Shuyi Jin sj445
ECE 590: Theory and Practice of Algorithms
October 9, 2024

**Homework 5**

**Question 1.** Give the sequence of configurations that Turing machine M1 enters when started on input string 11011.

The sequence of configurations:

The whole process:

$q_1\ 110\#11$

$x\ q_3\ 10\#11$

$x1\ q_3\ 0\#11$

$x10\ q_3\ \#11$

$x10\#\ q_5\ 11$

$x10\ q_6\ \#x1$

$x1\ q_7\ 0\#x1$

$x\ q_7\ 10\#x1$

$q_7\ x10\#x1$

$x\ q_1\ 10\#x1$

$xx\ q_3\ 0\#x1$

$xx0\ q_3\ \#x1$

$xx0\#\ q_5\ x1$

$xx0\#x\ q_5\ 1$

$xx0\#\ q_6\ xx$

$xx0\ q_6\ \#xx$

$xx\ q_7\ 0\#xx$

$x\ q_7\ x0\#xx$

$xx\ q_1\ 0\#xx$

$xxx\ q_2\ \#xx$

$xxx\#\ q_4\ xx$

$xxx\#x\ q_4\ x$

$xxx\#xx\ q_4$ (lack an outgoing transition)

$xxx\#xx\_\ q_{reject}$.

Figure 1. The answer

2. For the language $A = \{w\#w^R : w \in \{0,1\}^*\}$:

   (a) Describe an algorithm (implementation-level description) for a Turing Machine that decides A.

   (b) Formally define your Turing machine. Describe the transition function with a table or a state diagram.

**Question 2.**

(a) **Algorithm (Implementation-Level Description)**

1. Read the first unmarked symbol of $w$ from the left.
2. Mark the symbol (replace it with $X$ if it's a 0, or $Y$ if it's a 1).
3. Move right to the middle symbol #, skip it, and keep moving right to find the rightmost unmarked symbol (this should be part of $w^R$).
4. Check if the rightmost unmarked symbol matches the one marked in step 2:
   - If it matches, mark the symbol and move back to the leftmost unmarked symbol of $w$.
   - If it doesn't match, reject the string.
5. Repeat steps 1-4 until all symbols in $w$ and $w^R$ are marked.
6. Verify the middle of the string: After marking all symbols, the machine should find the # and ensure no unmarked symbols remain.
7. If all checks pass, accept the string.

(b) Formal Definition of the Turing Machine

Let $M = (Q, \Sigma, \Gamma, \delta, q_1, q_{\text{accept}}, q_{\text{reject}})$, where:

$$Q = \{q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8, q_{\text{accept}}, q_{\text{reject}}\}$$
$$\Sigma = \{0, 1, \#\}$$
$$\Gamma = \{0, 1, \#, X, Y, \square\}$$
$$\delta : Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$$
$$q_1 \text{ is the start state}$$
$$q_{\text{accept}} \text{ is the accept state}$$
$$q_{\text{reject}} \text{ is the reject state}$$

Transition Table

| **State** | **Read Symbol** 0 | **Read Symbol** 1 | **Read Symbol** # | **Read Symbol** $X$ |
|---|---|---|---|---|
| $q_1$ | $(q_2, X, R)$ | $(q_3, X, R)$ | $(q_8, R)$ | – |
| $q_2$ | $(q_2, R)$ | $(q_2, R)$ | $(q_4, R)$ | – |
| $q_3$ | $(q_3, R)$ | $(q_3, R)$ | $(q_5, R)$ | – |
| $q_4$ | $(q_6, X, L)$ | – | $(q_4, R)$ | – |
| $q_5$ | – | $(q_6, X, L)$ | $(q_5, R)$ | – |
| $q_6$ | $(q_6, L)$ | $(q_6, L)$ | $(q_7, L)$ | $(q_6, L)$ |
| $q_7$ | $(q_7, L)$ | $(q_7, L)$ | $(q_1, R)$ | – |
| $q_8$ | $(q_8, R)$ | $(q_8, R)$ | $(q_{\text{accept}}, R)$ | – |
| $q_{\text{accept}}$ | – | – | – | – |

Explanation of Key Transitions:

$q_1$: Start state. Reads the first unmarked symbol (either 0 or 1) from $w$, marks it (as $X$), and moves right.

$q_2$ and $q_3$: These states move to the middle # after marking a symbol in $w$.

$q_4$ and $q_5$: The machine finds and marks the rightmost unmarked symbol in $w^R$, checking if it matches the corresponding symbol in $w$.

$q_6$: The machine moves left across the middle # to return to the unmarked part of $w$.

$q_7$: The machine moves left to find the next unmarked symbol of $w$.

$q_8$: After all symbols are marked, the machine checks for the end of the string. If everything is valid, it moves to the accept state.

(We assume that any undefined transitions implicitly lead to the reject state $q_{\text{reject}}$.)