Qualifying Examination - Systems

Processes, Threads, and Synchronization

Problem 1. Describe one important advantage of user-level threads over kernel-level threads. Describe one important advantage of kernel-level threads over user-level threads.

Problem 2. There are two major ways to make a thread or process wait: *busy-waiting* and *blocking*.

a. Define these two terms.

b. On multi-processors, two implementations of semaphores may be available: one that uses busy-waiting, and one that uses blocking. Briefly describe the conditions under which the former are more efficient, and why. Similarly for the latter.

CPU Scheduling

Problem 3. Round-robin (RR) and shortest-job-first (SJF) are widely-discussed CPU-scheduling policies.

a. Describe one way in which RR is preferable to SJF. Note: It is clear that RR has less overhead than SJF, but please assume this difference is negligible.

b. Describe one way in which SJF is preferable to RR.

c. SJF is unimplementable. Why? Briefly describe how an approximation to SJF can be implemented.

Memory Management

Problem 4. This problem is about demand-paged virtual memory.

a. What is the main benefit (or benefits)?

b. Does it make sense to include paging in the design of a computer and OS that are to be used exclusively for batch processing (i.e., no interactive jobs)? Justify your answer.

Problem 5. This problem is about TLBs (also called *associative memory*). Consider a system with two-level page tables. Assume it takes 100 nsec to access physical memory, and 20 nsec to obtain a value from the TLB.

a. Suppose there is no TLB. What is the average time for one memory reference (i.e. the time to read one word from memory given its virtual memory address), assuming all relevant data is in memory (not paged out)?

b. Suppose a TLB is added, and that the hit ratio for the TLB is 0.95. What is the average time for one memory reference, assuming all relevant data is in memory (not paged out)?

c. Does the TLB need to be flushed on context switches? (Hint: It depends...)

Protection and Security

Problem 6. Claim: Protection in the UNIX file system is in some ways similar to access control lists and in some ways similar to capability lists. Do you agree? Justify your answer.

Architecture

Problem 7. Consider a uniprocessor system with a cache size of $K = 2^r$ 32bit words. Assume that a cache line is 8 words and that the time to load a new line on a cache miss is 800 ns. But the read time for one word on a cache hit is 10 ns.

A quick-sort like program is given by the following:

```
void Qsort_like(int N, int a[]){
    if(N<2) return;
    for(int i = 0; i < N; i++)
       if(a[i] > a[N-i-1]) swap(a[i], a[N-i-1]);
    Qsort_like(N/2, &(a[0]));
    Qsort_like(N/2, &(a[N/2]));
  }
```

If our processor is very fast, the execution time for this algorithm will be dominated by the memory reads and writes. if N is a power of 2 and the cache is turned off (i.e. K=0), the execution time for a call to this function will be approximately

$$T(N) = CN * log(N)$$

for some constant C (and the log is base 2.)

Compute an approximate execution time if $K > 0$ in terms of N, K and C.

Problem 8.

- A. Describe the following multi-processor interconnection networks: Mesh, n-dimensional torus, hypercube and fat-tree. Which have been used in real machines? (name them)
- B. Define the bisection bandwidth of an interconnection network. Given a machine with N processors, what is the bisection bandwidth for the networks above? What does the bisection bandwidth of a network tell us about worst case performance? If a computation of size N, when distributed over N processors requires N^2 messages, what is the expected behavior on each of the networks above?

Problem 9. Describe the basic ideas behind directory based coherency in a distributed shared memory multiprocessor.