OS :: Module 2 Quiz Study Guide - FA24.1

Overview

The module two quiz will broadly cover memory virtualization. For this course the book itself is an excellent study guide and you should expect questions that tie very closely to how the ideas are presented in your book. Note that I omit the dialogue chapters.

The slides cover the book well, however, it is a good idea to read/review the book and to use the book as a reference while studying the slides. Students generally want to use the slides, however, if you rely to heavily on them you may find that you have a shallow understanding. This is a really readable book and I would consider spending significant time working your way through the material in the book while using the slides to anchor your memories of the lectures.

Powers of 2

- 17 numbers separated by 16 commas, from 2^0 20 through 2^{16}, memorize!
  - *1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048, 4096, 8192, 16384, 32768, 65536*
  - ***You must have 100% of the digits correct for any points! Any incorrect digit will yield zero points!***

Memory Virtualization

Chapter 13 :: Address Spaces

- Make sure that you understand the different models of address spaces presented and be able to relate them to different requirements in the evolution of operating systems.
- Why do we need something like paging or segmentation?
- What involvement does the operating system have in setting up the address space for a process

Chapter 14 :: Memory API

This is a basic chapter that shouldn't really be very knew to you. That said, you will not be asked deep questions about calls such as mmap or sbrk.

- Make sure that you fully understand the roll of malloc and free and where they allocate memory
- You should understand the basic roll of the OS calls such as mmap or sbrk, but you are not required to understand how to use them in detail or know the parameters that you pass to them.

Chapter 15: Mechanism - Address Translation

- We spent a lot of time on this content in detail so it's really all on the table.
- Make sure that you understand the relationship between virtual address and physical addresses and in which space the program runs and what happens in hardware generally whenever we are relying on address translation
- Make sure that you understand what roll the hardware plays in address translation.
- You don't need to memorize x86 assembler, but, you may be asked about assembler operations so you should be able to recognize and reason about simple assembler that loads values into memory or into registers.
  - You should know the names of the x86 core registers. I'm not going to quiz you on this specifically, but if you don't know that eax is a general purpose register then you might not be able to understand some questions.

Chapter 16 :: Segmentation

- Make sure that you understand how base and bound/limit work and that segmentation is essentially a generalized base and bounds/limits
- Make sure that you know which aspects of the system are per/process and which are common to all processes
- Make sure that you know the basics of what happens on a context switch
- Review but don't dwell on 16.4-16.6

Chapter 17 :: Free Space Management

- Because of P3, you should be very very comfortable with everything through section 17.3
- Review but don't dwell on 17.4

Chapters 18/19/20 :: Paging

This is a important topic and one that students often have some trouble with the details. You will be asked some questions about the details in this section. I've grouped these chapters together.

- You should be able to reason about the number of virtual pages and the number of physical pages as well as the number of bits in each address given a particular set of parameters.
- Know what is executed in hardware and what is executed in software given RISC or CISC contexts
- Know who is responsible for the code executed in software
- Know the purpose of the TLB
- Be able to compute the hit rate of the TLB given the parameters
- Understand the motivations for each additional mechanism in the system
  - What does the TLB contribute and why do we need it?
  - What's the point of an inverted or a multi-level page table?
- Know what memory MUST be allocated for a given page table structure.
- You will not be tested on combing paging with segmentation.

In short, you should spend sufficient time with these chapters to make sure that you understand the concepts and detail involved. I don't just mean understand at a surface level, you should be able to reason about what benefits and costs a particular configuration provides.

Chapters 21/22 :: Beyond Physical Memory

- I don't expect you to be able to regurgitate the page fault control flow algorithm, but, you should study and understand them and make sure that you know which parts are controlled by software and which parts are controlled by hardware and why.
- Review 21.6 and the concept of the high watermark and low watermark to drive page replacements
- Have a conceptual understand of AMAT with respect to the probability of hits and misses, but there's no need to memorize any specific formula.
- Have a broad understanding of applicability and value of all of the page replacement policies.
- Understand the impact of different workloads.
- Know the basics of implementation and how we approximate LRU
- Be able to explain thrashing.
- Understand the basic workings of the clock algorithm.

Studying for Conceptual Exams: Tips and Strategies

Conceptual exams test your understanding of the principles and ideas behind the subject matter, rather than just your ability to recall facts or execute procedures. Excelling in such exams requires a deep engagement with the material and the ability to apply concepts in varied contexts. Here are strategies to enhance your conceptual learning:

1. **Understand, Don't Memorize**: Focus on understanding the "why" and "how" behind concepts. If you can teach the concept to someone else in your own words, you've likely understood it well.
2. **Create Connections**: Link new information to what you already know. Drawing connections between concepts helps in creating a bigger picture and facilitates deeper understanding.
3. **Utilize Concept Maps**: Visual aids like concept maps or flowcharts can help in visualizing relationships between different ideas and concepts, making it easier to recall and understand them during the exam.
4. **Engage with the Material Actively**: Use active study techniques such as self-quizzing, summarizing sections in your own words, and discussing topics with peers to deepen your understanding.
5. **Practice with Conceptual Questions**: Seek out or create practice questions that test the application of concepts in various scenarios, not just questions that ask for definitions or straightforward procedures.
6. **Reflect on Learning**: After studying a concept, take a moment to reflect on what you've learned and how you might apply this knowledge in real-world or hypothetical scenarios.

Studying for Conceptual Exams with Multiple Choice Questions: Tips and Strategies

This exam may have some MCQ questions and some open format questions. When it comes to conceptual exams containing multiple choice questions (MCQs), it's a common misconception that they will be easier than open-ended questions. However, conceptual MCQs require a deep understanding of the material, as they often test your ability to apply concepts in novel situations or distinguish between closely related ideas. Here are strategies to excel in these exams:

1. **Deep Dive into Concepts**: Ensure you're not just familiar with concepts but understand them profoundly—how they work, why they're used, and in what context they apply.
2. **Practice Critical Thinking**: Engage with materials in ways that challenge you to think critically. This could involve explaining why a concept is true, or under what conditions it might not hold.
3. **Use Active Recall**: Test yourself frequently on key concepts, but go beyond simple recall. Ask yourself how, why, and under what circumstances these concepts apply.
4. **Draw Connections**: Make connections between different topics covered in the course. MCQs often test your ability to synthesize information from various sections.
5. **Practice with MCQs**: Find or create MCQs that challenge your understanding of the material. This will help you get accustomed to the format and the level of detail required.
6. **Read Questions Carefully**: MCQs can be tricky. Pay attention to qualifiers like "always," "never," or "sometimes." The details can significantly alter what the question is asking.

Conceptual vs. Non-Conceptual Multiple Choice Questions: Examples

To demonstrate the difference, let's use examples from operating systems:

Example 1:

- *Conceptual MCQ:* CPU virtualization allows multiple processes to share the same physical CPU in a manner that is transparent to the processes. Which of the following is a primary benefit of CPU virtualization?
  - A) It increases the physical CPU speed.
  - B) It allows a process to use more CPU cycles than physically available.
  - C) It enables more efficient CPU utilization by allowing the OS to allocate CPU time among multiple processes.
  - D) It reduces the need for context switching.
- *Non-Conceptual MCQ:* Which command is used to display the current running processes in a UNIX system?
  - A) ls
  - B) ps
  - C) pwd
  - D) cd

Example 2:

- *Conceptual MCQ:* In the context of memory management, which of the following statements correctly contrasts paging and segmentation?
  - A) Paging allows for variable-sized address spaces, whereas segmentation does not.
  - B) Segmentation eliminates external fragmentation, whereas paging cannot.
  - C) Paging involves dividing memory into fixed-sized units, while segmentation involves variable-sized units.
  - D) Segmentation allows for easier memory sharing than paging.
- *Non-Conceptual MCQ:* Which function is used to request a specific amount of memory from the heap in C?
  - A) malloc()
  - B) free()
  - C) sizeof()
  - D) realloc()

Example 3:

- *Conceptual MCQ:* Considering a system experiencing a high rate of page faults, which of the following strategies could effectively reduce the page fault rate?
  - A) Decreasing the size of pages.
  - B) Increasing the number of frames in physical memory.
  - C) Reducing the level of multiprogramming.
  - D) Implementing a FIFO page replacement algorithm.
- *Non-Conceptual MCQ:* What does a page fault indicate?
  - A) A process is requesting more memory than is available.
  - B) The required page is not in memory, necessitating disk access.
  - C) There is an error in the program code.
  - D) The memory is full and cannot allocate more pages.

Conclusion

Conceptual MCQs can indeed pose a significant challenge, as they test not just your knowledge but your understanding and application of concepts. By studying strategically—focusing on deep understanding, critical thinking, and practice with conceptual MCQs—you can greatly improve your performance on these exams. Remember, the key is not just to know the material but to understand it well enough to apply it in various contexts, as reflected in the nuanced options of conceptual MCQs.

Answers to the Multiple Choice Questions

Here are the correct answers to the previously mentioned multiple choice conceptual questions for reference:

Example 1:

- Correct Answer: C) It enables more efficient CPU utilization by allowing the OS to allocate CPU time among multiple processes.

Example 2:

- Correct Answer: C) Paging involves dividing memory into fixed-sized units, while segmentation involves variable-sized units.

Example 3:

- Correct Answer: B) Increasing the number of frames in physical memory.

These answers underscore the importance of understanding the concepts behind the operations and structures within operating systems to navigate the complexities of conceptual multiple choice exams effectively.