Southern New Hampshire University  
CS 350 - Emerging Systems Architecture & Technology  
Mohamed Elhassan

1. **Why does the loop that processes the LED blinking need to run in a separate thread?**

Running the blinking loop in a separate thread lets the program keep doing other things, like checking for button presses or updating the display, without waiting for the entire Morse code to finish. Without this, the program would just freeze up while it's sending the code, which is a problem if you need real-time interactivity.

1. **What is the purpose of returning to the off state after each completed state action?**

Returning to the off state makes sure that the LED is turned off after every dot, dash, or pause, and it avoids things overlapping. It keeps the timing in check, too, because the next action (dot, dash, or pause) needs to happen after the LED has been turned off. It helps the program stay organized and keeps things neat.

1. **How could you integrate serial communications to facilitate changing the messages available to the program?**

You could use the pyserial library to listen for serial input from a device or the console. When it reads something, it could update the activeMessage in the program. For instance, the user could send a new message over serial, and the program would change what it's sending in Morse code based on that.

1. **How could you use the 16x2 display to provide debugging information to the user when they don’t have access to the application console?**

The 16x2 display can show live updates like the current state (dot, dash, etc.), the active message, or timing info. You could even display errors like “Invalid input” or show a simple progress update so users can see what's going on directly on the screen, even if they don’t have console access. It makes things more user-friendly.