1. In the iterative interpreter for Mini-Basic described in the last lecture (and in section 8.1.3 of the text), how are commands represented in the code store? (Choose one.) Answer: Token sequence
   (a) Source text
   (b) Token sequence
   (c) AST

2. Representing Mini-Basic commands as source text in the code store would (a) slow down loading, but speed up analysis within the fetch-analyze interpreter cycle, or (b) speed up loading, but slow down analysis within the fetch-analyze-execute interpreter cycle. (Choose one.) Answer: (b)

3. Here is the code for the Mini-Basic interpreter proper:

   ```java
   public void run() {
        // Initialize...
        CP = 0; status = RUNNING;

        do {
            // Fetch the next instruction...
            ScannedCommand scannedCom = code[CP++];

            // Analyze this instruction...
            Command analyzedCom = parse(scannedCom);

            // Execute this instruction...
            analyzedCom.execute((MiniBasicState) this);
        } while (status == RUNNING);
    }
   ```

   What is CP? Answer: The code pointer