**Chapter 17 - Exercises**

**17.1 List and briefly define three types of computer system organization.**

**+ Single instruction, single data (SISD) stream:** A single processor executes a single instruction stream to operate on data stored in a single memory.

**+ Single instruction, multiple data (SIMD) stream:** A single machine instruction controls the simultaneous execution of a number of processing elements on a lockstep basis. Each processing element has an associated data memory, so that each instruction is executed on a different set of data by the different processors.

**+ Multiple instruction, multiple data (MIMD) stream:** A set of processors simultaneously execute different instruction sequences on different data sets.

**17.2 What are the chief characteristics of an SMP (symmetric multiprocessor) ?**

+ There are two or more similar processors of comparable capability.

+ These processors share the same main memory and I/O facilities and are interconnected by a bus or other internal connection scheme, such that memory access time is approximately the same for each processor.

+ All processors share access to I/O devices, either through the same channels or through different channels that provide paths to the same device.

+ All processors can perform the same functions (hence the term symmetric).

The system is controlled by an integrated operating system that provides interaction between processors and their programs at the job, task, file, and data element levels.

**17.3 What are some of the potential advantages of an SMP compared with a uniprocessor?**

**Performance:** If the work to be done by a computer can be organized so that some portions of the work can be done in parallel, then a system with multiple processors will yield greater performance than one with a single processor of the same type.

**Availability:** In a symmetric multiprocessor, because all processors can perform the same functions, the failure of a single processor does not halt the machine. Instead, the system can continue to function at reduced performance.

**Incremental growth:** A user can enhance the performance of a system by adding an additional processor.

**Scaling**: Vendors can offer a range of products with different price and performance characteristics based on the number of processors configured in the system.

**17.4 What are some of the key OS design issues for an SMP?**

**Simultaneous concurrent processes**: OS routines need to be reentrant to allow several processors to execute the same IS code simultaneously. With multiple processors executing the same or different parts of the OS, OS tables and management structures must be managed properly to avoid deadlock or invalid operations.

**Scheduling:** Any processor may perform scheduling, so conflicts must be avoided. The scheduler must assign ready processes to available processors.

**Synchronization:** With multiple active processes having potential access to shared address spaces or shared I/O resources, care must be taken to provide effective synchronization. Synchronization is a facility that enforces mutual exclusion and event ordering.

**Memory management:** Memory management on a multiprocessor must deal with all of the issues found on uniprocessor machines. In addition, the operating system needs to exploit the available hardware parallelism, such as multiported memories, to achieve the best performance. The paging mechanisms on different processors must be coordinated to enforce consistency when several processors share a page or segment and to decide on page replacement.

**Reliability and fault tolerance**: The operating system should provide graceful degradation in the face of processor failure. The scheduler and other portions of the operating system must recognize the loss of a processor and restructure management tables accordingly.

**17.5 What is the difference between software and hardware cache coherent schemes?**

**Software Cache Coherent Scheme**

Software cache coherence schemes attempt to avoid the need for additional hardware circuitry and logic by relying on the compiler and operating system to deal with the problem. All the rules for ensuring cache coherence is embedded in a software code.

**Hardware Cache Coherence Scheme**

In hardware schemes, the cache coherence logic is implemented in hardware. The hardware has encoded in it all the needed data localities and caching strategies. There is minimal need for additional software code or instructions as the embedded system logic controller handles that already.

**17.6 What is the meaning of each of the four states in the MESI protocol?**

**Modified:** The line in the cache has been modified (different from main memory) and is available only in this cache.

**Exclusive:** The line in the cache is the same as that in main memory and is not present in any other cache.

**Shared**: The line in the cache is the same as that in main memory and may be present in another cache.

**Invalid:** The line in the cache does not contain valid data.