- Virtual memory, demand paging
    - Page replacement policies
        - Follow the algorithms for some example cases
        - The pros and cons of the algorithms
        - The relation of replacement policies to locality and working set sizes
        - Additional issues with page replacement
    - Memory page sharing
        - How to do it and the pros and cons
    - Memory protection
        - How does the process address space look like? Why?
        - What are to be protected?
        - Why there would be access violations?
        - Why there would be the buffer overflow problem and how exactly it happens?
        - Given scenarios and decide whether they may happen and whether the protection mechanisms can help
    - Some questions may involve earlier materials in memory management
- Disk
    - Disk basic terminologies, arm, track, sector, their characteristics
    - Disk access time: seek time, rotational delay, transfer time
        - Some cases consider average
    - Disk access time for files
        - Allocation strategy with access time
    - Disk arm scheduling algorithms and their impact in file systems
    - RAID with different designs
        - Basic design and data layout on parallel disks
        - Read/Write behaviors and performance
        - Fault tolerance capability and performance after failure(s)
        - Comparisons between the designs
    - RAID-DP
        - Understand the disk layout and recovery mechanisms
- Clock
    - The design of the clock itself (the registers and their values)
    - OS supported timer, the data structure
        - Why this data structure?
        - How it works, especially for some specific examples
    - Potential pros and cons with an alternate design
- Terminal
    - The basic concept on how keyboard, display, and OS are interacting
    - The intelligent keyboard-monitor interactions
- File system
    - Basic file system concepts
        - Different allocation strategies and their pros and cons
        - Which method is suitable for what type of file accesses, may give scenarios to ask you to rank various methods
    - Basic Unix file system

- Inode structure, may change it to test your understanding
- Directory structure and how to find the inum of a file
- From inum, how to find the inode of the file
- From inode, how to find a certain data block of the file
- If we want to access a certain block of the file, where is it on the disk (track number, sector number, etc.)

- The cylinder groups in FFS, its design goal, and detailed design
    - Same as the key points as above, but associate with cylinder group
- Journaling and fault tolerance in FS
    - Basic hardware support for dealing with failures
    - Given a failure scenario, what would be the consequence for the file system? E.g., file block lost, wrong data or garbage in the block, etc.
    - If some problem is detected, what activities have occurred to cause the problem? What actions can be taken to recover from the problem?
- LFS
    - Basic concept of the approach
    - inode block problem and the imap solution
    - Layout of the information in a segment and why we need each part
    - Final solution: imap in CR, why it has to be there in CR? Why the solution can support delayed write?
    - How to perform read?
- NFS
    - The concept of v-node and its relation to i-node
    - The basic flows regarding how to mount an external disk to a local directory (or drive) and how to retrieve an NFS file, what data structures need to be maintained
    - The data structure for maintaining file descriptors of a process
- File system design concepts
    - Compare different file system designs in general and for some specific scenarios
    - Given a different FS design or alter an existing design, what are the pros and cons or even flaws

- Processor scheduling
    - Basic processor scheduling policies
        - Follow the algorithms to schedule a given set of tasks
        - MLFQ and Unix MLFQ are especially important
        - Potential problems, pros and cons, comparisons
        - Compare the algorithms for some given scenarios
    - Real time scheduling algorithms
        - Follow the algorithms to schedule a given set of tasks
        - Compare the algorithms for some given scenarios
        - Fully understand priority inversion and resolution
    - May give modified algorithms
        - Understand their pros and cons and compare them with existing algorithms
    - Some of the questions may involve the earlier materials on process states

- Project related questions
    - Fork, thread, pipe, signal, socket
    - The basic concept of these functions
    - Under an example scenario, which ones are the best to use and “why”
        - May need to use multiple functions
        - There will be no question on thread specific behaviors, but combined with other functions
    - The behaviors of a program with these functions
        - Try out the behaviors if these functions are called differently
    - Projects 2 and 3
        - Basic designs
        - Thread synchronization and semaphore coding for Project 3
        - Different handling between the ready queue and terminal queue (how they are different and why)
        - Socket communication steps and select command

- Sample Questions
    - Variations of questions in the homework assignments
    - Concept based questions
        - Which of the following page replacement algorithms is most similar to the least recently used algorithm?
        - Why is there the buffer overflow problem when any virtual address is supposed to be valid?
        - Rank the following processor scheduling algorithms in terms of the response time they yield
        - Rank the following parallel disk designs in terms of their fault tolerance levels
        - Most likely this type of questions will involve algorithms that are modified from what you have learnt
    - More potential questions have been discussed at a high level in the review, you need to study in more depth according to the review statements
    - The sample questions are given to show the type of questions you may get besides those appear in homework #3, but the exam questions will be different