3331 Assignment 3

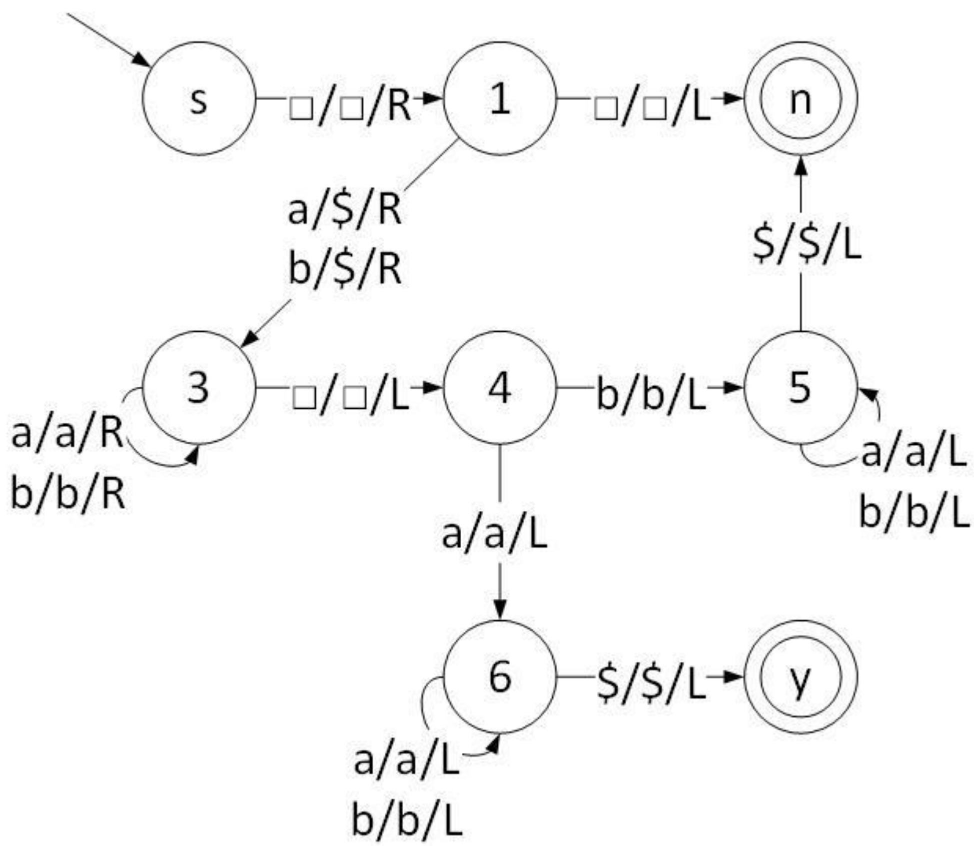
Zaid Albirawi
250626065

1. Construct a deterministic Turing machine M that decides the language

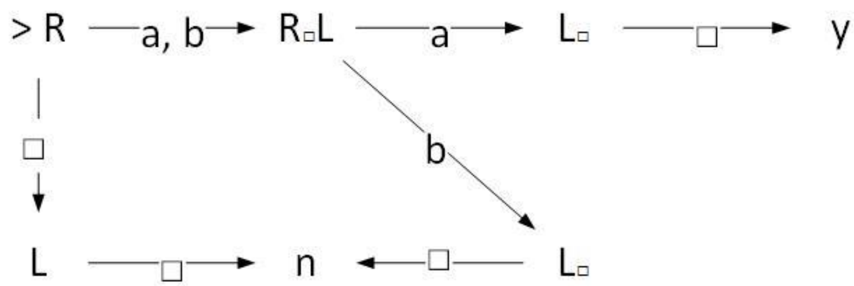
$$L = \{\omega \in \{a, b\}^* | \omega \text{ ends in } a\}.$$

M starts with the initial configuration $(s,\ \underline{\square}\omega)$ and halts with the configuration $(q, \underline{\square}\omega')$, for the appropriate $q \in \{y, n\}$.

   a. Describe M in details using the directed graph whose edges are labelled by transitions (such as the one in Example 17.2, p. 368 of textbook).



b. Describe M using the macro language (such as the one in Example 17.8, p. 376-377 of textbook).



2. The universal Turing machine U, on input $< M, \omega >$, simulates the work of the Turing machine M on input ω. Explain what U does on input $< U, < U, < M, \omega >>>$.

   First, to differentiate between the different universal machines we rewrite the input to $< U_1, < U_2, < M, \omega >>>$. The universal Turing machine will simulate the work of the Turing machine U_1 on the input $< U_2, < M, \omega >>$. While the machine U_2 from that input will also simulate the work of machine M on the input ω. Discussing this in further detail, since a universal Turing machines input must be in the following format, $< M, \omega >$, we denote the input of the universal machine to be $< M'', \omega'' >$. This input is equal to $< U_1, < U_2, < M, \omega >>>$. Furthermore, we do the same thing for the universal machine U_1, let the input of U_1be $< M', \omega' >$ which is equal to $< U_2, < M, \omega >>$. Hence, this is what that operations would look like,

   $< U, < U_1, < U_2, < M, \omega >>>> = < U, < M'', \omega'' >>$
   $< U_1, < U_2, < M, \omega >>> = < U_1, < M', \omega' >>$
   $< U_2, < M, \omega >>$

3. Describe in clear English a Turing machine that semidecides the language

$$L = \{< M > | M \text{ accepts at least two strings}\}.$$

   Let the Turing machine M run the dovetailing algorithm to enumerate all $\omega \in \Sigma^*$ lexicographically. Run the machine until two strings are found. If two strings are found, halt and go to accepting state. Else keep enumerating strings.

4. Prove that the set D of decidable languages is closed under union and concatenation. (Clear English description of the necessary Turing machines is sufficient.)

   Closure under Union: Let M_1 and M_2 be Turing machines that accept the decidable languages L_1

and L_2 respectively. Also, let M be a Turing machine that simulates the work of M_1 and M_2. Therefore, M will be able to decide $\omega \in L_1 \cup L_2$ since both language are decidable. The idea behind this is for the machines M to put the M_1 and M_2 tapes on its tape and work on them simultaneously, if either of the tapes reach a halting state, then halt and accept.

Closure under Concatenation: For this proof we have a string that is a concatenation of two languages, L_1 and L_2. The problem there is that we don't know which part of the string belongs to which language. Therefore, we will need to a universal Turning machine U that will simulate the work of the machines M_1 and M_2 that accept the languages L_1 and L_2 respectively. The idea is that we split ω into the prefix $x \in L_1$ and postfix $y \in L_2$, $\omega = xy$. What we then need to do is $< U, < M_1, x >>$, on all the prefixes of ω until it accepts, if there does not exists a prefix that halts on accepting, then reject. If the prefix x is accepted by U then $< U, < M_2, y >>$, if U accepts, then accept, else, reject and try the next prefix, if you run out of prefixes, reject. Since both languages are decidable, both the M_1 and M_2 machines will have to either accept or reject, which implies that machine U will have to either accept or reject.