Learning Guide Unit 2

Reading Assignment

As you go through the readings and watch the video, consider the following:

1. How do processes and threads differ in terms of system performance, and what role does the Process Control Block (PCB) play in managing these processes?
2. Compare pre-emptive and non-pre-emptive CPU scheduling techniques, and explain how context switching ensures stability during multitasking.

Read:

1. Hailperin, M. (2019). *Operating systems and middleware: Supporting controlled interaction* https://gustavus.edu/academics/departments/mathematics-computer-science-and-statistics/max/os-book/.Licensed under a Creative Commons Attribution-ShareAlike 3.0 Unported License.

    - View the online book.
    - Read Chapter 2: Threads (pp. 21 till pp.39)
    - Read Chapter 3: Scheduling (pp. 45 till pp. 81)

    The chapters on threads and scheduling introduce core concepts essential for optimizing the performance of modern operating systems. The section on threads explores the creation, management, and synchronization of lightweight processes, highlighting their role in achieving multitasking and parallelism. It discusses multithreading models, the benefits of concurrency, and the challenges of thread coordination. The scheduling chapter delves into various CPU scheduling algorithms, both pre-emptive and non-pre-emptive, and explains how they allocate processor time to ensure efficient execution of processes. Together, these chapters provide a comprehensive understanding of how threads and scheduling contribute to system performance and resource management.

Additional Readings:

1. Arpaci-Dusseau, R. H., & Arpaci-Dusseau, A. C. (2018). *Operating systems: three easy pieces (1.01 ed.).* Arpaci-Dusseau Books. https://pages.cs.wisc.edu/~remzi/OSTEP/

    - View the online book.
    - Read Chapter 7: Scheduling ; Introduction (pp. 59 till pp. 68)

    The scheduling chapters in this book introduce essential concepts and strategies for managing CPU and process scheduling in operating systems. These chapters cover various scheduling algorithms, including pre-emptive and non-pre-emptive approaches, and explain how they impact system performance, responsiveness, and resource utilization. Topics such as round-robin, priority scheduling, and multi-level queues are discussed in detail, highlighting their use cases and trade-offs. Through practical examples and analysis, the chapters offer a comprehensive understanding of how scheduling decisions affect multitasking, process prioritization, and overall system efficiency in diverse computing environments.

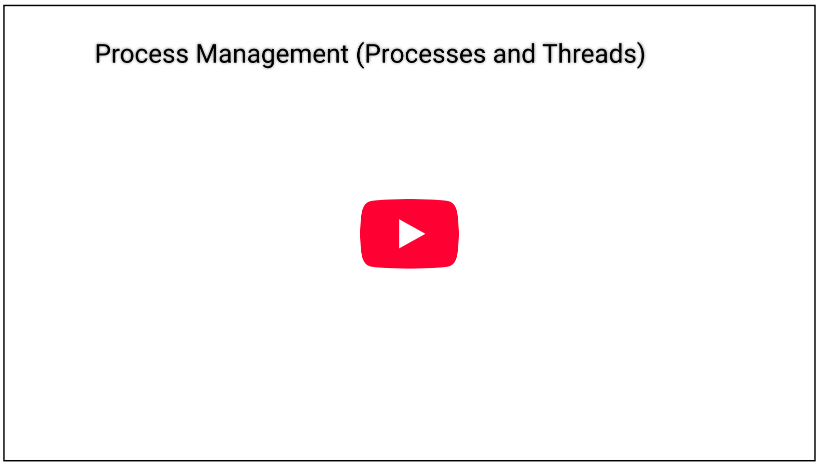
Watch:

1. Neso Academy. (2019, May 24). *Introduction to threads* [Video]. YouTube.

- This video covers the basics of threads in operating systems, exploring the differences between single-threaded and multi-threaded processes. It also highlights the key benefits of multi-threaded programming, such as improved performance and resource efficiency.

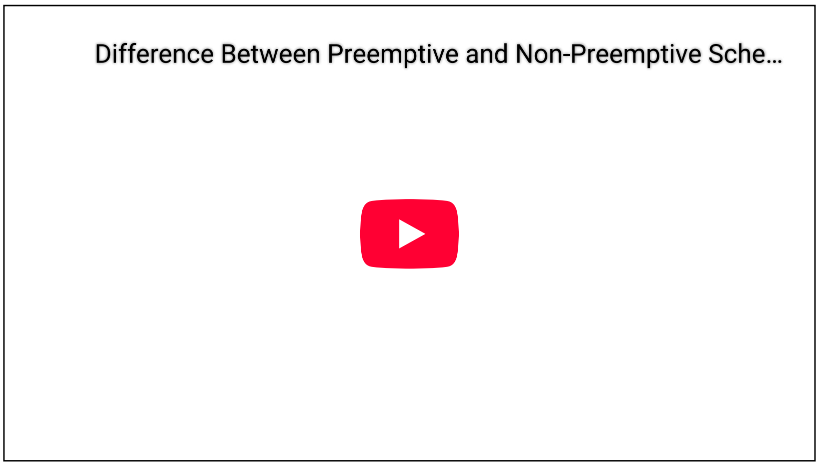
2. Neso Academy. (2018, June 29). *Process management* (Processes and threads) [Video]. YouTube.

- This video provides an introduction to process management, explaining the key differences between processes and threads. It covers how threads function within processes and their impact on system performance and efficiency.



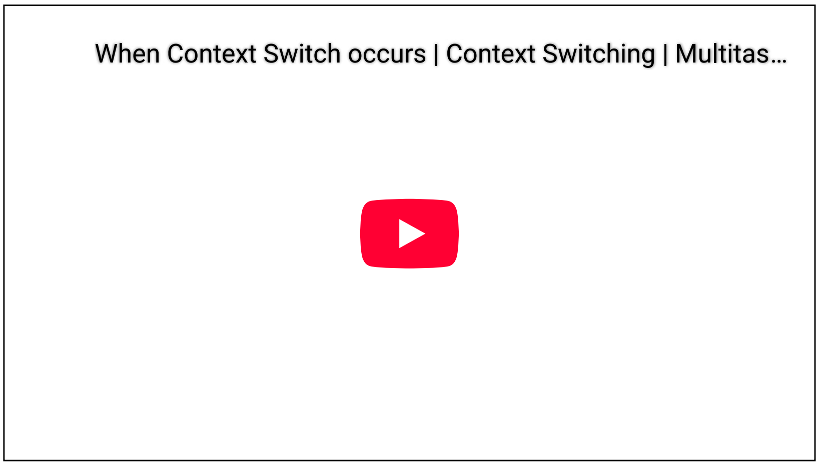
3. About Technology. (2022, March 31). *Difference between preemptive and Non-Preemptive scheduling* [Video]. YouTube.

- This video explains the key differences between preemptive and non-preemptive CPU scheduling, highlighting how preemptive scheduling allows for task interruption and resource reallocation, while non-preemptive scheduling completes tasks without interruption.



4. iFocus Institute. (2019, September 14). *When Context Switch occurs | Context Switching | Multitasking* [Video]. YouTube.

- This video explains the concept of context switching, focusing on its importance in managing multiple processes within an operating system. It also demonstrates how context switching ensures seamless multitasking and efficient CPU scheduling in various real-world scenarios.



When Context Switch occurs | Context Switching | Multitas...