# OS Basis – Process Management

## Definition of Process and Process model

Diagram

Description automatically generated

## Process operations and states

Timeline

Description automatically generated

Diagram

Description automatically generated

## Threads model: Definitions, model and implementation

Diagram

Description automatically generated

## Threads vs Processes

A process is an instance of a running program in an operating system. It has its own **memory space,** resources, and system state.

A thread is a lightweight process that executes within a process. It shares memory and resources with other threads within the same process, making it efficient for tasks that can run concurrently.

In summary, a process is a self-contained execution environment, while a thread is a smaller unit of execution within a process.

Table, timeline

Description automatically generated

Diagram

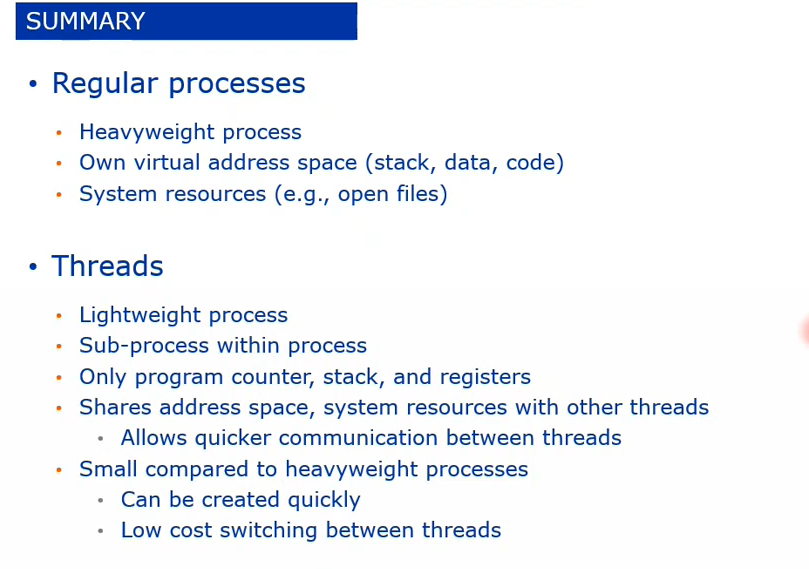
Description automatically generated

Question ; what is the problem when one process have multiple thread ?

multiple threads within the same process share the same memory space, which means that they have access to the same data. If one thread modifies the shared data, it can potentially affect the behavior of other threads that are accessing the same data. This is called a race condition and can lead to unexpected results or data corruption. To prevent this from happening, proper synchronization mechanisms, such as locks or semaphores, must be used to ensure that only one thread can access the shared data at a time.

Text, timeline

Description automatically generated



## Basis of Scheduling: Scheduler, Dispatcher, preemptive vs. non-preemptive, context switching

## Scheduling Algorithms

## Scheduling Criteria

## Scheduling Evaluation

# OS Basis – Memory Management

## Definition of address space

## Swap memory vs Virtual memory

## Virtual memory

### Mapping

### Address translation

### Page replacement

## Segmented memory

# OS Basis – File System