
- An independent path of execution in a process.
- **Light-weight process.**
- Used to achieve parallelism by dividing a process's tasks which are independent path of execution.
- E.g., Multiple tabs in a browser, text editor (When you are typing in an editor, spell-checking, formatting of text and saving the text are done concurrently by multiple threads.)

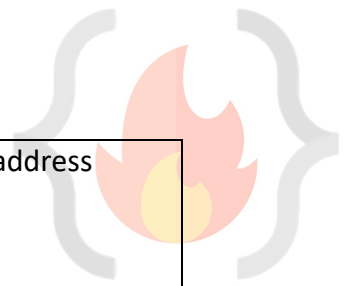
Multi-Tasking — between process	Multi-Threading
The execution of more than one task simultaneously is called as multitasking.	A process is divided into several different sub-tasks called as threads, which has its own path of execution. This concept is called as multithreading.
Concept of more than 1 processes being context switched.	Concept of more than 1 thread. Threads are context switched.
No. of CPU 1.	No. of CPU $\geq 1$ . (Better to have more than 1) <b>otherwise sqsly work..</b>
<b>Isolation and memory protection</b> exists. OS must allocate separate memory and resources to each program that CPU is executing.	<b>No isolation and memory protection,</b> resources are shared among threads of that process. OS allocates memory to a process; multiple threads of that process share the same memory and resources allocated to the process.

**Thread Scheduling:**

Threads are scheduled for execution based on their priority. Even though threads are executing within the runtime, all threads are assigned processor time slices by the operating system.

**Difference between Thread Context Switching and Process Context Switching:**

Thread Context switching	Process context switching
OS saves current state of thread & switches to another thread of same process.	OS saves current state of process & switches to another process by restoring its state.



Doesn't includes switching of memory address space. (But Program counter, registers & stack are included.)	Includes switching of memory address space.
Fast switching.	Slow switching.
CPU's cache state is preserved.	CPU's cache state is flushed.

CodeHelp