

## A PROPER PLAN ON HOW TO STUDY OPERATING SYSTEMS EFFECTIVELY

● Planning is the process of thinking about activities required to achieve a desired goal,

● Desired goal :

- Understand the basics of operating system
- Score good marks in competitive exams like GATE
- Pass your college semester exam of OS
- Crack your interviews

● Activities:

- Follow the order of the topics as I mention below
- Understand what to study vs what to not
  - Studying for GATE vs studying for semester exam
  - Theory vs Examples
- Ask questions whenever you have doubt - google it or ask in comments,
- Solve as much examples as you can
- Pause the video and try to solve problem. Then watch my solution
- Go slow. Don't aim for just completing the playlist. Exception exists for college students who are just studying for passing it 😊. Otherwise, take your time.
- Read my notes on each topic

● Before I go through the planned syllabus for OS, let me discuss what I expect from you

- Subscribe if you like the content - positive feedback
- Dislike and comment on the video - ~~negative feedback~~
- Issue with quality of the video or sound - ~~comment~~
- I will fix the issues and upload video again

## **CONTENTS**

### **1. Overview of operating system**

- a. **What is an operating system - OS vs kernel**
- b. **Functions of an operating system**
- c. **System calls in operating system**
- d. **Interrupts in operating system**
- e. **Signals in operating system - interrupts vs signals**

### **2. Process Management**

#### **A. Process Concepts**

- a. **What is a process - programs vs process**
- b. **How does OS keeps track or represent a process - Process Control Block(PCB)**
- c. **Different states a process goes through**
- d. **Scheduling queues in operating system**
- e. **Schedulers in operating system**
- f. **Context switch in operating system**
- g. **Dispatcher in operating system**
- h. **Process creation and lifecycle management with fork(), exec(), wait() and exit() system calls**
- i. **Examples of process creation with fork()**
- j. **Basics of interprocess communication**

#### **B. Multithreaded programming**

- a. **What is a thread**
- b. **Process vs Thread**
- c. **User level threads vs kernel level threads**
- d. **Multithreading models**

#### **C. Process scheduling**

- a. **Already covered the scheduling queues and schedulers in process concepts section. If forgot, revisit it**
- b. **CPU bound vs IO bound process**
- c. **Preemptive vs nonpreemptive scheduling**
- d. **CPU scheduling criteria**
- e. **Scheduling algorithms with advantages, disadvantages and examples**
- f. **Examples with different arrival time, IO and context switch**

### **3. Process coordination**



## **A. Process synchronisation**

- a. Race condition in operating system
- b. Critical section and critical section problem
- c. Software solution to critical section problem - Peterson solution and its drawbacks
- d. Hardware solution to critical section problem - test and set vs swap instruction solution
- e. Semaphore and mutex
- f. Spin lock vs mutex vs binary semaphore vs counting semaphore
- g. Producer consumer problem and its solution using mutex and semaphore
- h. Reader writer problem and its solution using mutex and semaphore

## **B. Deadlocks**

- a. What is deadlock and its characteristics
- b. How to detect deadlock
- c. Methods for handling and preventing deadlocks

## **4. Memory management**

### **A. Memory management techniques**

- a. Contiguous memory management techniques
- b. Fixed partitioning method vs variable partitioning
- c. First fit, next fit and best fit algorithms
- d. Logical address space vs physical address space
- e. Memory mapping and protection in contiguous memory management
- f. Noncontiguous memory management techniques
- g. Paging in operating system
- h. Pages, frames, page table concepts
- i. TLB in operating system
- j. Memory protection in paging
- k. Segmentation memory management technique

### **B. Virtual memory management**

- a. What is virtual memory and virtual memory management
- b. Demand paging in operating system
- c. Page replacement techniques
- d. FIFO vs Optimal vs LRU page replacement algorithm with

examples

- a. **Frame allocation techniques and thrashing**

## 5. **Storage management**

- a. **File representation and operation on files**
- b. **Types of files**
- c. **Access methods for files**
- d. **Directory and its structure**
- e. **File protection**
- f. **How a file system is implemented**
- g. **Disk scheduling**
- h. **RAID levels**