

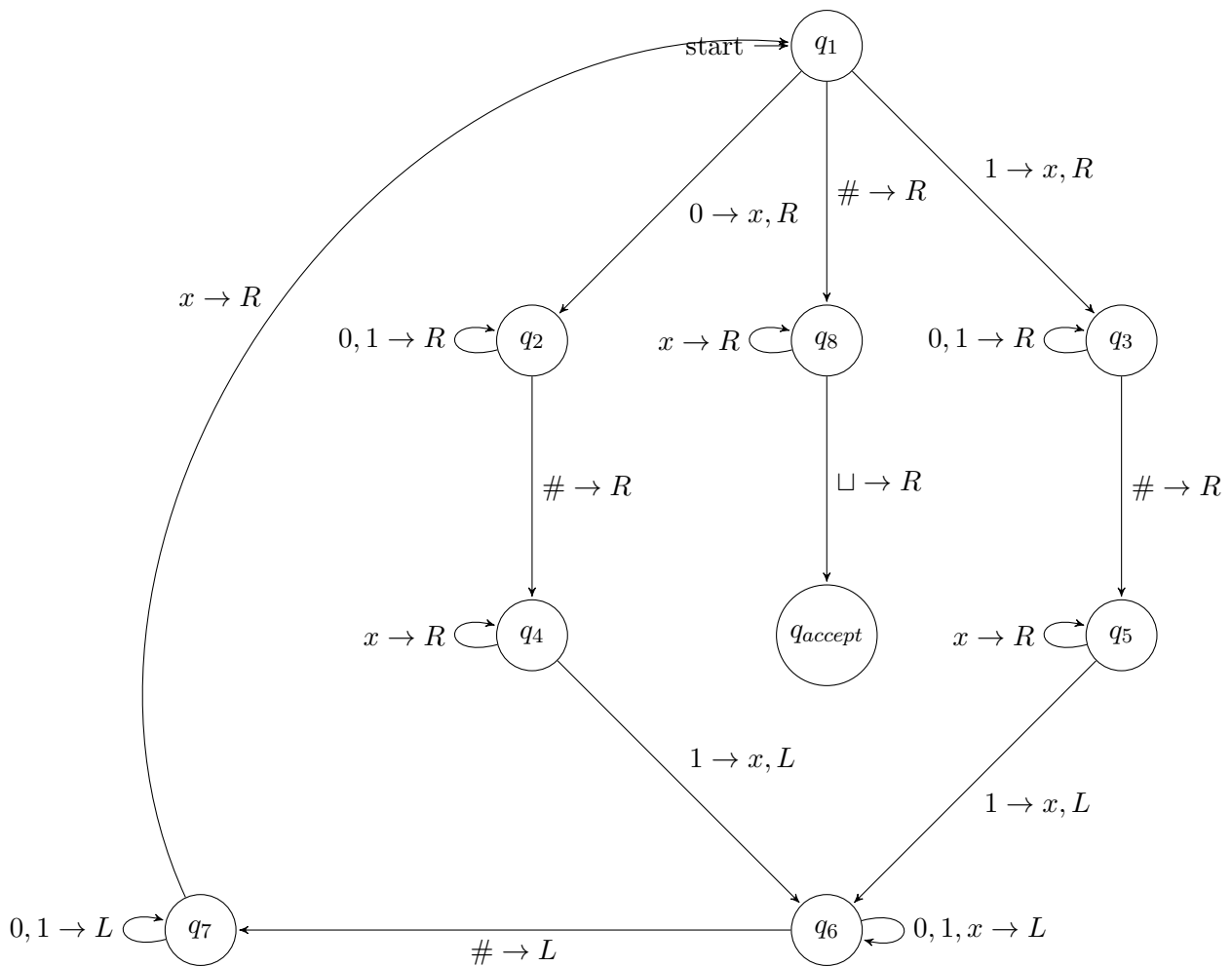
Homework 4 – Due Thursday, March 4, 2021 at 11:59 PM

Reminder Collaboration is permitted, but you must write the solutions *by yourself without assistance*, and be ready to explain them orally to the course staff if asked. You must also identify your collaborators and write “Collaborators: none” if you worked by yourself. Getting solutions from outside sources such as the Web or students not enrolled in the class is strictly forbidden.

Note You may use various generalizations of the Turing machine model we have seen in class, such as TMs with two-way infinite tapes, stay-put, or multiple tapes. If you choose to use such a generalization, state clearly and precisely what model you are using.

Problems There are 4 required problems.

- (Low-Level to Implementation-Level)** The following is the state diagram of a Turing machine using input alphabet $\Sigma = \{0, 1, \#\}$ and tape alphabet $\Gamma = \{0, 1, \#, x, \sqcup\}$.



The notation “ $a \rightarrow R$ ” is shorthand for “ $a \rightarrow a, R$.” The reject state and transitions to the reject state are not shown. Whenever the TM encounters a character for which there is no explicit transition that means that the TM goes to the reject state. Use the convention that the head moves right in each of these transitions to the reject state.

- (a) Give the sequences of configurations that this TM M enters when given as input strings (i) $010\#111$, (ii) $101\#110$, and (iii) $0\#\#0$. Use the same representation for your configurations as we did in lecture 9.
- (b) Give an implementation-level description of the Turing machine described by this state diagram. Hint: The machine is almost identical to Example 3.9 in Sipser.
- (c) What is the language decided by M ?

2. (Implementation-Level to Low-Level)

- (a) Give a state diagram of a TM whose implementation-level description is below. The input alphabet of this TM is $\{a, b\}$ and the tape alphabet is $\{a, b, \sqcup\}$.

Input : String w

1. If the first symbol is blank then *accept*. If it is b then *reject*. If it is a then erase this a (i.e., replace it with a blank), move the head right, and go on to the next step.
 2. Repeatedly move the head right until the blank symbol is found. After it is found move the head one cell to the left (to the last symbol of the string) and go on to the next step.
 3. If this last symbol is not b then *reject*. Otherwise, erase this b , move the head one cell left, and go on to the next step.
 4. If this last symbol is not (another) b then *reject*. Otherwise, erase this b , move the head one cell left, and go on to the next step.
 5. Repeatedly move the head left until the blank symbol is found. After it is found move the head one cell to the right (to the first non-blank symbol of the string) and go back to step 1.
- (b) Give the sequences of configurations that your TM enters when given as input strings (i) $aabbbb$, and (ii) $aabb$.
 - (c) What language is decided by the TM from part (a)?

3. (Programming TMs)

- (a) Let $u \in \{0, \dots, 9\}^8$ be your BU university identification (just the numbers – no leading U or any intermediate dashes). Let s be the sum of the digits of u . This should be a number between 0 and 72. Now convert s to a 7-bit binary number x , with the most significant bit on the left and (most likely) at least one leading 0. Record u , s , and x . The binary number x is your special input that you’ll use below.

Example: If Jenny’s BU UID is U-08-67-5309, she would take $u = 08675309$, $s = 38$, and $x = 0100110$.

Write Turing machines that decide the following languages. That is, the machines should always halt after a finite number of steps on every input, and accept a string w if and only if w is in the given language. Implement your TMs in the following environment: <http://morphett.info/turing/turing.html>. Your solution should contain:

- (i) An implementation-level description of your code.
 - (ii) Code that we can copy from your submission and run directly on that website. (Please add comments and make it as readable as possible.)
 - (iii) The sequences of configurations your TM enters on test inputs 00, 00111, and your special input x . (We will also run your solution on additional test inputs to check its correctness.)
- (b) $L_1 = \{w \in \{0, 1\}^* \mid w \text{ contains an odd number of 1's}\}$.
- (c) $L_2 = \{w \in \{0, 1\}^* \mid \text{there are at least as many 1's in } w \text{ as there are 0's}\}$.

4. (Closure properties)

- (a) Show that the class of **decidable** languages is closed under complement.
- (b) Explain why your construction from part (a) **fails** to show that the **Turing-recognizable** languages are closed under complement. (That is, if L is Turing-recognizable, explain why your construction from part (a) does not necessarily produce a TM recognizing \bar{L} .)

You may use high-level descriptions and multi-tape TMs to solve this problem.