P2 A y 9

Vamos a demostrar que dado un autómata apilador "M que acepta por stack vacío ó estado final", podamos construir un M' tal que

$$\mathcal{L}(M) = \mathcal{L}(M')$$

y además el lenguaje de stack vacío y estado final son el mismo para M'.

Suponga se tiene $M = (Q, \Sigma, \Gamma, \delta, s, \perp, F)$

y sean u, t estados que no están en Q y $\perp\!\!\perp$ un nuevo símbolo que no está en Γ.

Luego definimos G y H tal que

$$G = \begin{cases} Q & \text{si } M \text{ acepta por stack vacío} \\ F & \text{si acepta por estado final} \end{cases}$$

$$H = \begin{cases} \{\perp\} & \text{si } M \text{ acepta por stack vacío} \\ \Gamma \cup \{\perp\!\!\perp\} & \text{" " " estado final.} \end{cases}$$

Luego nuestro $M' = (Q \cup \{u, t\}, \Gamma \cup \{\perp\!\!\!\perp\}, \delta', u, \perp\!\!\!\perp, \{t\})$

Donde δ' tiene todo lo de δ junto con las siguientes transiciones:

$((u, \epsilon, \perp\!\!\!\perp), (s, \perp \underline{\perp\!\!\!\perp}))$

$((q, \epsilon, A), (t, A))$, $q \in G$ y $A \in H$

$((t, \epsilon, A), (t, \epsilon))$, $A \in \Gamma \cup \{\perp\!\!\!\perp\}$

La idea es simular ~~el~~ M dentro de M' (usando $\perp\!\!\!\perp$ como el ~~el~~ tope del stack nuevo que contiene al de M) y si vaciamos el stack ~~o no~~ o estamos en un estado final, M' se asegurará de manera no determinista de ir a t y de ahí vaciar el stack, haciendo que ambos lenguajes coincidan.

P.D. $L(M) = L(M')$

$\underline{L(M) \subseteq L(M')}$

Suponga M acepta por stack vacío $\Longrightarrow$

~~X~~

sea $X \in L(M)$, por lo tanto

$$(s, X, \perp) \xrightarrow[M]{n} (q, \epsilon, \epsilon)$$

para algún n.

Por lo tanto

$$(u, X, \perp\!\!\perp) \xrightarrow[M']{1} (s, X, \perp\perp\!\!\perp) \xrightarrow[M']{n} (q, \epsilon, \perp\!\!\perp) \xrightarrow[M']{1} (t, \epsilon, \perp\!\!\perp)$$

$$\xrightarrow[M']{1} (t, \epsilon, \epsilon)$$

$\Rightarrow X \in L(M')$

Ahora suponga M acepta por est. final:

$$(s, X, \perp) \xrightarrow[M]{n} (q, \epsilon, \gamma), \quad q \in F$$

Entonces para M':

$$(u, X, \perp\!\!\perp) \xrightarrow[M']{1} (s, X, \perp\perp\!\!\perp) \xrightarrow[M']{n} (q, \epsilon, \gamma\perp\!\!\perp) \xrightarrow[M']{1} (t, \epsilon, \gamma\perp\!\!\perp)$$

$$\xrightarrow[M']{*} (t, \epsilon, \epsilon).$$

$\therefore L(M) \subseteq L(M')$ en ambos casos.

<u>$L(M') \subseteq L(M)$</u>

sea $X \in L(M') \Longrightarrow$

$$(u, x, \perp\!\!\perp) \xrightarrow[M']{1} (s, x, \perp\perp\!\!\perp) \xrightarrow[M']{n} (q, y, \gamma\perp\!\!\perp) \xrightarrow[M']{1} (t, y, \gamma\perp\!\!\perp)$$
$$\xrightarrow[M']{*} (t, \epsilon, \epsilon).$$

para algún $q \in G$. Pero $y = \epsilon$ ya que M' no puede leer nada una vez que está en el estado t. Por ende

$$(s, x, \perp) \xrightarrow[M]{n} (q, \epsilon, \gamma) \qquad (1)$$

Ahora considerando este paso

$$(q, \epsilon, \gamma\perp\!\!\perp) \xrightarrow[M']{1} (t, \epsilon, \gamma\perp\!\!\perp)$$

Si M acepta por estado final entonces $q \in F$ por la definición de G. Por otro lado si M acepta por stack vacío entonces $\gamma = \epsilon$.

$\therefore$ Si M acepta por stack (1) se ve como

$$(s, x, \perp) \xrightarrow[M]{n} (q, \epsilon, \epsilon)$$

y si acepta por est final

$$(s, x, \perp\!\!\perp) \xrightarrow[M]{n} (q, \epsilon, \gamma) \quad q \in F$$

$\therefore$ $x \in L(M)$ y $L(M) \subseteq L(M)$

$$\Rightarrow L(M) = L(M')$$