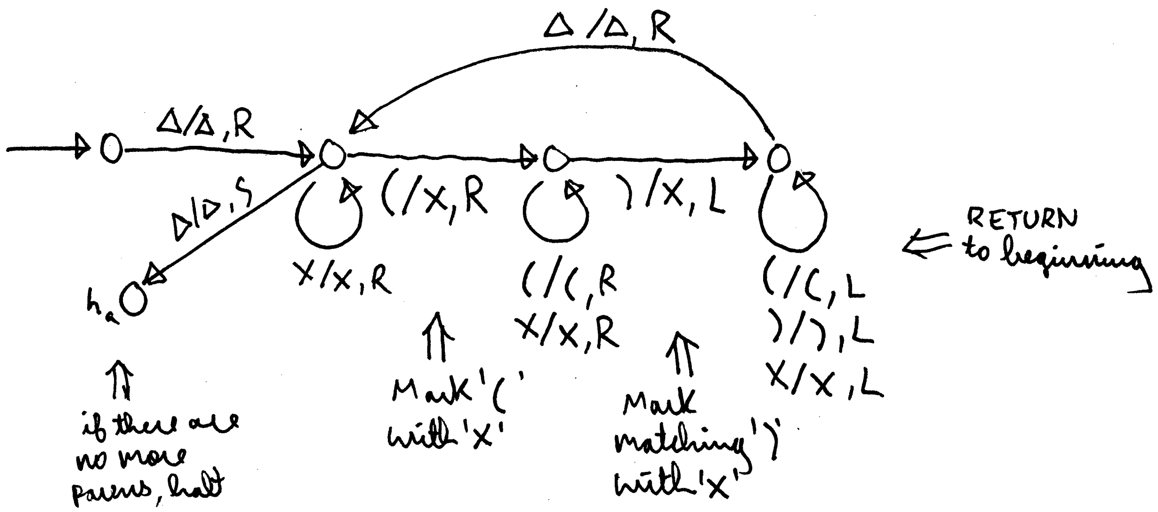
HOMEWORK 1 SOLUTIONS

9.1. $(q_0, \underline{\Delta}aaba) \vdash (q_1, \Delta\underline{a}aba) \vdash (q_2, \Delta A\underline{a}ba) \vdash (q_2, \Delta Aa\underline{b}a) \vdash$
$(q_2, \Delta Aab\underline{a}) \vdash (q_2, \Delta Aaba\underline{\Delta}) \vdash (q_3, \Delta Aaba) \vdash (q_4, \Delta Aa\underline{b}A) \vdash$
$(q_4, \Delta A\underline{a}bA) \vdash (q_4, \Delta\underline{A}abA) \vdash (q_1, \Delta AabA) \vdash (q_2, \Delta AA\underline{b}A) \vdash$
$(q_2, \Delta AAb\underline{A}) \vdash (q_3, \Delta AA\underline{b}A) \vdash (q_4, \Delta A\underline{A}BA) \vdash (q_1, \Delta AA\underline{B}A) \vdash$
$(q_5, \Delta A\underline{A}BA) \vdash (q_5, \Delta\underline{A}aBA) \vdash (q_5, \Delta aaBA) \vdash (q_6, \Delta\underline{a}aBA) \vdash$
$(q_8, \Delta\underline{a}BA) \vdash (q_8, \Delta Aa\underline{B}\Delta) \vdash (h_r, \Delta Aa B\underline{A})$ ∎

9.6d.



Algorithm:
- Mark 1st '(' with 'X' — If there are none halt
- Scan right to matching ')', mark it with 'X'
- Return to beginning of tape

9.9. $L(T_1)=L_1$, $L(T_2)=L_2$, we want to build T s.t. $L(T)=L_1L_2$.

✗ <u>METHOD ONE</u>: Add Λ-transition from final states of T_1 to start state of T_2. This won't work: T_1 and T_2 expect an input string in their respective language, but $w \in L_1L_2$ may be such that $w \notin L_1$ and $w \notin L_2$. We must split w into x,y such that $w=xy$ and $x\in L_1$ and $y \in L_2$.

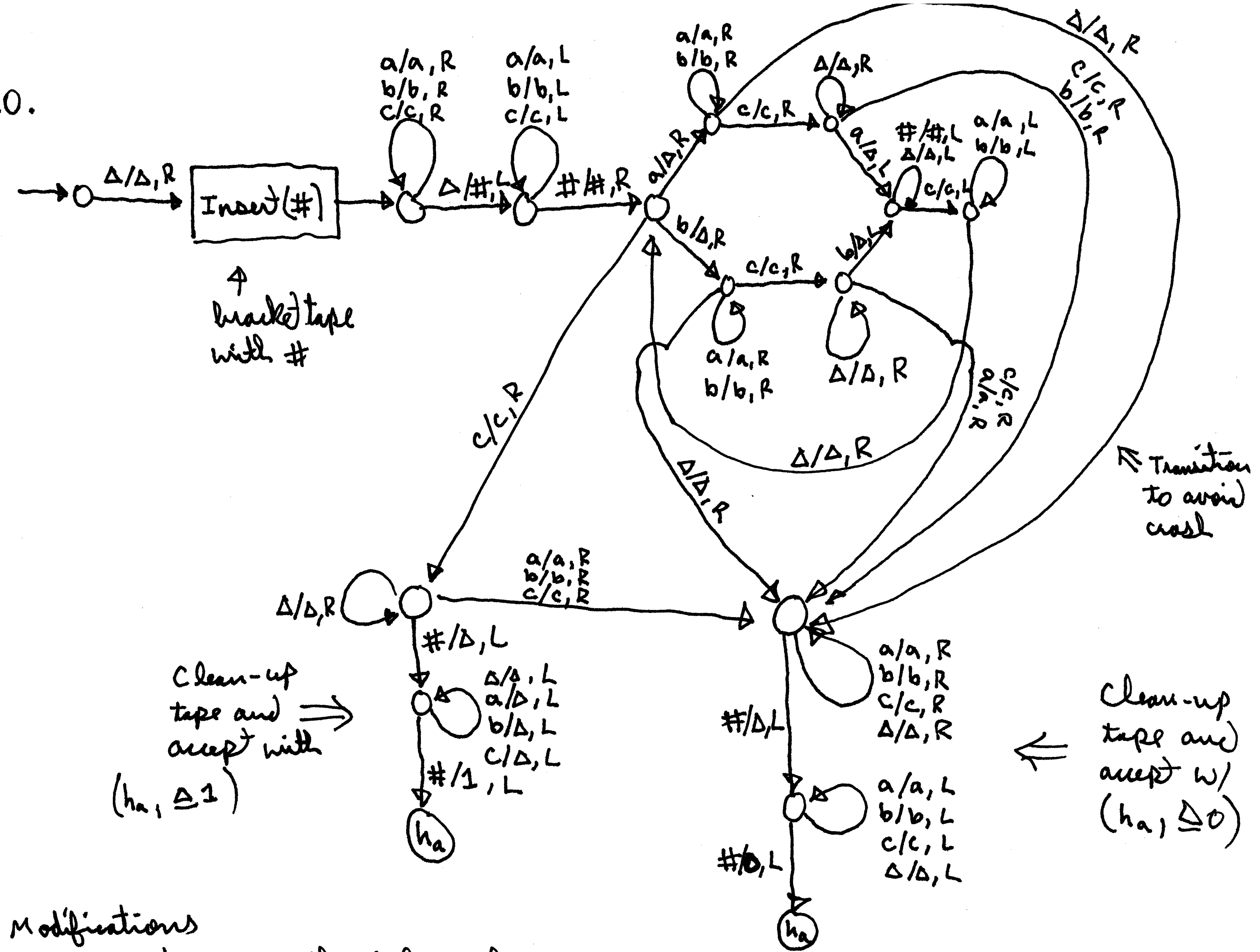
✗ <u>METHOD TWO</u>: Try every possible split $w=xy$ one at a time. This won't work either. It may be that $x\in L_1$ and $y \in L_2$, but some prefix of x causes T_1 to go into an infinite loop.

✓ <u>METHOD THREE</u>: Try every possible split $w=xy$ in parallel. That is, simultaneously simulate T_1 and T_2 on every possible value of x and y, iteratively advancing each simulation by one step. This works, but it is hard to do!

✓ <u>METHOD FOUR</u>: Nondeterministically choose x,y s.t. $w=xy$; run T_1 on x and T_2 on y and accept iff both machines accept. This works.

9.16. $f(x)=a^nb^m$ where n is the number of 'a's in x and m is the number of 'b's.

9.20.

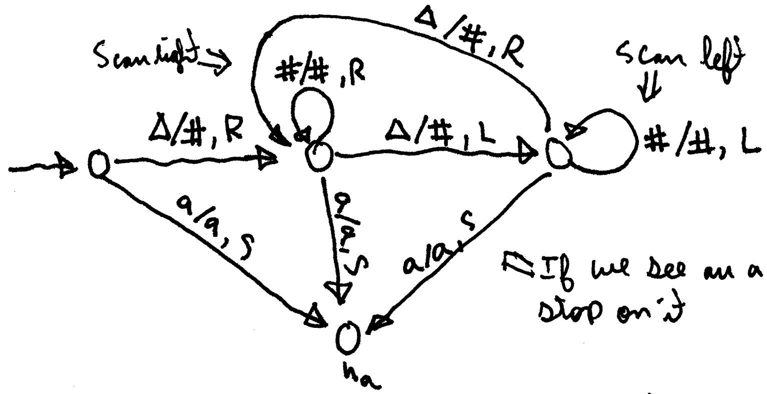


Modifications

1) Bracket tape with # for clean up.
2) Add clean-up and write '1' in place of h_a.
3) Add clean-up and write '0' in place of all clashes.

9.24. We begin in the state $(q_0, x\underline{c}y)$ — i.e. anywhere on the tape. Note that if there are more than one 'a' or no 'a's the behavior is unspecified.

We also assume the input alphabet is $\{a\}$ — they don't specify.



The TM does a scan from left to right over an ever-increasing portion of the tape. If it sees an 'a' it halts and accepts.