Operating Systems and Architecture
Comprehensive Exam

January 9, 2001
Computer Science Department
University of New Mexico

Do any five questions. Be **sure** that it is absolutely clear which questions you are answering.

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.

1. **File Systems.** (20 points) Define what it means to "mount" a file system on a directory. Define it from the point of view of the OS user who is doing the mounting (that is, define the effect, not the implementation). What is the purpose of this operation? There are two kinds of mounting, mounting a file system on a local disk and mounting a file system on a remote disk (one available over the network). Describe how each of these types of mounting is implemented. Do this as follows. Describe the two basic operations, open and read. First describe how these are implemented on a normal file (not on a mounted file system and not remote). Describe the steps in the operation, paying particular attention to the OS data structures that are used and modified. Then describe the same operations on a file that is locally mounted. Finally describe these operations on a file that is remotely mounted. These do not have to be separate descriptions; in fact, it is better to combine them to show the similarities and differences.
2. **User code in the kernel.** (20 points) Suppose we would like to allow users to insert user code into the kernel. Why would you want to do this? Give some examples of where it would be useful. What are the problems that can occur if we try to do this? What are some possible solutions to these problems?
3. **OS design**. (20 points) Microkernel designs were once the rage in operating systems. Today, one rarely hears of microkernel designs. Contrast the microkernel approach to other OS design strategies. Describe the factors that lead to the rise of the microkernel approach. Is the current lack of interest in microkernel designs due to widespread acceptance of this approach or were there factors that lead to favoring other approaches? Be certain to support your responses with examples when possible.
4. **System Design**. (20 points) While processors continue to get faster, interrupt latencies (the time it takes to respond to an interrupt) are not improving at nearly the same rate. In many respects, this is similar to the situation in memory systems where memory systems are not improving at the same rate as the processor. Discuss the extent to which these problems are related. You should consider the source of the problem, the degree of effort being directed toward each problem, the extent to which a solution to one problem can be applied to the other, and likelihood that either or both of these problems will continue to be significant problems in the future.
5. **System Design**. (20 points) It is often said that bandwidth is easy, latency is hard.

Examples of this truism can be found in all aspects of system design ranging from instruction interpretation, to memory access, to job scheduling, to communication networks. Describe as many examples of this truism as you can, being certain to explain the circumstances under which it's easy to improve bandwidth.

6. **Architecture**. (20 points) Describe, in general terms, the factors that influence the cache hit rate. Discuss the similarities and differences between the design of a cache system and the design of a demand-paged virtual memory system.
7. **Architecture**. (20 points) In recent years, we have seen the development of computer architectures based on super-scalar designs, super-pipeline designs, and very large instruction words. Give a brief description of each approach. Be certain to discuss the expected benefits and drawbacks of each approach. When appropriate cite specific examples of machines based on each approach.