NOMBRE: Rafael Fernández Sánchez NOTA:

Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación

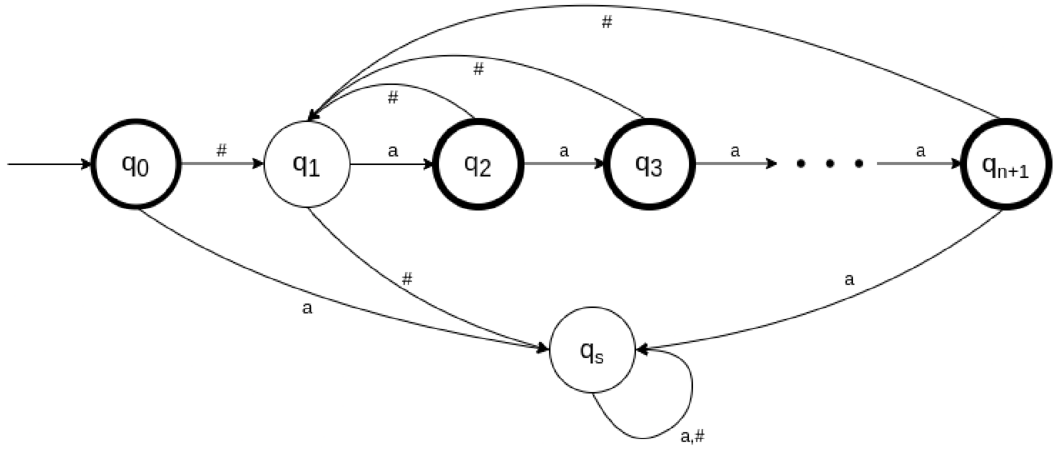
IIC2223 — Teoría de Autómatas y Lenguajes Formales 2020-2

Tarea 1 – Respuesta Pregunta 1

1.1

Un autómata que acepta el lenguaje pedido es el siguiente: $\mathcal{A}_n = (Q, \Sigma, \delta, q_0, F)$, donde:
$\Sigma = \{\#, a\}$
$Q = \{q_0, q_1, ..., q_n\} \cup \{q_s\}$
$q_0 = q_0$
$F = \{q_0\} \cup \{q_2, q_3, ..., q_{n+1}\}$
$\delta = \{((q_0, \#), q_1), ((q_0, a), q_s), ((q_1, a), q_2), ((q_1, \#), q_s)\} \cup$
$\{((q_s, a), q_s), ((q_s, \#), q_s)\} \cup$
$\{((q_2, a), q_3), ((q_3, a), q_4), ..., ((q_n, a), q_{n+1})\} \cup$
$\{((q_2, \#), q_1), ((q_3, \#), q_1), ..., ((q_{n+1}, \#), q_1)\} \cup$
$\{((q_{n+1}, a), q_s)\}$

Este autómata se puede representar gráficamente de las siguiente forma:

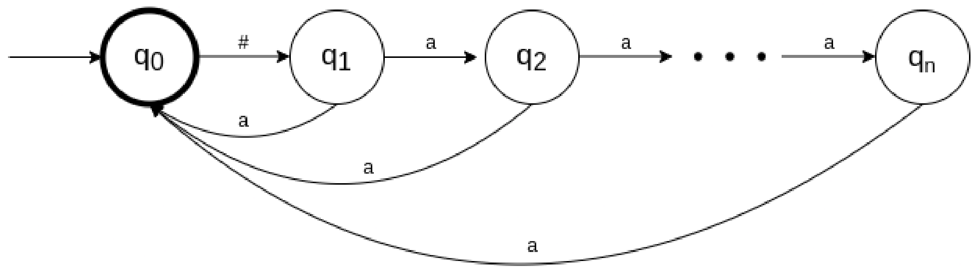


El autómata $\mathcal{A}_n$ tiene un estado sumidero q_s donde cualquier ejecución que lo contenga, será de rechazo. Podemos ver que $\epsilon \in \mathcal{L}(\mathcal{A}_n)$ ya que $q_0 \in F$. $\mathcal{A}_n$ verificará que la primera letra sea $\#$, de lo contrario pasa al sumidero q_s y rechaza. Luego, verifica que la segunda letra sea a. Para las letras siguientes, vemos que se simula un contador usando los estados $q_2, ..., q_{n+1}$, de forma que si la letra es a, cambiamos de estado q_i a q_{i+1}, simulando un aumento en una unidad en el contador. En el caso que haya $i \geq n$ letras a seguidas, se pasará del estado q_{i+1} a q_s, rechazando así la palabra. Por último, en el caso que se encuentre $\#$ y el estado es entre q_2 y q_{n+1} (contador entre 1 y n), el autómata regresa a q_1, simulando que el contador se establece en cero. De esta forma logramos que $\mathcal{L}(\mathcal{A}_n) = L_n$

1.2

Podemos construir $\mathcal{B}_n = (Q, \Sigma, \Delta, I, F)$, de la siguiente manera:
$\Sigma = \{\#, a\}$
$Q = \{q_0, q_1, ..., q_n\}$
$I = \{q_0\}$
$F = \{q_0\}$
$\Delta = \{(q_0, \#, q_1)\} \cup \{(q_1, a, q_2), ..., (q_{n-1}, a, q_n)\} \cup \{(q_1, a, q_0), ..., (q_n, a, q_0)\}$

Este autómata se puede representar gráficamente de las siguiente forma:



Podemos ver que $\mathcal{B}_n$ acepta a ϵ ya que $q_0 \in F$. Luego, se verifica que la primera letra sea #. En caso contrario se rechaza ya que no hay transición para a en q_0. Luego, para cada a, se va "simultáneamente"al estado q_{i+1} y q_0. Si la cantidad de a seguidas es menor o igual n, entonces siempre habrá una ejecución que termina en q_0, ya que se podrá ir en la última a, desde q_n a q_0, terminando la ejecución en el estado final q_0 y por lo tanto aceptando. En caso que se tenga una cantidad seguidas de a mayor a n, entonces todas las ejecuciones terminarán en q_0, pero quedarán a por consumir, y dado que q_0 no tiene transición para a, se rechazarán esas palabras. Por último, siguiendo la lógica anterior, si el autómata lee # al estar en el estado q_0 (lo cual se dará si hay n o menos letras a seguidas), entonces pasará al estado q_1, reinciando el proceso. De esta forma logramos que $\mathcal{L}(\mathcal{B}_n) = L_n$