

Final
May 31, 2007 14.00–16.00
Good Luck!

Solve 8 questions. Each question is 13 points

1. You are supposed to design a highly reliable operating system. What is meant by “reliable”? Which criterias and measures should be taken care?
2. Describe the four main parts of the operating system.
3. Why does the operating system need to provide mechanisms to facilitate process cooperation?
4. On UNIX, which of the following are considered system calls? Describe and compare briefly *system calls* and *library functions*.
 - (a) `read()`
 - (b) `printf()`
 - (c) `malloc()`
 - (d) `open()`
5. What is multiprogramming? Describe and compare single programming and pure multiprogramming.
6. For each of the following system calls, give a condition that causes it to fail: **fork** and **exec**.
7. Describe four ways to *prevent* deadlock by attacking the conditions required for deadlock.
8. Describe internal and external fragmentation and explain the difference between them. Which one occurs in paging systems? Which one occurs in systems using segmentation?
9. What is the difference between a physical address and a virtual address? Assume we have a system that performs dynamic relocation of processes in physical memory. A process that has just started needs 2000 total bytes of memory in its address space. The OS currently has two free regions: the first between physical addresses of 1000 and 2000, and the second between physical addresses 5000 and 7000. What value would end up in the base register for this process, and what value would be in the bounds register?
10. Explain the UNIX index node (inode) structure in detail.