CSC 339 – Theory of Computation
Fall 2023

Tutorial 7
Turing Machines

Exercise 1

Give a low-level and full description of a Turing machine for the language of strings of length at least two that end in 00. The alphabet is $\Sigma = \{0,1\}$.

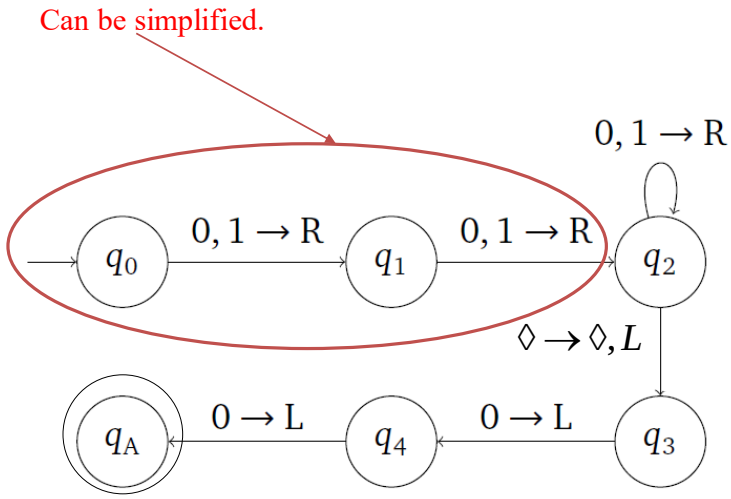
Solution

1. If the first symbol is blank, reject. (Because the input is empty.)
2. Move right.
3. If the current symbol is blank, reject. (Because the input has length 1.)
4. Move right until a blank is found.
5. Move left. (The head is now over the last symbol.)

Solution

6. If the current symbol is not a 0, reject.
7. Move left. (The head is now over the second-to-last symbol.)
8. If the current symbol is not a 0, reject.
9. Accept.

Solution



5

Exercise 2

Design a Turing machine that recognizes the language $\{a^n b^n c^n | n \geq 0\}$. The alphabet is $\Sigma = \{a, b, c\}$.

Solution

- Detailed description of the Turing machine:

1. If the first symbol is blank, accept. (Because the input is empty.)
2. If the first symbol is not an *a*, reject. (Because the input has more *b*'s or *c*'s than *a*'s.)
3. Replace the *a* by an *x*.
4. Move right, skipping *a*'s and *y*'s, until a *b* is found. Replace that *b* with a *y*. If no *b* was found, reject.

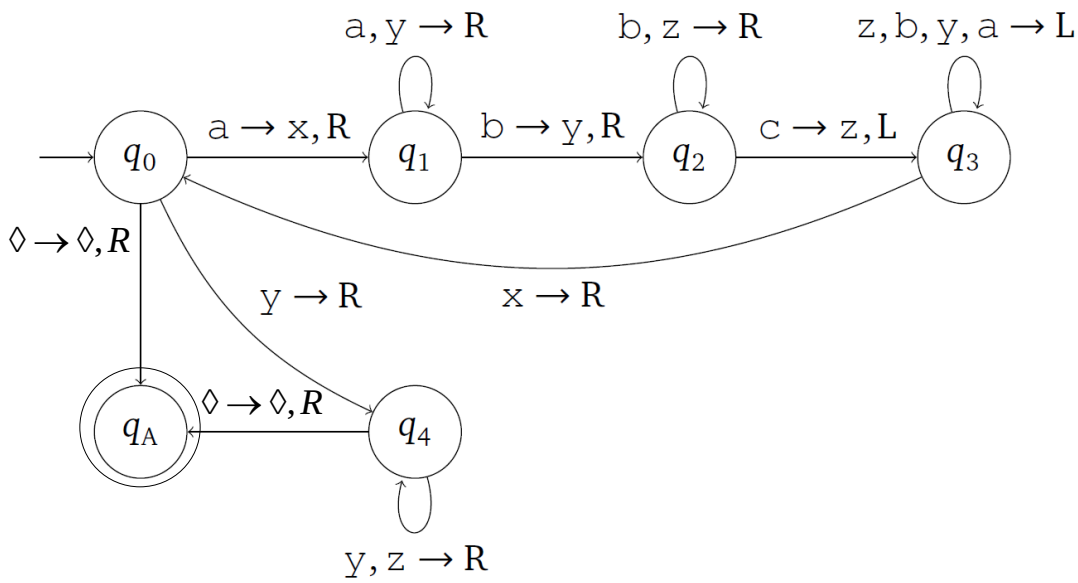
7

Solution

5. Move right, skipping *b*'s and *z*'s, until a *c* is found. Replace that *c* with a *z*. If no *c* was found, reject.
6. Move left until an *x* is found. Move right. (The current symbol is the first *a* that hasn't been crossed off. Or a *y*.)
7. Repeat Steps 3 to 6 as long as the current symbol is an *a*.
8. When that symbol is a *y* instead, scan the rest of the memory to verify that all *b*'s and *c*'s have been crossed off. This can be done by moving right, skipping *y*'s and *z*'s until a blank symbol is found. If other symbols are found before the first blank, reject. Otherwise, accept..

8

Solution



9

Exercise 3

Give a low-level and full description of a TM for the language $\{w\#w | w \in \{0,1\}^*\}$. The alphabet is $\Sigma = \{0,1,\#\}$.

Solution

The basic idea is to move back and forth between the strings on either side of the # sign to verify that their symbols match. Symbols that have been matched need to be crossed off so we know they have been dealt with.

Solution

1. If the first symbol is blank, reject. (Because the input is empty.)
2. If the first symbol is a # sign, move right. If that symbol is blank, accept. Otherwise, reject.
3. Replace the current symbol with an *x*.
4. Move right, skipping over 0's and 1's, until a # sign is found. If none is found, reject.

12

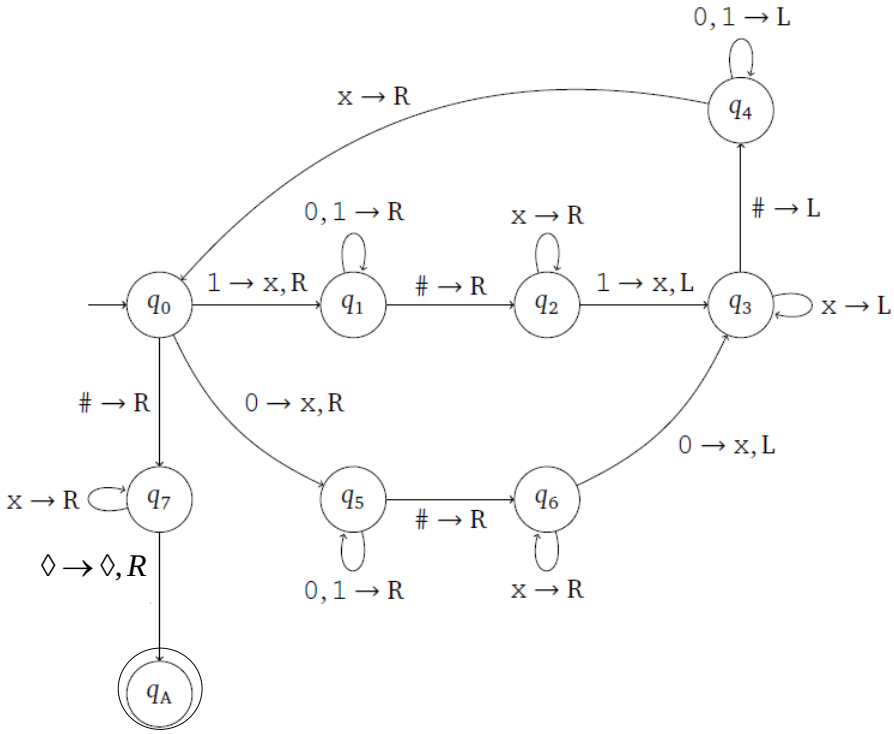
Solution

5. Move right, skipping over *x*'s, until a 0 or a 1 is found. If none is found, reject.
6. If the current symbol is not identical to the one remembered in Step 3, reject. Otherwise, replace that symbol with an *x*.
7. Move left to the # sign.
8. Move left until an x is found. Move right. (The current symbol is the first symbol that hasn't been crossed off on the left side of the # sign. Or the # sign itself.)

Solution

9. Repeat Steps 3 to 8 until the current symbol is a # sign.
10. Scan the rest of the memory to ensure that all symbols to the right of the # sign have been crossed off. This can be done by moving right, skipping over *x*'s, until a blank is found. If other symbols are found before the first blank, reject. Otherwise, accept.

Solution


0, 1 → L
x → R
q4
0, 1 → R
x → R
→ L
1 → x, R
→ R
1 → x, L
x → L
q0
q1
q2
q3
→ R
0 → x, R
0 → x, L
x → R
q7
q5
q6
→ R
◊ → ◊, R
0, 1 → R
x → R
qA

15

Exercise 4

Give a low-level and full description of a TM for the language $\{ww | w \in \{0,1\}^*\}$. The alphabet is $\Sigma = \{0,1\}$.

Exercise 5

Give a low-level and full description of a TM for the language $\{0^{2n}1^n | n \geq 0\}$. The alphabet is $\Sigma = \{0,1\}$.

Exercise 6

Give a low-level and full description of a TM for the language $\{a^n b^m a^{n+m} | n, m > 0\}$. The alphabet is $\Sigma = \{a, b\}$.

Exercise 7

Give a low-level and full description of a TM for the language $\{a^i b^j a^{i*j} | i, j > 0\}$. The alphabet is $\Sigma = \{a, b\}$.