CSCI 1670/2670 Homework Assignment 2

Due 11:59pm March 10, 2025

1. [30%] We would like to adapt stride scheduling for use with multithreaded processes. The intent is that each process would purchase a certain number of tickets, as in standard stride scheduling. These tickets would be used to determine the processor time to be allocated to the process as a whole. Within a process, the tickets would be distributed among the process's threads.

   For example, a four-threaded process might purchase 10 tickets and distribute two tickets to each of three threads, and four tickets to the fourth thread.

   Another process, say a two-threaded process, might purchase six tickets, with one going to one of its threads and five to the other.

   In general, an n-threaded process must purchase at least n tickets so that all of its threads can get processor time.

   The scheduler, when deciding which thread to run, will first determine which process to select a thread from based on per-processor meters governed by the number of tickets held by each process. Then, given the process, it will decide which thread within the process to run based on per-thread meters governed by the number of tickets held by each thread within the process. Recall that a meter runs at a speed that is 1/n, where n is the number of tickets held. A process's meter runs only when one of its threads is running; a thread's meter runs only when that thread is running. (See lecture slide XII-14.)

   Assuming the two processes of the example are the only active processes, one will have a meter running at 1/10 the "fair" meter rate, and the other will have a meter running at 1/6 the "fair" rate. When the first (four-threaded) process is chosen to execute (i.e., when its meter has a lower value than that of the other processes), whichever of its threads that has the lowest value on its meter will run. The process's meter will be incremented by 1/10; the thread's meter will be incremented by 1/n, where n is the number of tickets given to that thread. Thread and process meters are initialized to 1/n, where n is the number of tickets assigned to them.

   a. [10%] In normal stride scheduling, the thread with five tickets would run first. Which thread runs first with adapted stride scheduling?

   b. [10%] Among the two processes (again, assuming there are no other active processes) there are 16 tickets outstanding. With normal stride scheduling, with only per-thread meters as explained in lecture, after 16 clock ticks the value of each thread's meter would be increased by 1. Would this per-thread meter increase be different using the adapted stride scheduling described above? Explain.

   c. [10%] Suppose we make a further modification to stride scheduling, allowing each process to distribute any number of tickets to its threads, regardless of the number of tickets it, itself, has purchased. The number of tickets a thread has would be used to determine the fraction of the process's running time given to the thread. Thus, for example, our second (two-threaded) process might distribute one ticket to each of its threads, rather than distributing six tickets to them. After 16 clock ticks, how much actual processor time would each of these threads have run?

2. [30%] Due to market pressure, the manufacturer of the CS167 disk drive has come out with a CS167-II drive that spins 50% faster than the original drive (now called the CS167-I). The specs for the new and old drives are as follows:

	CS167-II	***CS167-I***
Rotation speed	15,000 RPM (4 milliseconds/ revolution)	10,000 RPM (6 milliseconds/ revolution)
Number of surfaces	8	8
Sector size	512	512
Sectors/track	500-1000 (750 average)	500-1000 (750 average)
Tracks/surface	100,000	100,000
Storage capacity	307.2 billion bytes	307.2 billion bytes
Average seek time	4 milliseconds	4 milliseconds
One-track seek time	.2 milliseconds	.2 milliseconds
Maximum seek time	10 milliseconds	10 milliseconds

a. [10%] By what factor is the maximum (one-track) transfer speed of CS167-II faster than that of CS167-I?

b. [10%] Assuming S5FS file systems are on both a new and an old disk drive, how much faster can we expect the average file to be read on the new drive than on the old?

c. [10%] Assuming log-structured file systems are on both a new and an old disk drive, how much faster can 300KB of data be written on the new drive than on the old?

3. [20%] The text, starting on page 242 (lecture slide XV-27), discusses a technique known as soft updates. It mentions on page 245 that FFS using soft updates may require more disk writes than Async FFS (FFS that does all disk updates immediately and doesn't wait for them to complete). Give an example in which more disk writes would occur with soft updates than with Async FFS. Hint: consider a disk block containing two inodes. One refers to a directory. A directory is a special case of a file, containing a sequence of component-name/inode-number pairs.

4. [20%] RAID level 4, in handling a one-block write request, must write to both a data disk and the check disk. However, it need not read data blocks from the other data disks in order to recompute the check block. Explain why not. Assume that the check block is the exclusive-or of the data blocks.