Alex Ojemann 109722375

A *one-counter automaton (1CA)* is like a Turing machine but its tape is *read-only* (you can move back and forth but cannot modify the contents), and in lieu of writable memory, it has a single non-negative counter which it may increment, decrement, and test for 0. The input $x$ is given on the tape surrounded by markers, so that the tape contents are $\vdash x \dashv$. Similar to a Turing machine, a 1CA starts in its start state $s$, the read head over $\vdash$, and the counter at 0. At each step, the 1CA can read the current tape symbol and test the counter for 0, and based on this information and the current state, it can increment/decrement the counter, move the head, and enter a new state. (Note: decrementing the counter at 0 keeps the counter at 0.)
Decribe a 1CA which accepts the language $L = \{(a^n b^n)^n : n \geq 1\}$. For example, the first three strings in $L$ are $ab, aabbaabb, aaabbbaaabbbaaabbb \in L$. Your description should be at the same level as the examples in the book (see H8.1).

*Proof.* If the input string is empty, halt and reject immediately, as n must be greater than 1.
Otherwise, read the first letter. If it's not (a), halt and reject immediately. If it is (a), move right and increment the counter up 1. Continue reading (a)s and adding 1 to the counter until the first (b). From there, subtract 1 from the counter for each (b) read and move right. Before each subtraction, test the counter for 0. If it's 0 and there's another (b), halt and reject. If an (a) is read and the counter is not at 0, halt and reject. If an (a) is read and the counter is at 0, repeat this entire process. This will repeat itself until either the machine halts and rejects or it reaches the end of the string.
At this point, the machine will have halted and rejected on all strings except those that start with a sequence of (a)s and are followed by a sequence of (b)s of equal length and have this pattern repeating throughout the string. Now we need to make sure that there are n of these sequences of $a^n b^n$. We start reading (b)s from the right end of the string and moving left without incrementing the counter. Once we read in an (a) we continue to do this. But once we read in the next (b), representing the right end of the next $a^n b^n$ sequence, and we increment the counter up 1. We repeat this process until we reach the left end of the string. Once we do, the counter represents the number of sequences of $a^n b^n$ that are in the string. Now, we read in the first (a) in the string since the tape is at the left end again, subtract 1 from the counter, and move right. Repeat this until we read the first (b). If the counter is at 0 when this occurs, we halt and accept because the number of $a^n b^n$ sequences in the string is equal to the length of the first sequence of (a)s. Otherwise we halt and reject because it's not. $\square$