We will give a Turing machine for the language

$$\mathcal{L}_S = \{\ v\#w \mid v, w \in \{a, b\}^* \text{ and } v \text{ is a substring of } w\ \}.$$

Before giving the Turing machine, try to answer the question: Is it context-free?

Design of the Turing Machine

Since v has to occur somewhere in w, we can write

$$\mathcal{L}_S = \{\ v\#w_1\, v\, w_2 \mid v, w_1, w_2 \in \{a,b\}^*\ \}.$$

We build the TM in two parts: The first part replaces w_1 by #-symbols. Since we do not know in advance where the second occurrence of v starts, this part is non-deterministic.

If the replacement was done well, one obtains a word of form $v\,\#^{i+1}\,v\,w_2$, where i is the length that w_1 had.

The second part recognizes words of form

$$\{\ v\,\#^{i+1}\,v\,w_2 \mid v, w_2 \in \{a,b\}^*\ \}.$$

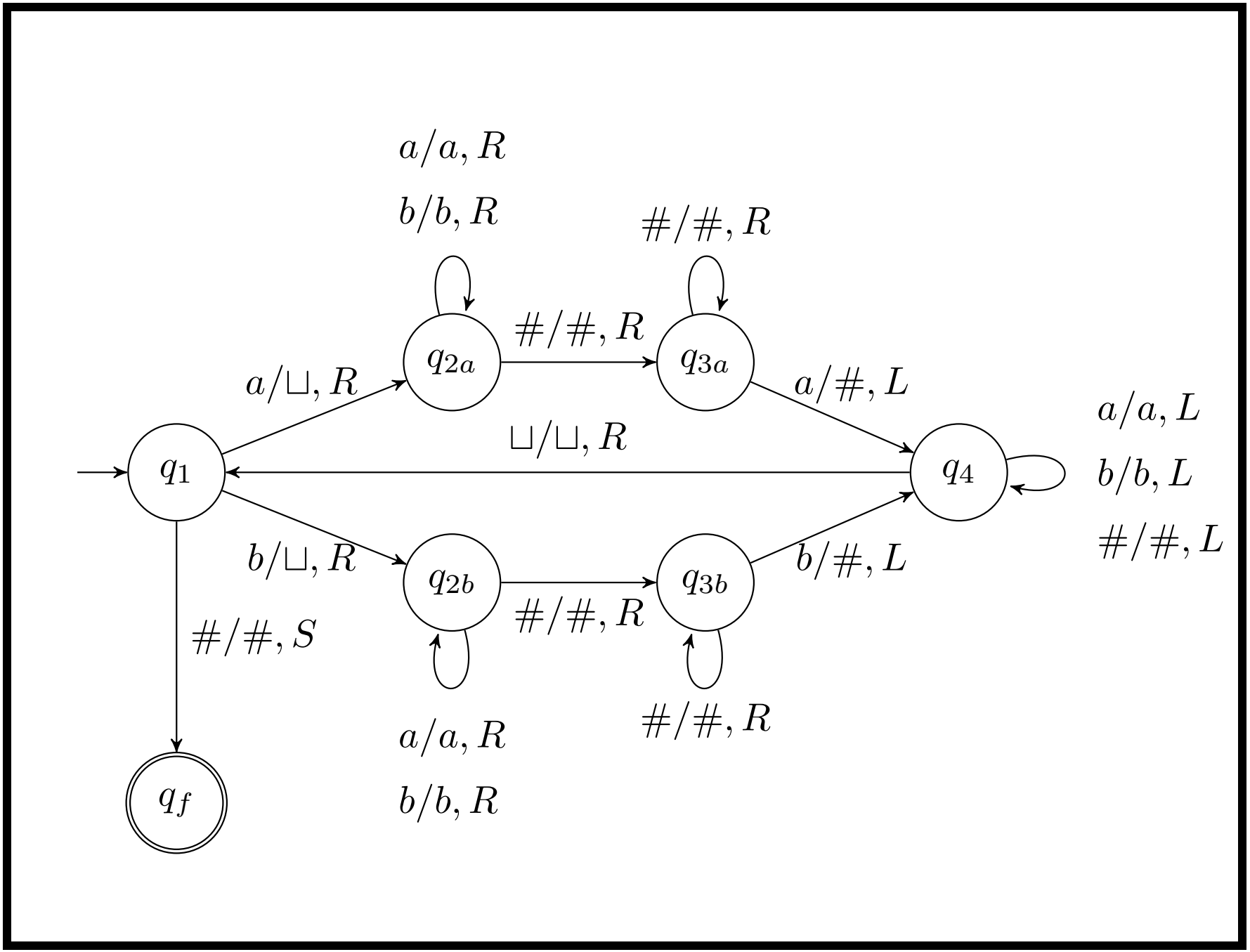
The TM for the second part is deterministic. We will give it on the next slide.

Turing Machine for $\{ v\#^j vw_2 \mid v, w_2 \in \{a,b\}^*,\ j > 0 \}$

This TM works in the same way as the TM in Example 3.9 in Sipser.

In state q_0 it looks at the first symbol of the word.

- If it is #, it accepts.
- If it is a, the TM replaces it by $\sqcup$ and moves to state q_{2a}. In this state, it moves the tape head to the first symbol after the #-s. If this symbol is an a, the TM replaces it by # and moves back to the beginning of the word (in state q_4).
- If the first character is b, the TM does the same but it goes through the states q_{2b}, q_{3b}.


$a/a, R$
$b/b, R$
$\#/\#, R$
$\#/\#, R$
q_{2a}
q_{3a}
$a/\sqcup, R$
$a/\#, L$
$a/a, L$
$\sqcup/\sqcup, R$
q_1
q_4
$b/b, L$
$\#/\#, L$
$b/\sqcup, R$
q_{2b}
q_{3b}
$b/\#, L$
$\#/\#, R$
$\#/\#, S$
$\#/\#, R$
$a/a, R$
$b/b, R$
q_f

Turing Machine that non-deterministically replaces $v\#w_1\,v\,w_2$ by $v\#^{i+1}w_2$

We use capital Q instead of small q, so that we can distinguish the states from the states of the previous Turing machine. In the final transition, it connects to the starting state q_1 of the previous TM.

The header of this slide is a bit misleading. The TM starts in Q_1 and looks for the # symbol. After that, it nondeterministically replaces zero or more a and b-s by #. When it feels that it has done enough, it moves to state Q_3, which moves back the tape head to the beginning. Once it sees a $\sqcup$ on the tape head, it continues in state q_1 of the TM on slide 4.

If the starting word had form $v\#w_1\,v\,w_2$ it is possible to obtain a word of form $v\#^i vw_2$ with $i > 0$, which will be accepted by the other automaton. Otherwise, it is not possible.

TM that replaces a few a-s and b-s by $\#$