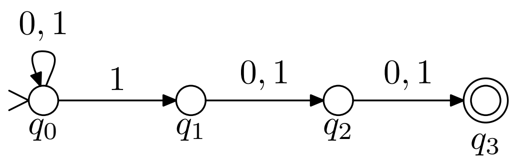
Solutions to Supplementary Problems

Problem S3.1 Design a nondeterministic finite automaton that tests whether in a given binary input sequence the third-to-last bit is a 1. Make the automaton deterministic using the subset construction.

Solution. The language is recognised by a nondeterministic automaton $M = (Q, \Sigma, \delta, q_0, F)$, where

$$\begin{aligned} Q &= \{q_0, q_1, q_2, q_3\} \\ \Sigma &= \{0, 1\} \\ F &= \{q_3\}, \end{aligned}$$

and the transition function δ is defined as in the following diagram:

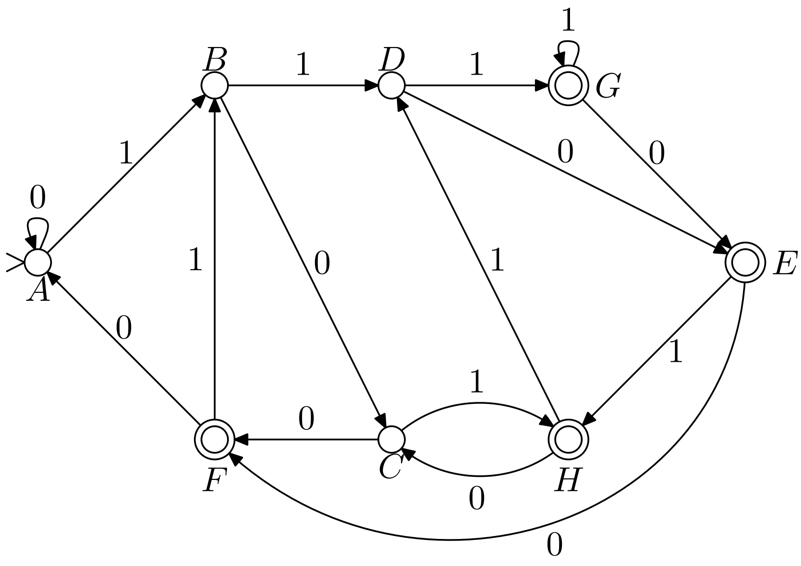


We construct a corresponding deterministic automaton $\widehat{M}$ by taking all possible subsets of Q as its states ($\widehat{Q} = \mathcal{P}(Q)$). Each subset corresponds to a possible "superposition" of alternate states that may emerge in a computation of M. For example, when M has read the input 010 it can be either in state q_0 or in q_2. The automaton $\widehat{M}$ will then reach the superposition state $\{q_0, q_2\}$ on the same input.

We construct the transition function $\hat{\delta}$ for $\widehat{M}$ according to the following table:

$\hat{q}$	0	1	new name
$\{q_0\}$	$\{q_0\}$	$\{q_0, q_1\}$	$\rightarrow A$
$\{q_0, q_1\}$	$\{q_0, q_2\}$	$\{q_0, q_1, q_2\}$	B
$\{q_0, q_2\}$	$\{q_0, q_3\}$	$\{q_0, q_1, q_3\}$	C
$\{q_0, q_1, q_2\}$	$\{q_0, q_2, q_3\}$	$\{q_0, q_1, q_2, q_3\}$	D
$\{q_0, q_2, q_3\}$	$\{q_0, q_3\}$	$\{q_0, q_1, q_3\}$	$E \rightarrow$
$\{q_0, q_3\}$	$\{q_0\}$	$\{q_0, q_1\}$	$F \rightarrow$
$\{q_0, q_1, q_2, q_3\}$	$\{q_0, q_2, q_3\}$	$\{q_0, q_1, q_2, q_3\}$	$G \rightarrow$
$\{q_0, q_1, q_3\}$	$\{q_0, q_2\}$	$\{q_0, q_1, q_2\}$	$H \rightarrow$

The initial state of $\widehat{M}$ is $A = \{q_0\}$, where q_0 is the initial state of M, and the accepting final states of $\widehat{M}$ are all those superposition states in $\widehat{Q}$ that contain some original accepting state in M, i.e. in this case q_3. In the table, the initial state A of $\widehat{M}$ is marked with an incoming arrow and the final states E, F, G, H with outgoing arrows.



Problem S3.2 Show that if a language $L \subseteq \{a, b\}^*$ is recognised by some finite automaton, then so is the language $L^R = \{w^R \mid w \in L\}$. (The notation w^R means the reverse of string w, that is, the string where the symbols of w are in reverse order.)

Solution. Let $M = (Q, \Sigma, \delta, q_0, F)$ be a finite automaton that recognises L (that is, $L = \mathcal{L}(M)$). We use it to form an ε-automaton $\widehat{M}$:

$$\begin{aligned}\widehat{M} &= (Q \cup \{\hat{q}_0\}, \Sigma, \hat{\delta}, \hat{q}_0, \{q_0\}) \\ \hat{\delta} &= \{(q_i, a, q_j) \mid \delta(q_j, a) = q_i\} \\ &\qquad\qquad \cup \{(\hat{q}_0, \varepsilon, q_i) \mid q_i \in F\},\end{aligned}$$

where $\hat{q}_0 \notin Q$.

Intuitively this definition means that we get an automaton for L^R by reversing all the transitions of M and adding a new initial state that has ε-transitions to all the final states of M. The new automaton has only one accepting final state, which corresponds to the initial state of M.

The automaton $\widehat{M}$ starts its computation on an input w^R by nondeterministically moving to one of the states corresponding to the accepting states of M. After that it follows the transitions of M in reverse order. The input w^R is accepted if this leads to the initial state of M, as this means (by reversing the transition path) that M would have an accepting run on the input w.

The diagram below presents an example of this construction: