Homomorphism

A homomorphism is a map $h : \Sigma^* \to \Gamma^*$
s.t. $\forall x, y \in \Sigma^*$, $h(xy) = h(x)\, h(y)$ — (1)
$h(\epsilon) = \epsilon$ — (2)

(2) $\Rightarrow$ (1)

$$|h(\epsilon)| = |h(\epsilon\epsilon)|$$
$$= |h(\epsilon)\, h(\epsilon)|$$
$$= |h(\epsilon)| + |h(\epsilon)|$$
$$\Rightarrow |h(\epsilon)| = 0 \quad \Rightarrow h(\epsilon) = \epsilon$$

We shall see applications later on.

// Minimization

1. $\forall x \in \Sigma^*$, $\hat{\delta}'([p], x) = [\hat{\delta}(p, x)]$

$x = \epsilon \rightarrow$

$\hat{\delta}'([p], \epsilon) = [p]$ defn of $\hat{\delta}'$

$= [\hat{\delta}(p, \epsilon)]$ defn of $\hat{\delta}$

Assume $\hat{\delta}'([p], x) = [\hat{\delta}(p, x)]$, & $a \in \Sigma$

$\hat{\delta}'([p], xa) = \delta'(\hat{\delta}'([p], x), a)$ defn of $\hat{\delta}'$

$= \delta'([\hat{\delta}(p, x)], a)$ Ind Hyp

$= [\delta(\hat{\delta}(p, x), a)]$ defn of δ'

$= [\hat{\delta}(p, xa)]$ defn of $\hat{\delta}$

2. $L(M/\approx) = L(M)$

$x \in L(M/\approx) \iff \hat{\delta}'(s', x) \in F'$ — defn of acceptance

$\iff \hat{\delta}'([s], x) \in F'$ — defn of s'

$\iff [\hat{\delta}(s, x)] \in F'$

$\iff \hat{\delta}(s, x) \in F$

$\iff x \in L(M)$

Regular languages are closed under Homomorphism

Let h be any homomorphism defined over Σ^*.

① If R is a regular expression defined over Σ, $h(R)$ is also regular.
What really is $h(R)$? for any $a \in \Sigma$ in R, then replace 'a' by the string $h(a)$. → can prove this inductively.

↳ on regex.

② Given, $h : \Sigma^* \to \Gamma^*$, if some $L \subseteq \Sigma^*$ is regular,
$h(L) = \{ h(w) / w \in L \} \subseteq \Gamma^*$ is also regular.

③ Can show that, $h(L_1 \cup L_2) = h(L_1) \cup h(L_2)$,
$h(L_1 \circ L_2) = h(L_1) \circ h(L_2)$, $h(L_1^*) = (h(L_1))^*$