```
START

CLASS KeyboardListener
    # Class to handle keyboard events
    DECLARE keystrokes_list

    FUNCTION __init__
        # Initialize the listener and keystrokes list
        keystrokes_list ← empty list

    FUNCTION start_listener
        # Start listening for keyboard events
        DEFINE listener using pynput.keyboard.Listener
        DEFINE on_press function to append key to keystrokes_list
        BIND on_press function to listener
        START listener

    FUNCTION stop_listener
        # Stop listening for keyboard events
        IF listener is not null THEN
            STOP listener
        END IF
    END CLASS

CLASS MouseListener
    # Class to handle mouse events
    DECLARE clicks_count

    FUNCTION __init__
        # Initialize the listener and clicks count
```

```
        clicks_count ← 0

    FUNCTION on_click (x, y, button, pressed)
        # Function to handle mouse click events
        IF pressed THEN
            INCREMENT clicks_count
        END IF

    FUNCTION start_listener
        # Start listening for mouse events
        DEFINE listener using pynput.mouse.Listener
        DEFINE on_click function to handle mouse clicks
        BIND on_click function to listener
        START listener

    FUNCTION stop_listener
        #  Stop listening for mouse events
        IF listener is not null THEN
            STOP listener
        END IF
    END CLASS

CLASS Logger
    # Class to log data
    DECLARE logger

    FUNCTION __init__
        # Initialize logger
        logger ← null
```

```
    FUNCTION open_connection
        # Open connection
        logger ← python_.Logger()
        logger.connect()

    FUNCTION log_data(data)
        # Log data
        logger.log(data)

    FUNCTION close_connection
        # Close connection
        IF logger is not null THEN
            logger.disconnect()
        END IF
    END CLASS

FUNCTION main
    # Main function to manage program flow
    DISPLAY "Benchmarker"
    DISPLAY "1. Capture Keyboard Events"
    DISPLAY "2. Capture Mouse Events"
    DISPLAY "3. Log Data"
    DISPLAY "4. Exit"
    INPUT choice

    SWITCH choice
        CASE 1:
            DECLARE keyboard_listener ← new KeyboardListener()
            CALL keyboard_listener.start_listener()
        CASE 2:
```

```
            DECLARE mouse_listener ← new MouseListener()
            CALL mouse_listener.start_listener()
        CASE 3:
            DECLARE logger ← new dataLogger()
            CALL logger.open_connection()
            # logging data
            FOR i ← 1 TO 10
                CALL logger.log_data("log Message " + i)
            END FOR
            CALL logger.close_connection()
        CASE 4:
            DISPLAY "Exiting program"
        DEFAULT:
            DISPLAY "Invalid choice"
    END SWITCH
END FUNCTION

CALL main()

END
```