Capítulo 1

Task 11: Show Fault Details

1.1 Task Description

The goal of this task is to display detailed information about page faults. This involves using specific registers and fields in the CPU state to extract and log fault-related details.

1.2 Planned Implementation

1. **Access Fault Address:** Use the `cr2` register to retrieve the address that caused the page fault. This can be done using the `get_cr2()` function.
2. **Retrieve Faulting Instruction:** Extract the `eip` (Instruction Pointer) field from the `state` structure to identify the instruction that caused the fault.
3. **Analyze Fault Details:** Use the `error` field in the `state` structure to determine the nature of the fault:
   - Bit 0: Page not present.
   - Bit 1: Write operation.
   - Bit 2: User-mode access.
   - Bit 3: Reserved bits violation.
   - Bit 4: Instruction fetch.
4. **Log Fault Information:** Print the following details:
   - Faulting address (`cr2`).
   - Faulting instruction (`eip`).
   - Error code and its interpretation.
5. **Test the Implementation:** Trigger a page fault intentionally (e.g., by accessing unmapped memory). Verify that the logged details match the expected behavior.

1.3 Expected Outcome

- The system should log detailed information about page faults, including:
  - The address that caused the fault.
  - The instruction responsible for the fault.
  - The nature of the fault based on the error code.

1.4 Implementation Details

(To be filled after implementation)

1.5 Challenges

(To be filled after implementation)

1.6 Final Outcome

(To be filled after implementation)