

# Operating System

## Quiz : [12]

[CSL301]  
Concurrency & I/O  
Time: 20 min

Instructor: Dr. Dhiman Saha

November 4, 2025



# Q1: Condition Variable 'while' Loop

## Short Descriptive

When a thread wakes up from a `pthread_cond_wait` call, why is it essential to re-check the shared condition (e.g., `while (count == 0)`) inside a `while` loop rather than just proceeding after an `if` check?

## Q2: 'trylock' and Back-off

### Short Descriptive

The `vector-try-wait.c` solution uses `pthread_mutex_trylock()`. If it successfully locks `v1` but `trylock` fails on `v2`, the code immediately unlocks `v1` before retrying.

In above sentences, explain the exact concurrency problem that unlocking `v1` solves. What deadlock condition is this strategy designed to prevent?

## Q3-Q4: I/O Terminology (Fill in the Blanks)

### Question 3

When the OS interacts with a device by repeatedly reading its status register until it is ready, this inefficient technique is known as \_\_\_\_\_.

### Question 4

To avoid the high CPU overhead of Programmed I/O (PIO), a special hardware component called a \_\_\_\_\_ engine can be used to manage the transfer of data between memory and the device.

## Q5-Q6: I/O Terminology (Fill in the Blanks)

### Question 5

A system where device registers are made available at specific memory addresses, allowing the OS to use standard load and store instructions to communicate with them, is using a technique called \_\_\_\_\_.

### Question 6

In a high-load scenario, such as a network server, a "flood" of interrupts can prevent the OS from ever running user-level processes, causing the system to be busy but make no progress. This state is known as \_\_\_\_\_.

## Q7: Polling vs. Interrupts (MCQ)

### Problem Statement

While interrupts allow the OS to overlap computation with I/O, they are not always the best solution. In which scenario is polling considered a **better** choice than interrupts?

- A) When a device is very slow, and the OS needs to run other processes
- B) When the system needs to reduce the CPU overhead of moving data.
- C) When a device is very fast, and the cost of handling an interrupt is higher than the time spent waiting.
- D) When a device needs to transfer large blocks of data directly from memory.

## Q8: Device Driver Purpose (MCQ)

### Problem Statement

What is the primary purpose of a device driver?

- A) To allow user applications to directly control hardware for better performance.
- B) To provide a hardware-specific interface for the CPU's in and out instructions.
- C) To encapsulate device-specific interaction details, providing a standard, abstract interface to the rest of the OS.
- D) To replace the need for a DMA controller by using the CPU for I/O.



**Thank  
You**

