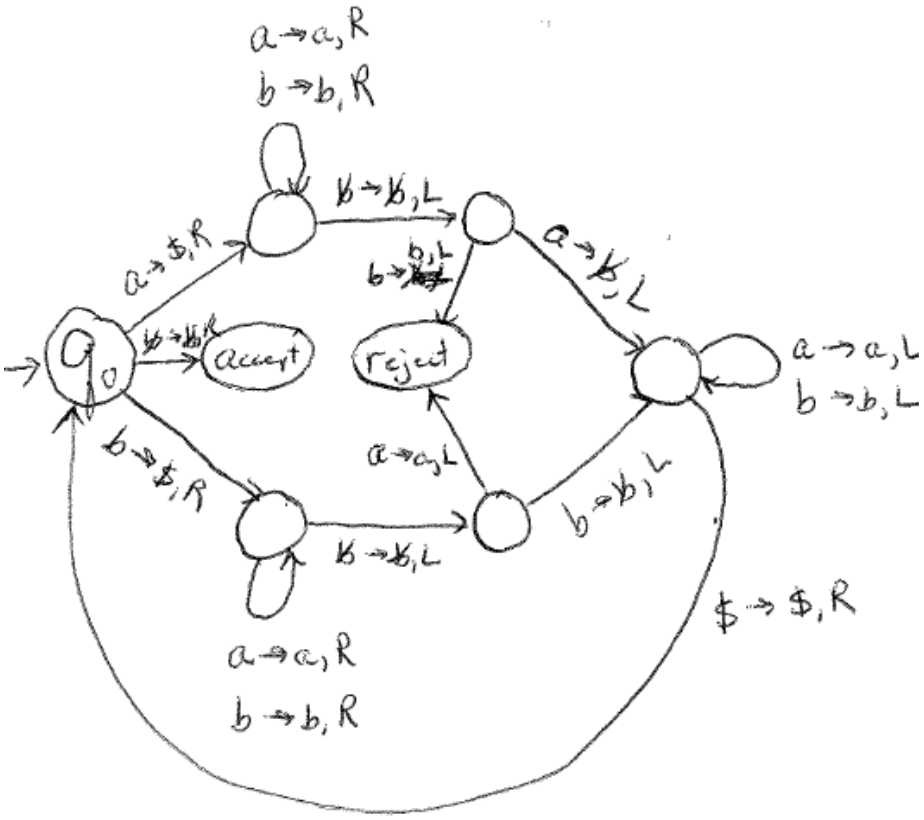
Theory of Computation, CSCI 438 spring 2022
Introduction to Turing Machines, pg. 165-170, March 21st

1. Create a Turing Machine that recognizes $L=\{ww^r \mid w \in \{a,b\}^*\}$

High level plan:
Repeatedly read the beginning character, remember it and go to the end and see if the last character matches the first. Replace character at the beginning with '$', to remember that they are matched, and characters at the end with blank. When none left, accept. If at any point there is not a match, reject.

Detailed plan:
Let _ indicate blank.

loop {
    If '_' accept
    Read and remember the first character, and replace by a '$'.
    Travel right across 'a's and 'b's, until reach a blank. Travel left.
    If character doesn't match remembered character, reject. Otherwise, replace character with a '_' and move left.
    Travel left across 'a's and 'b's, until reach a '$'. Travel right.
}



2. Create a TM which recognizes $\{a^n b^n c^n \mid n \geq 0\}$

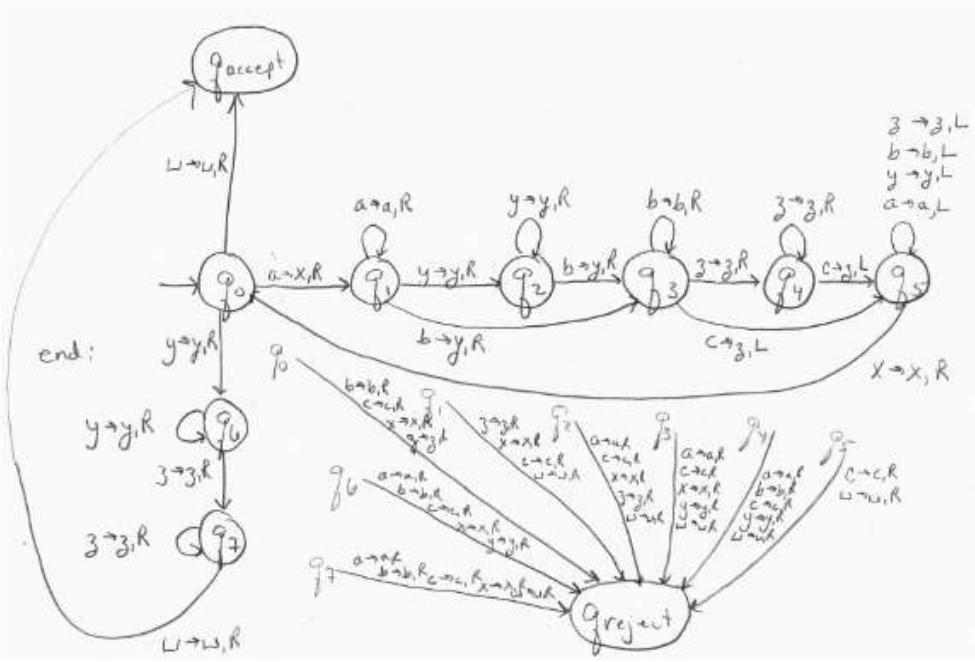
Let _ indicate blank.

High level plan:
Replace first ‘a’ with ‘x’, then replace first ‘b’ with y, then first ‘c’ with z. Repeat, if possible, until all ‘a’s have been replaced/matched. When no more a’s, travel right making sure that there are only y’s followed by only z’s.

Detailed plan:
if ‘_’ accept
loop {
    if ‘y’ go to end
    if ‘a’ replace with ‘x’ and move right; if not ‘a’ reject
    move right over ‘a’s until see a ‘y’ or ‘b’; if see any others, reject
    move right over ‘y’s until see a ‘b’’; if see any others, reject
    replace ‘b’ with a ‘y’ and move right
    move right overs ‘b’s until see a ‘z’ or ‘c’; if see any others, reject
    move right over ‘z’s until see a ‘c’; if see other, reject
    replace ‘c’ with a ‘z’ and move left

    // Successful match of a, b and c. Go back left to look for next a.
    Move left over ‘z’s, ‘b’s, ‘y’s and ‘a’s until see ‘x’. When see ‘x’ move right.
}

end:
move right over ‘y’s until see ‘z’; if any others reject
move right over ‘z’ until see ‘_’’ if any others reject



3. Create a TM for the language $L=\{a^i b^j c^k \mid i+j=k \text{ and } i,j,k \geq 0\}$.
(That is, this will be an addition checker.)

High-level plan:

1. For each ‘a’, replace ‘a’ with ‘x’, go to end of the input and replace a ‘c’ with ‘_’
2. For each ‘b’, replace ‘b’ with ‘x’, go to end of the input and replace a ‘c’ with ‘_’
3. When no more ‘b’, check that the symbol is a ‘_’ and accept

Detailed plan:

Loop while see ‘a’s {
    Replace ‘a’ with ‘x’
    Travel right across ‘a’s
    Travel right across ‘b’s
    Travel right across ‘c’s
    When reach blank, move left and if ‘c’, replace with ‘_’
    Travel left across ‘c’s, ‘b’s and ‘a’s until see ‘x’. Move right on ‘x’
}
Loop while see ‘b’s {
    Replace ‘b’ with ‘x’
    Travel right across ‘b’s
    Travel right across ‘c’s
    When reach blank, move left and if ‘c’, replace with ‘_’
    Travel left across ‘c’s and ‘b’s until see ‘x’. Move right on ‘x’
}
If see ‘_’ accept.