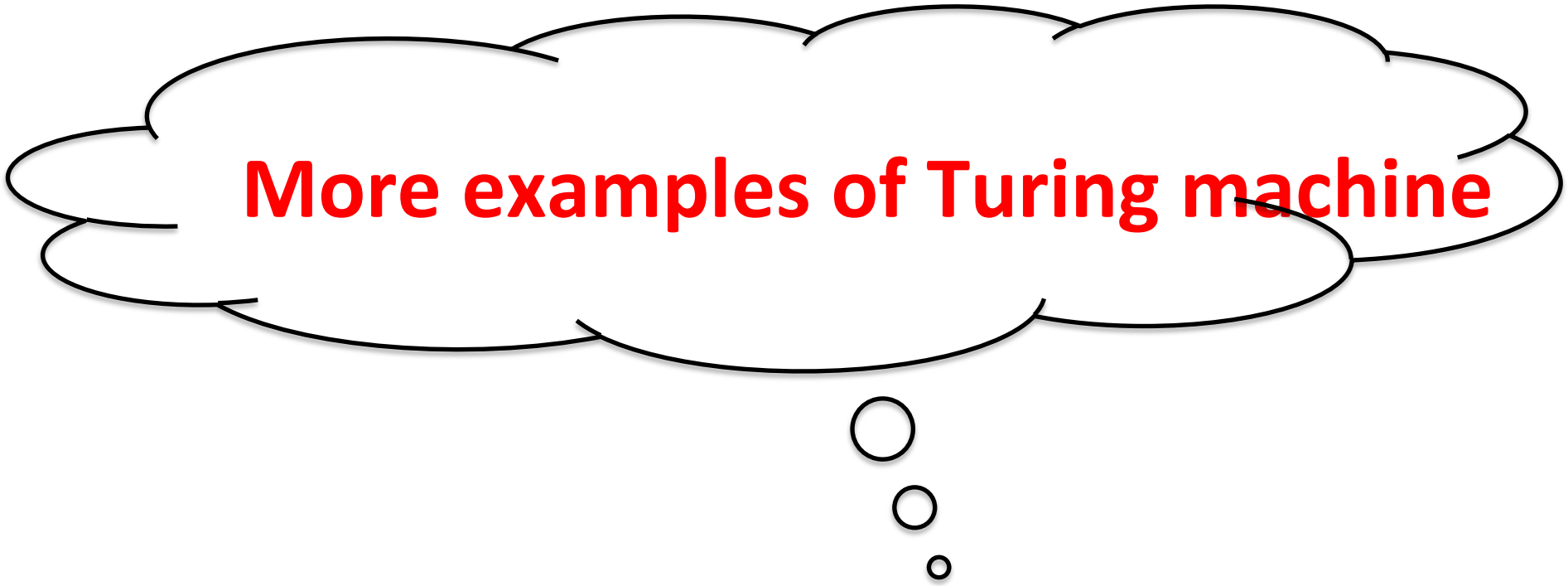
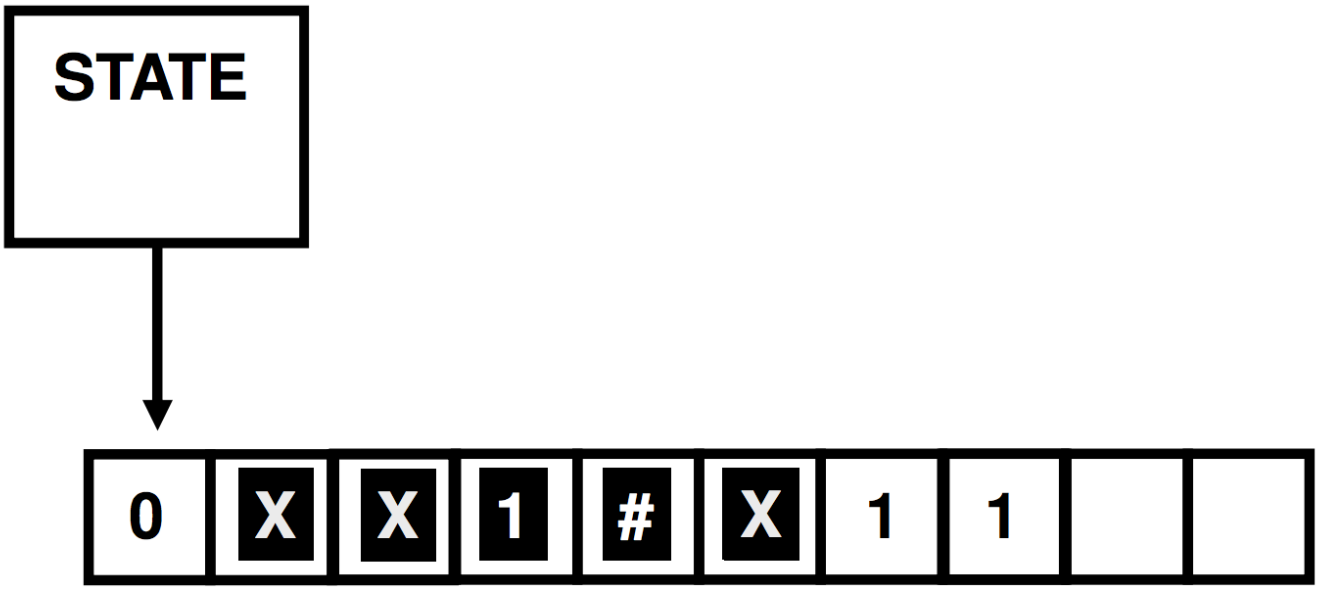

More examples of Turing machine

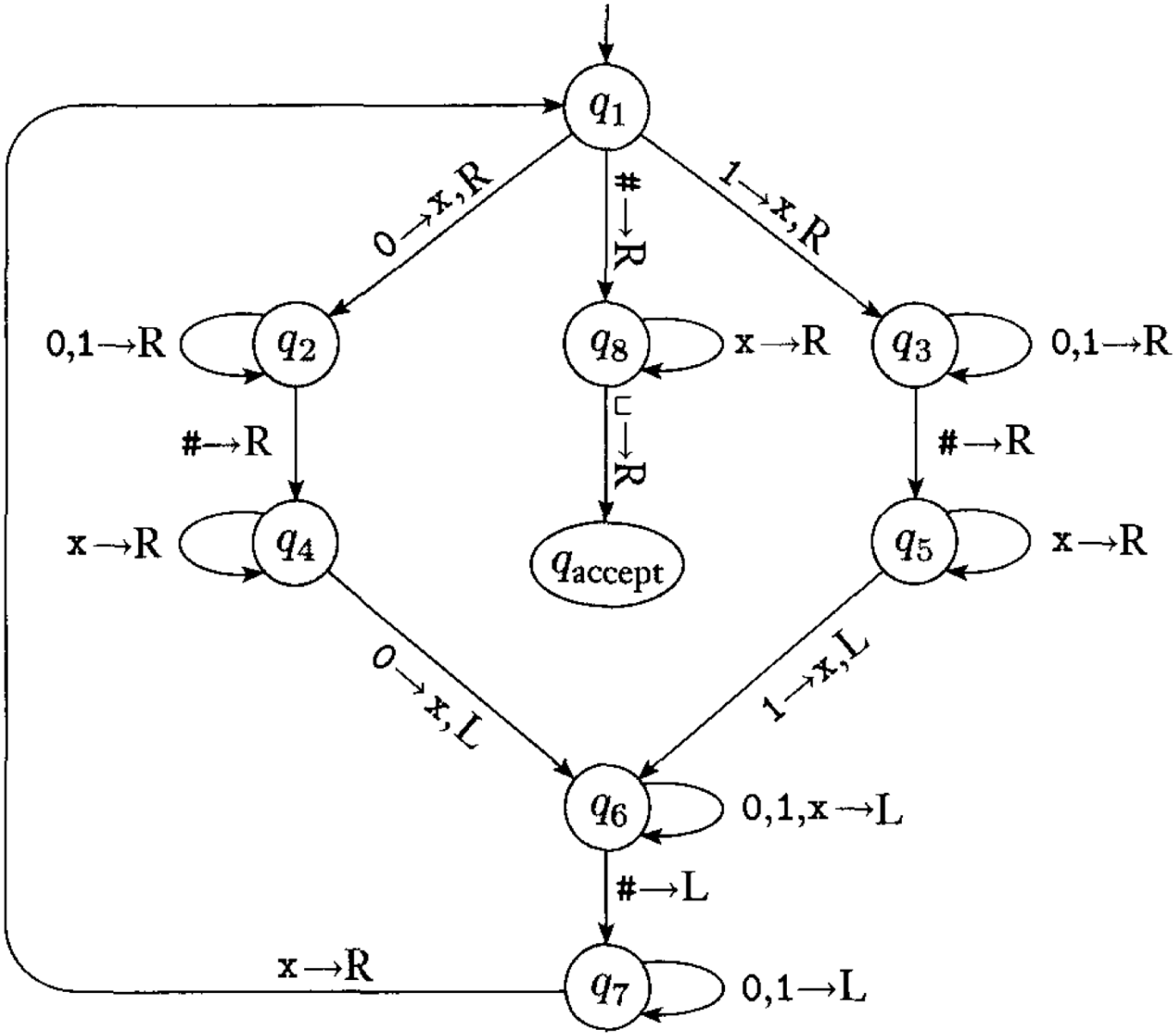
An interesting example (algorithm)

Deciding the language L = { w#w | w ∈ {0,1}* }

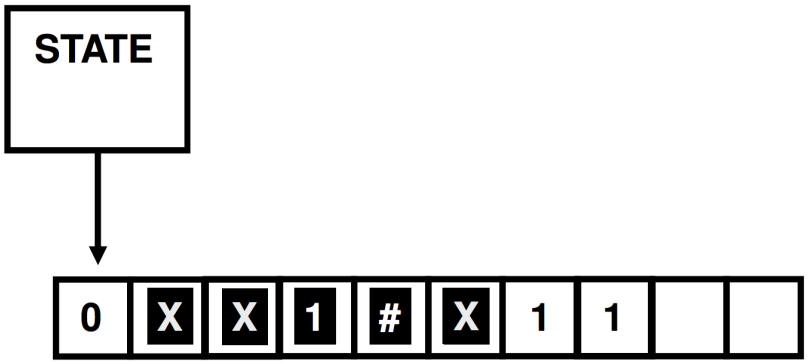


1. If there's no # on the tape (or more than one #), *reject.*
2. While there is a bit to the left of #,
   Replace the first bit with X, and check if the first bit b to the right of the # is identical. (If not, *reject*.)
   Replace that bit b with an X too.
3. If there's a bit to the right of #, then *reject* else *accept*

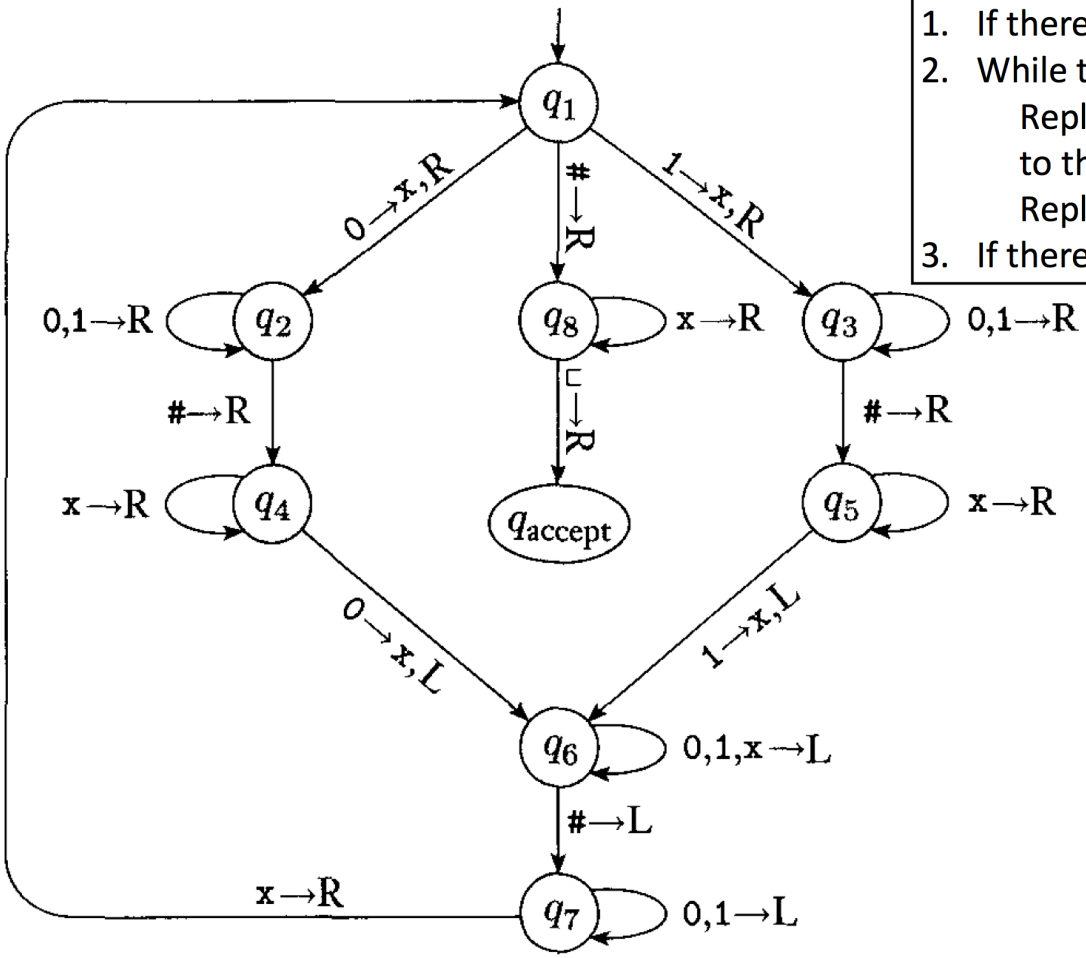
The corresponding Turing machine



Deciding the language L = { w#w | w ∈ {0,1}* }



1. If there's no # on the tape (or more than one #), *reject.*
2. While there is a bit to the left of #,
   Replace the first bit with X, and check if the first bit b to the right of the # is identical. (If not, *reject*.)
   Replace that bit b with an X too.
3. If there's a bit to the right of #, then *reject* else *accept*



More examples

Design a Turing machine which decides these languages.

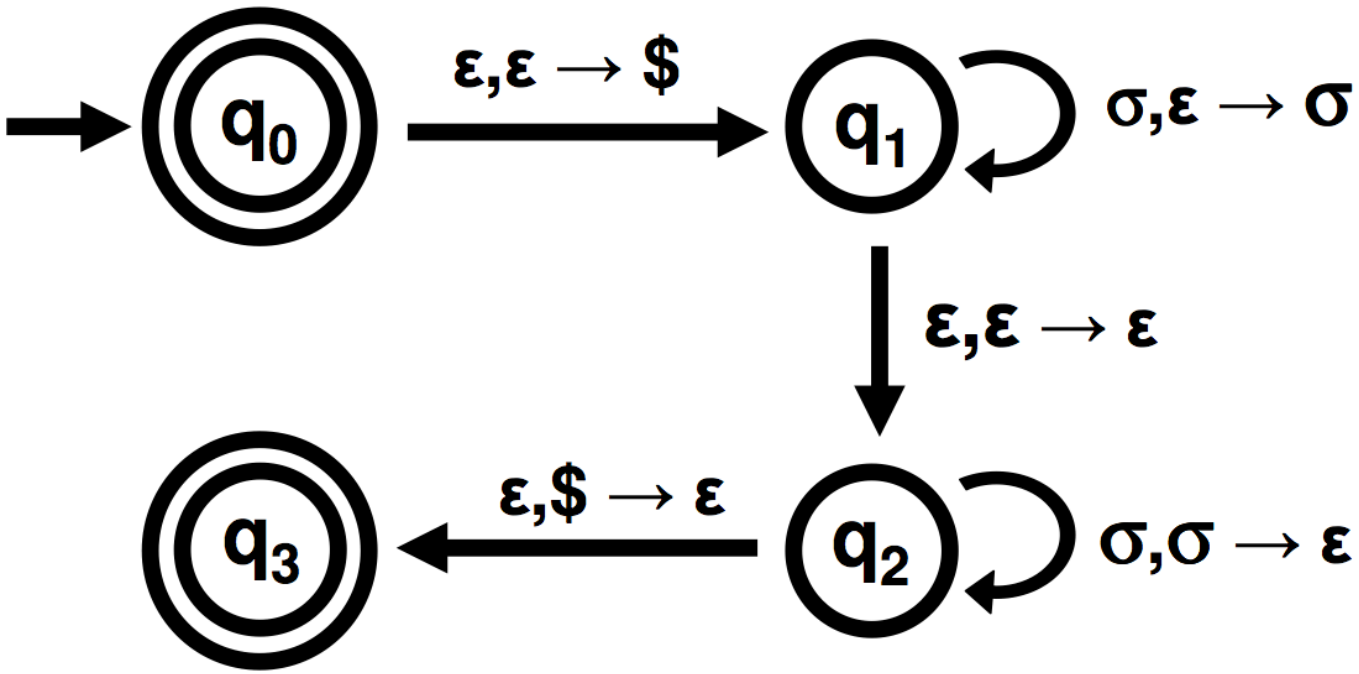
$$B = \{\mathtt{a}^n\mathtt{b}^n\mathtt{c}^n \mid n \geq 0\}$$

C = { w | w has equal number of 1s and 0s}

Palindromes can be easily recognized by a PDA

EVEN-LENGTH PALINDROMES

$\Sigma = \{a, b, c, \ldots, z\}$, $\sigma \in \Sigma$



Hilbert and his 10th Problem

- In 1900: Posed 23 "challenge problems" in Mathematics
- The 10th problem:

  Devise an algorithm to decide

  if a given polynomial has an integral root.
- We now know: This is undecidable!
  - Needed a definition of "algorithm", which was given by Church and Turing (independently)