1. Software problem #1 (our best guess)

A. Three threads:

--Hand: sets the collimator/turntable position

--Treat: sets a bunch of other parameters. Part of its job takes eight seconds, during which time it’s ignoring everything else.

--Vtkbp (keyboard handler): invoked when user types. It parses the input, and writes to a two-byte shared variable, "MEOS" (mode/energy offset)

--"Treat" reads top byte, sets current and energy

--"Hand" reads bottom byte, sets the collimator/turntable position

B. Pseudocode:

Vtkbp (gets and parses keyboard input):

```cpp
data_completion_flag = 0

while (1) {
    wait_for_keyboard_activity();
    /* there was some keyboard activity; let's check it */
    if (cursor_in_bottom_right) {
        parse_the_input();
        set the MEOS variable
        set data_completion_flag = 1;
        signal hand thread
        signal treat thread
    } else {
        /* operator still typing */
        data_completion_flag = 0;
    }
    yield();
}
```

Hand (sets the turntable position):

```cpp
while (1) {
    wait until signalled
    read bottom byte of MEOS variable
    /* next line executes quickly */
    set turntable position
    yield();
}
```

Treat (sets a bunch of parameters and delivers treatment):

```cpp
dataent() { /* this is a subroutine that was called */
    while (1) {
        wait until signalled
        read top byte of MEOS variable
        set_energy_and_current();
        set_bending_magnets(); /* this takes eight seconds */
        if (data_completion_flag == 1)
            break;
    }
    /*
    * now we leave the subroutine and progress to a state in
    * which the machine will accept a "beam on" command
    */
    return;
}
```

2. Software problem #2 (simplified)

[Simplifying here and condensing to one thread of control; in reality, the functions below are spread over two different threads, but that is not actually the problem, despite what the paper sometimes says. The problem appears to be given by the following simplified description.]

```cpp
class3 = 0;

while (1) {
    if (in field light position) {
        increment class3;
    } else {
        if (operator pressed set) {
            if (class3 != 0) {
                move turntable out of field light position;
                break;
            }
        }
    }
}
```

What’s the issue here? (Hint: class3 is only one byte.)