

# Operating Systems and Architecture Comprehensive Exam

August 17, 1999  
Computer Science Department  
University of New Mexico

The test monitor will answer some parts of these questions for you if you are unsure of what some of these terms mean. The only penalty is that you will not get any points for the part that was answered for you. You can still answer the other parts of the question. This can get you started on a question even if you are not sure of the definitions.

Answer any five questions but you must answer at least two of the last three questions (questions 5, 6, and 7). You do not have to do all the parts of a single question. Instead you can pick the parts you want to answer. If you answer only parts of questions you still must answer the equivalent of two of the last three questions. Be **sure** that it is absolutely clear which questions you are answering.

1. **Java OS** (20 points) In the literature they have talked about a “Java operating system”. This is an operating system for a system that only runs Java programs. (Lisp machines also ran only Lisp and had an operating system to support that.) How would a Java OS compare to a regular OS? What parts would it have and what parts would it not have (if any)? What parts would have their function changed?
2. **Hardware abstraction layer** (20 points) A recent approach to operating system portability is the use of a hardware abstraction layer (HAL). Explain what a HAL is and its role in porting the operating system. Define a possible interface for a HAL. Explain your design decisions. Give some alternative you considered and explain why you choose the one you did.
3. **Distributed file systems** (20 points) Consider the path of a file open system call and a file read system call. Follow each of these through the parts of the file and IO systems in a typical operating system. It will go through several levels, getting closer to the hardware at each level and finally ending up as a read of the disk device. There are, at least, the logical and physical levels of the file system. Now assume that the operating system implements a distributed file system and the file is really on another machine in the network. Follow the open and read requests in this situation and show the parts of each system they go through.
4. **Address spaces** (25 points) Most operating systems provide a separate address space for each process but *single-address-space (SAS) operating systems* provide a single virtual address space that is shared by the operating system and all the processes the operating system runs.
  - (a) (3 points) In such a system it would be better to call the units of execution threads rather than processes. Explain why.
  - (b) (8 points) Suppose you were to take a more conventional multiple-address-space operating system and convert it to a SAS operating system. Describe the changes you would have to make to accomplish this.
  - (c) (6 points) What are the advantages and disadvantages of a SAS operating system over a multiple-address-space operating system?
  - (d) (5 points) How would memory protection be handled in a SAS operating system?
  - (e) (3 points) Some recent processors have 64-bit virtual address spaces. Why are

SAS operating systems attractive for such processors?

5. **Consistency and coherence** (20 points) In the context for a shared memory multiprocessor systems, explain the difference between the terms consistency and coherence as these terms relate to the manipulation of data structures.
6. **Memory barrier** (15 points) Most of the newer instruction sets include a memory barrier instruction. What does this instruction do and why is it needed.
7. **Memory gap** (20 points) The growing disparity between processor speeds and memory speeds is perhaps the most significant trend of the past ten years. Both memory and processors are getting faster, so what's the problem? Why can't we make memory as fast as processors (caution, this is a bit of a trick question)? What does the future look like—will this problem go away, or is it likely to become more and more significant? Discuss approaches to addressing this problem.