Problem

Let A be any language. Define *DROP-OUT*(A) to be the language containing all strings that can be obtained by removing one symbol from a string in A. Thus,

$$\textit{DROP-OUT}(A) = \{xz \mid xyz \in A \text{ where } x, z \in \Sigma^*, y \in \Sigma\}.$$

Show that the class of regular languages is closed under the *DROP-OUT* operation. Give both a proof by picture and a more formal proof by construction as in Theorem 1.47.

THEOREM **1.47**

The class of regular languages is closed under the concatenation operation.

Step-by-step solution

Step 1 of 4

Given that

A is any language and

$DROP\_OUT(A) = \{xz \mid xyz \in A \text{ where } x, z \in \Sigma^*, y \in \Sigma\}$

We have to prove that class of regular languages closed under $\mathrm{DROP\_OUT}$ operation.

i.e. if *A* is a regular language then $DROP\_OUT(A)$ is also regular.

We have to take that *A* is regular and we have to prove that $DROP\_OUT(A)$ is regular.

Since *A* is a regular language, it must be recognized by a DFA.

Let $M = (Q, \Sigma, \delta, q_0, F)$ be the DFA recognizes *A*.

Now we will construct an NFA $N = (Q', \Sigma \cup \{\epsilon\}, \delta', q_0', F')$ that we recognize $DROP\_OUT(A)$.

There are two copies of Machine *M.*

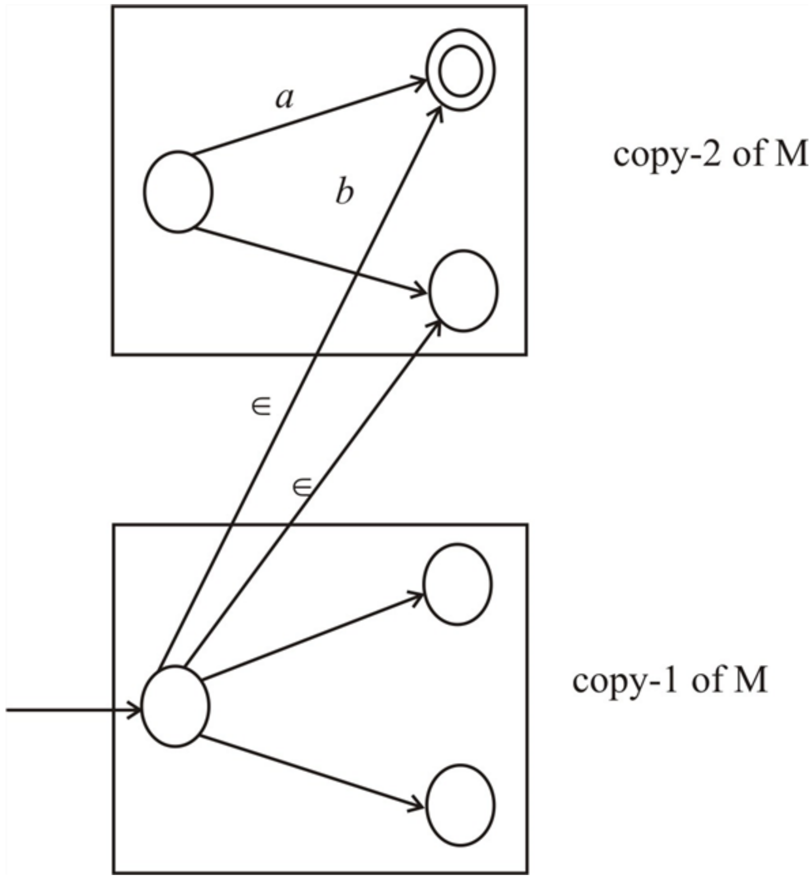
• Copy 1: Copy 1 corresponds to the state of having 'not yet skipped a symbol'

• Copy 2: Copy 2 corresponds to the state of having "already skipped a symbol".

Comment

Step 2 of 4

(i) Proof by picture:-



Comment

Step 3 of 4

$N=(Q',\Sigma\cup\{\in\},\delta',q_0',F')$

- $Q'=\{(q,b)\mid q\in Q, b\in\{0,1\}\}$ = set of states
- $q_0'=$ start state

$=(q_0,0)$

- $F'=$ set of final states

$=\{(q,1)\mid q\in F\}$

- δ' is gives as follows:

$\rightarrow\ \delta'((q,b),a)=\{(\delta(q,a),b)\}\ \forall q\in Q, b\in\{0,1\}, a\in\Sigma$

This means that both the copy1 and copy2 of the machine *M* do exactly as the original machine does on every symbol a of the alphabet

$\rightarrow\ \delta'((q,0),\in)=\{(\hat{q},1)\mid \exists a\in\Sigma, \delta(q,a)=\hat{q}\}$

Also at every stage, the machine has the option to skip a character. The only accepting sates are in copy 2. This means, the machine cannot accept a string without skipping a character.

Comment

Step 4 of 4

(ii) <u>Formal proof:</u>

$\rightarrow$ The formal proof is given by induction on the length the string.

$\rightarrow$ An appropriate inductive hypothesis is to assume that, for any string *w* of length *k*,

• The machine *M* stays in the copy -1 iff it has not yet skipped a symbol.

i.e. $\delta'^*((q_0,0),w)=(q_1,0)$ iff $\delta^*(q_0,w)=q_1$

• The machine *M* jumps to the copy-2 iff there is some symbol a that is skipped.

i.e. $\delta'^*((q_0,0),w)=(q_1,1)$ iff $\delta^*(q_0,w_1 a w_2)=q_1$.

So in both (i) and (ii) we constructed an NFA N that recognizes the language $DROP\_OUT(A)$.

Thus $DROP\_OUT(A)$ is regular.

Hence class of regular languages closed under $\mathrm{DROP\_OUT}$ operation.

Comment