Int 201: Decision Computation and Language
Tutorial 9 Solution

Dr. Chunchuan Lyu

November 19, 2025

Question 1. Show step by step processing of input string 01#01 on the Turing Machine M in Figure1, using configuration descriptions. i.e., start with $q_1$01#01. Note that $x \rightarrow R$ means, read x and move right without writing.

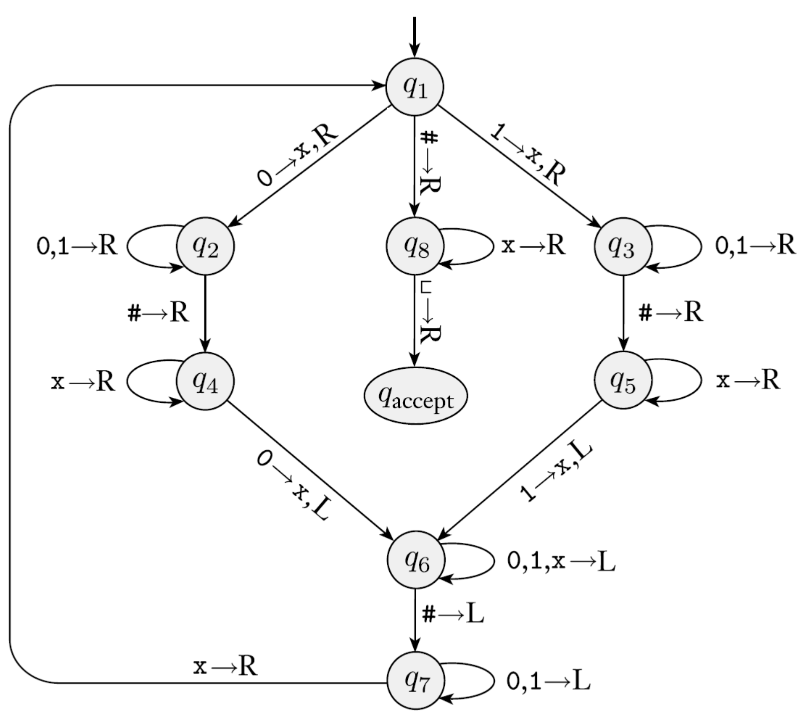

Figure 1: Turing Machine M

Solution 1. 1. $q_1$01#01

2. $xq_2 1\#01$
3. $x1q_2\#01$
4. $x1\#q_4 01$
5. $x1q_6\#x1$
6. $xq_7 1\#x1$
7. $q_7 x1\#x1$
8. $xq_1 1\#x1$
9. $xxq_3\#x1$
10. $xx\#q_5 x1$
11. $xx\#xq_5 1$
12. $xx\#q_6 xx$
13. $xxq_6\#xx$
14. $xq_7 x\#xx$
15. $xxq_1\#xx$
16. $xx\#q_8 xx$
17. $xx\#xq_8 x$
18. $xx\#xxq_8$
19. $xx\#xxq_{accept}$

Question 2. Write down the set description of the language being accepted by the Turing Machine M in Figure1. Justify your answer.

Solution 2. $L = \{w\#w | w \in \{0,1\}^*\}$, the machine cross over 0s or 1s pair before and after the # sign, and accept the input if all strings before # matches strings after # and all inputs are exhausted.

Question 3. Is the Turing Machine M in Figure1 a decider (i.e., all inputs halt on M)? Justify your answer.

Solution 3. Yes, this TM is a decider, as the zigzag procedure always terminates for finite length input string. In fact, the machine rejects whenever there is a mismatch, therefore for any length $2n+1$ (if length not odd, it rejects sooner than the time spend on accepting $2n+3$), the longest running step happen when the string gets accepted. Total time for accepting $2n+1$ length-ed string is the sum of all the zigzag plus final checking all inputs being exhausted. So, it is $O(n^2)$.

Question 4. Are all context-free languages Turing-decidable? Justify your answer.

Solution 4. Yes, as running CYK algorithm takes finite time. A TM can simulate the running of CYK, and TM accepts the string iff CYK produce a parse. Therefore, all CFL are Turing-decidable. Alternatively, this is a corollary from lecture 10, where we built a more powerful device that handles all grammars and inputs at once.

Question 5.

$N =$ On input w

1. Check if $w \in a^*b^*c^*$, reject if not.
2. Count the number of a's,b's, and c's
3. Accept if all counts are equal; reject if not

Write down the set description of the language being accepted by the Turing Machine N of the above high-level descriptions.

Solution 5. $L = \{a^k b^k c^k | k \geq 0\}$

Question 6. A queue automaton is like a push-down automaton except that the stack is replaced by a queue. Formally, see Introduction to Queue Automata.

Explain in high-level description how to simulate a TM with a queue automaton and vice versa. [1]

Solution 6. The key insight is to use a marker in the queue to signify the beginning of tape in TM.

See for a full proof

[1] As a corollary, a two-stack PDA can simulate a queue automaton, and therefore a two-stack PDA is equivalent to a Turing machine.