**Thread:**

A thread is a path of execution within a process. A process can contain multiple threads.

**Process vs Thread:**

A single thread of process is represented by its address space.

In the process, virtual memory addresses are created every time we create.

In thread, virtual memory address is not created and VMA of process is used.

**Uses of Threads:**

There can be multiple threads belonging to the process.

We can process different inputs if we use threads as threads execute parallely.

To have different threads operate on different portions of the code in a specific function.

Threads can be prioritised based on the importance of tasks they have.

Threads are useful to hide latency which is associated with input-output operations and this is also useful even in a single CPU as it hides latency of operations.

**Thread Mechanisms:**

To create and manage threads.

To safely coordinate among threads running concurrently in the same address space.

Mutual Exclusion - gives exclusive access to only one thread at a time.

It uses mutex to mute the waiting processes.

Waits on other threads based on specific conditions before proceeding, condition variable.