Systems Qualifying Exam, August 1997
Computer Science Dept., Indiana University, Bloomington

Operating Systems

Problem 1. Consider an OS with a preemptive process scheduler. Describe the steps that occur when a timer interrupt goes off, causing a new process to be scheduled. Include steps related to: interrupt handling, process management, and process scheduling. Assume scheduling is round-robin with a fixed-size quantum.

Problem 2. Consider a system that supports paged virtual memory. Suppose the users start so many processes that the total number of pages actively used by the processes exceeds the number of page frames in the system. Then the page fault rate will be high, and the system will exhibit very poor performance. Describe (at least) one practical technique for dealing with this problem. Telling the users to run fewer processes is not a practical technique! Assume that buying more RAM is also not a practical technique.

Problem 3. Multi-level page tables were invented to solve what problem? Describe how multi-level page tables work.

Problem 4. An important task of a filesystem is keeping track of which blocks contain data for which file and in which order: for example, that data for file *QexamSolns* is contained in the disk blocks with numbers 3285, 3286, and 1688, in that order. (Another important function of a filesystem is access control, but that is ignored in this question. :-)

Describe two reasonable ways to implement this functionality of the file system. For simplicity, you can assume that the directory structure is not hierarchical, so there is a single directory for the entire disk. Describe the advantages and disadvantages of both of your proposed implementations.

Pipelined, VLIW and Super-Scalar Architectures

Instruction-level parallelism can be obtained in several ways. Pipelining allows a series of instructions to execute, each using a different part of a processor (memory bus, ALU, registers, etc.). A superscalar architecture has (or may have) multiple functional units, register sets, etc. so that more than one stream of instructions may be pipelined simultaneously under the control of the CPU, and so that functional units can be dynamically allocated between instruction streams. A VLIW architecture packs several instructions into one, and supports each instruction with a separate CPU; it is a limited form of parallel processor. A VLIW architecture depends heavily on the compiler to extract the parallelism by scheduling (ordinary) instructions into multiple streams, and then combining the instructions in the streams into a single stream of VLIW instructions.

If you believe any assumptions are necessary to answer the question, please state them as part of your response.

```
1. (40%) Comparing Instruction-Level Parallelism.

Given:

   - a pipelined architecture with six stages

   - a superscalar architecture with four adders, four logic units, and two multipliers

   - a VLIW architecture with four pipelined CPUs, each with six stages;

Assume:
```

- that an instruction mix exists to fully utilize each machine (an unrealistic assumption, but chosen to simplify this part of the question)

- that the duration of the clock cycle for all processors is identical, which means that all instruction level parallelism results from performing more operations during the clock cycle, and not from running a processor or a functional unit at a faster clock rate. Note that some functional units may take multiple clock cycles to complete their operation. You should identify these units, and choose a plausible value for the number of clock cycles based on your knowledge of logical and arithmetic operations.

Task:

Please compute the speedup of each architecture compared to an unpipelined uniprocessor and list your results in a table. Name and describe the sources of speedup (if any) that differ between architectures.

2. (40%) Restrictions on Instruction-Level Parallelism

In part (1) the unrealistic assumption was made that all machines are fully utilized.

Name the factors that prevent this assumption from being realized. Be specific; should you not recall the name of a factor, give a clear and CONCISE description of it.

For each factor, there are hardware and/or software techniques that attempt to overcome it. Name or BRIEFLY describe them.

3. (20%) Memory Hierarchy and Instruction-Level Parallelism

So far the focus of the question has been on the processor. However, every processor must be supported by memory, and memory can be organized to take advantage of the principle of locality (the 90-10 rule).

List the elements in a typical memory hierarchy, excluding mass-storage devices (disk drives, tape, etc.).

Locate these two features:

- level 2 cache

- interleaving

in the memory hierarchy.

For each of the architectures in part (1) explain how each feature contributes to a performance increase, if it does.