HW#1 – Jiading Zhou

1. What are the three main purposes of an operating system?

* To provide an environment for executing user programs, facilitating the interaction between hardware and software components​.
* To act as a resource allocator, efficiently managing resources such as memory, CPU, I/O devices, and storage among various programs and users.
* To serve as a control program, ensuring orderly and secure operations by monitoring and managing hardware and software​​.

2. What is the main difficulty that a programmer must overcome in writing

an operating system for a real-time environment?

* The main challenge lies in meeting timing constraints. Real-time operating systems must guarantee that tasks are completed within strict deadlines, which requires precise scheduling and resource management​​.

3. How does the distinction between kernel mode and user mode function

as a rudimentary form of protection (security) system?

* Privilege Separation:
  + Kernel mode has full control over the system, allowing execution of privileged instructions, such as managing I/O devices and memory allocation.
  + User mode restricts access to these operations to prevent unauthorized or accidental system changes.
* Controlled Access:
  + User programs cannot directly access hardware or critical resources. If they need such access, they must request it through system calls, which transition the system into kernel mode temporarily. The kernel validates these requests before executing them, ensuring safety.
* Error Prevention:
  + Any attempt to execute privileged instructions (e.g., hardware management) in user mode triggers a trap, effectively halting the operation and protecting the system from corruption or misuse.
* Switching Modes:
  + The operating system starts in kernel mode, initializes critical components, and then transitions to user mode to run non-privileged applications. Mode switching ensures that user programs are isolated from kernel-level operations.

4. Give two reasons why caches are useful. What problems do they solve?

What problems do they cause? If a cache can be made as large as the

device for which it is caching (for instance, a cache as large as a disk),

why not make it that large and eliminate the device?

* Why caches are useful?
  + Caches speed up data access by storing frequently used data in faster memory​.
  + reduce latency, reduce the need to access slower secondary storage repeatedly, improving overall system performance​.
* Problems they solve:
  + Reduce latency and improve the efficiency of accessing frequently needed data​.
* Problems they cause:
  + Managing caches introduces complexity, such as deciding which data to keep or replace.
  + They increase system cost due to the use of faster, more expensive memory technologies.
* Why not make a cache as large as the device:
  + Caches are expensive and designed for temporary storage, not long-term or large-scale storage. Creating a cache the size of a device would eliminate the economic and efficiency benefits​.

5. What is the purpose of interrupts? How does an interrupt differ from a

trap? Can traps be generated intentionally by a user program? If so, for

what purpose?

* Purpose of interrupts:
  + Interrupts enable asynchronous signals from hardware to notify the CPU of events, preventing the need for constant polling and allowing efficient CPU use​.
* Difference between an interrupt and a trap:
  + Interrupt: Hardware-generated signals for external events like I/O completion.
  + Trap: Software-generated interrupts triggered by user programs to request operating system services or handle errors​​.
* Can traps be generated intentionally?
  + Yes. Traps are used by user programs to request system calls or access privileged operations via the operating system, such as performing I/O tasks​​.