Good question! join() and wait/notify serve different purposes in thread coordination, even though they both involve waiting. Here’s a breakdown of when each is useful:

**1. join()**

* **Purpose**: Used when one thread needs to wait for another specific thread to complete its execution.
* **Usage**: In your example, join() is sufficient because the main thread only needs to wait for myThread and myThread2 to complete before printing the final counter value.
* **When to Use**: Use join() when the task in one thread depends on the completion of another thread, like waiting for a set of parallel tasks to finish before proceeding with further steps.

**2. wait() and notify()**

* **Purpose**: Used for more complex coordination between multiple threads, especially when they need to communicate and take action based on shared resources or conditions.
* **Usage**: wait() causes the current thread to pause until another thread calls notify() or notifyAll() on the same object. This is particularly useful in producer-consumer situations, where threads need to wait until certain conditions are met to proceed (e.g., a producer thread notifying a consumer thread when new data is available).
* **When to Use**: Use wait() and notify() when threads need to communicate, such as coordinating access to shared resources where one thread’s work depends on the state set by another thread (like waiting for a buffer to be non-empty).

**Key Differences:**

* **join() is simpler**: It waits for the completion of a thread and is easy to use when you don’t need any additional coordination.
* **wait/notify is for inter-thread communication**: Useful when multiple threads depend on each other's states, rather than just waiting for one to finish.

**Example Scenario for wait and notify**

Imagine a case with two threads: a Producer that generates data and a Consumer that processes it. The Consumer can wait for data availability using wait(), and the Producer notifies the Consumer when it has produced new data with notify().

**In Summary**

For your current code, join() is indeed sufficient since main only waits for the completion of myThread and myThread2. You would use wait and notify if, for example, myThread needed to pause and resume based on some condition set by myThread2.