# Lab 3 Scheduling – Video Summary

## 1. Introduction to Scheduling and Process Types:

In the video, the concept of scheduling within an operating system is introduced, with a specific focus on how this works in Windows and Linux environments. This introduction forms the foundation for understanding how modern operating systems manage processes efficiently.

The video emphasizes the importance of distinguishing between two primary process types: interactive and computation processes. Interactive processes, exemplified by applications like text editors and web browsers, place a high premium on response time. The speed at which these processes respond to user input is critical to ensuring a smooth and responsive user experience. In contrast, computation processes, often seen in the background, are more concerned with consistent CPU access for their operations. These processes aim to efficiently complete their tasks without interruption.

## 2. Scheduling Algorithms and Multiple Queues:

A core aspect of the video is the discussion of scheduling algorithms tailored to these two process types. For interactive processes, the round-robin scheduling algorithm is recommended. What's notable is that the quantum value, the time allocated to each process, doesn't significantly affect this recommendation. The key here is ensuring that interactive processes are ready to respond quickly to user input.

Computation processes, on the other hand, are better served by a first-come, first-served scheduling policy. This approach allows them to complete their CPU-bound tasks without interruptions. The video also introduces the idea of using two separate ready queues. One queue is designated for interactive processes and employs the round-robin scheduling policy. The other queue is reserved for computation processes and utilizes the first-come, first-served approach. This separation helps ensure that each process type receives the scheduling treatment that aligns with its specific requirements.

## 3. Handling Priorities and Starvation:

The video highlights the importance of managing priorities within the scheduling process. In this context, interactive processes are granted higher priority. They may preempt computation processes if necessary to maintain low response times. Preemption is the process of interrupting a computation process to allow an interactive process to execute and maintain responsiveness.

However, the video acknowledges that priority-based scheduling can potentially lead to a problem known as process starvation. This occurs when lower-priority processes do not

receive an adequate share of CPU time, which can negatively impact system efficiency. To address this issue, the video introduces the concept of "aging." Aging involves gradually increasing a process's priority over time, ensuring that lower-priority processes eventually receive their fair share of CPU time. This approach helps prevent process starvation, ensuring that all processes are given an opportunity to run efficiently.