1. SketchServer and SketchServerCommunicator:
   * In SketchServer, multiple SketchServerCommunicator threads are added to the comms ArrayList concurrently.
   * However, addCommunicator() and removeCommunicator() methods, which modify the comms ArrayList, are synchronized to ensure thread safety.
   * Without synchronization, concurrent modifications to the ArrayList could lead to issues like ConcurrentModificationException or data corruption.
2. Sketch:
   * In Sketch, methods like contains(), add(), recolor(), delete(), and move() manipulate the idShapes TreeMap.
   * Since these methods can be accessed by multiple threads simultaneously (e.g., from different SketchServerCommunicator threads), they are synchronized to prevent race conditions and ensure data consistency.
   * Without synchronization, simultaneous access to the TreeMap could result in unpredictable behavior, such as incorrect updates or lost data.
3. EchoServer:
   * In EchoServer, each EchoServerCommunicator thread handles communication with a client.
   * While each thread operates independently, there's no shared state or resource that requires synchronization in this simple echo server implementation.
   * However, in a more complex server where shared resources (e.g., a common data structure or file) are accessed and modified by multiple threads, synchronization would be essential to prevent data corruption and ensure thread safety.