

Operating Systems Qualifying Exam

Spring 2012

Department of Computer Science
New Mexico State University

Note: Please attempt all questions. Justify all your answers. This examination is closed book and notes. You may feel free to use whatever additional reference material you wish, but no electronic aids are allowed. Please note the following instructions:

- show your work whenever appropriate. There can be no partial credit unless you show how you derived your answers
- be succinct. You may lose points for facts that, while true, are not relevant to the question at hand

[1.] The following table gives the arrival times, execution times, and deadlines for some processes in a real-time system. Answer the following questions. [15 pts + 15 pts = **30 pts**].

Process	Arrival	Execution	Deadline
P1	0	16	20
P2	5	20	30
P3	10	5	30
P4	15	5	35

- Identify when each process is going to end if a Round-Robin scheduling without preemption is used with a time quantum of two. You are encouraged to draw the time line to illustrate your answer.
- Does the RR scheduling above meet the deadline of the processes? If not, does there exist a scheduling algorithm that can **assign** the processes priorities which will result in all deadlines being met, assuming a strict priority-based preemptive scheduling algorithm. **Identify** when each process is going to end.

[2.] Assume you have a computer system with four page frames, using an LRU page replacement strategy. Initially, the page frames are all empty. Simulate this page replacement strategy on the following page reference string: 6 7 3 7 9 1 5 9 2 4 2 7 9. [**10 pts**]

[3.] The following questions deal with multithreaded programming. [**15 pts**]

- Differentiate between user level threads and kernel level threads. Can a system be designed to have both user level and kernel level threads?
- Assume that you have a single processor system running the standard Linux kernel, will the use of POSIX threads for a program help in speeding up of the execution of the program through concurrency? Explain.
- In a multiprocessor system, would the result be different? Explain.

[4.] Answer the following general questions:

- Explain how the use of resource allocation graphs can identify deadlocks. When is the resource allocation graph not useful to prevent deadlocks. How do you address the situation? [5 pts]
- Comment on the efficiency of using a *spin lock* to achieve mutual exclusion between two processes in a uniprocessor and a multiprocessor system. Can you improve the efficiency? Explain. [10 pts]
- Why is the page table size chosen to be a power of 2? Can the size be a power of 3? [5 pts]
- Explain what is the requirement that a mutual exclusion primitive has to satisfy to ensure mutual exclusion during code execution. Is this essential requirement provided by the hardware or the OS? [10 pts]

[5.] Some computer system uses a virtual memory scheme with the following characteristics:

Type: Multi-level Page Table

Virtual Memory Address Width: 64 bits

Page Size: 32K

Page Table Entry Size: 8 Bytes

Bits of Physical Frame Number in Page Table Entry: 40

- How large is the virtual memory? [5 pts]
- How wide is a physical address? [10 pts]