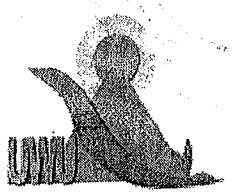

UVA WELLASSA UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE & TECHNOLOGY

END SEMESTER EXAMINATION – SEMESTER I – 2008/2009

CST302-2 OPERATING SYSTEMS

Time Allowed: **TWO HOURS** Answer Four Questions. All questions carry equal marks.

Question 1

Memory management systems and their intricacies can have a strong impact upon system performance. In this question, we explore how memory management should evolve on some interesting hardware configurations. In particular, we focus on hardware systems that support "large" page sizes; assume in this case we have a machine that in addition to a standard 4 KB page allows for large 1 MB pages.

A. Describe the motivation for large pages. Why should hardware systems provide this kind of support?

B. Assume that we want to provide an application interface to allow applications to ask for large pages. How does an OS typically hand out memory, and how should this API be enhanced to include the ability to request small or large pages?

C. Assume that we want to get the benefits of large pages without changing applications or the memory-request interface. Describe how an OS could transparently use large pages, and how it would do so.

Question 2

File systems are often designed with specific workloads in mind. In this question, we explore some traditional file systems, and discuss what their strengths and weaknesses are with regards to various types of workloads.

A. Consider the Berkeley Fast File System (FFS). How does FFS allocate files on disk? What types of workloads work well on FFS?

B. Consider the Log-Structured File System (LFS). How does LFS allocate files on disk? What types of workloads work well on LFS?

C. Now consider both FFS and LFS. Describe a workload that does not work well on either system.

D. Finally, consider NFS, What types of workloads work well on NFS?

Question 3

In this question, you will explore different types of support for multi-threaded applications.

A. You have a multi-threaded application that you would like to port to a new operating system. One common option is to use user-level threads; the other option is to use kernel threads. What are the advantages and disadvantages of user-level versus kernel threads? What are the characteristics of your multi-threaded application?

B. It turns out that you don't have to choose between either type of thread package, because this OS has support for scheduler activations. Unfortunately, there isn't yet a traditional thread library implemented on top of scheduler activations -- this is your job! First, what events must your thread library communicate to the kernel? For each possible event, give a short example of how the event could occur. Second, what events must the kernel communicate to the thread library? For each event, briefly describe how your thread library should react.

Question 4

The classic log-structured file system presents an entirely different way to manage disk storage. Data is never-overwritten in place; instead, all data and meta-data are batched and then appended to the end of an on-disk log.

A. Assume that the log is divided into segments, and that data is batched in a segment and then written to disk. One key parameter is the selection of the segment size. Given a modern disk, how would you select the size of the segment? Be quantitative in your answer.

B. Sometimes, a segment is written to disk before it is full. Describe two different scenarios where that could occur. Given a modern disk, how would smaller segments affect the performance of writes? Be quantitative in your answer.

C. Suppose we modify the LFS, adding mechanisms to enable in-place block update. That is, given a block to write out, the file system can now choose whether to append it to the end of the log, or to overwrite the last live version. Describe how this approach could improve performance.

Question 5

Assume you have a RAID-4 system with D+ 1 disk, with 1 disk used for redundancy information (the rest for data). Each disk has B blocks of capacity, and all disks are of identical make, model, and performance. Unfortunately, one of your data disks fails and will not work anymore, and you must replace it with a brand new disk. The system must then carry out the task of updating the new disk so that the RAID-4 returns to its normal ``working" status (i.e., all the data disks have all their data and the parity disk has all its necessary parity information), a process known as reconstruction.

A. Describe how the process of reconstruction would work for RAID-4. How does the process change if a data disk or parity disk fails?

B. During this reconstruction, how many blocks must be read from the RAID-4 storage system, and how many written?

C. Now assume that we had instead created a storage system with mirroring instead of RAID-4, with a total of 2D disks for the system. Describe how reconstruction would work in the mirrored system.

D. Which is faster - reconstruction in RAID-4 or reconstruction in the mirrored system? Justify your answer.

Question 6

A. What is file metadata?

B. Explain with the aid of a diagram how file metadata is managed in:

(i) The UNIX file-system;

(ii) The FAT32 file-system;

(iii) The NTFS file-system.

C. A researcher suggests using non-volatile flash memory to store the NTFS log file. He believes this will improve performance, reduce power consumption and make the system more resilient to failure. Is he right? Briefly justify your answer in each case.

D. Consider the UNIX file system. Assume that disk blocks are 4 kilobytes in size and each pointer to a disk block requires 8 bytes. What is the largest possible file size using this design? (You don't need to calculate the final numeric answer, just provide an expression for evaluating it and explain how it is derived). Explain and justify any assumptions you make.