Autómatas de Pila (APs)

Matemáticas Computacionales (TC2020)

M.C. Xavier Sánchez Díaz
sax@itesm.mx

¿Autómatas de Pila?

Descripción informal de un AP

Un Autómata de Pila (AP) es un Autómata pero con una Pila (duh).

El autómata tiene una pila, la cual sirve como **memoria** extra para poder hacer operaciones más complicadas.

Los Autómatas de Pila son **equivalentes** a las Gramáticas Libres de Contexto—sirven para representar lenguajes libres de contexto.

¿Autómatas de Pila?

Descripción informal de un AP

Un Autómata de Pila (AP) es un Autómata pero con una Pila (duh).

El autómata tiene una pila, la cual sirve como **memoria** extra para poder hacer operaciones más complicadas.

Los Autómatas de Pila son **equivalentes** a las Gramáticas Libres de Contexto—sirven para representar lenguajes libres de contexto.

¿Autómatas de Pila?

Descripción informal de un AP

Un Autómata de Pila (AP) es un Autómata pero con una Pila (duh).

El autómata tiene una pila, la cual sirve como **memoria** extra para poder hacer operaciones más complicadas.

Los Autómatas de Pila son **equivalentes** a las Gramáticas Libres de Contexto—sirven para representar lenguajes libres de contexto.

¿Qué es un Autómata de Pila?

Descripción informal de un AP

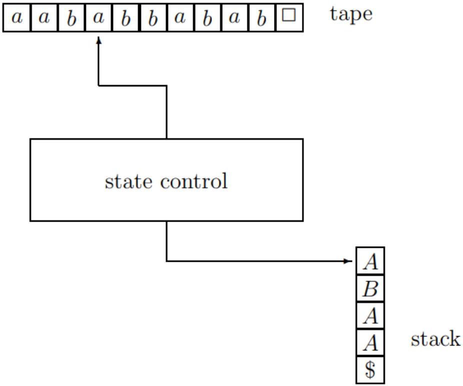


Figure 3.1: *A pushdown automaton.*

Imagen de Maheshwari and Smid's *Theory of Computation*, 2017

¿Qué es un Autómata de Pila?

Descripción informal de un AP

Un AP tiene varios elementos:

1) Una cinta dividida en **celdas**

- Cada celda contiene un **símbolo** de la palabra de entrada el cual pertenece a un **alfabeto** Σ.
- Al final de la cinta, tenemos un **símbolo nuevo** para representar el final de la palabra: $\square$. Este nuevo símbolo no pertenece al alfabeto de la palabra.

2) Un cabezal en la cinta que puede leer el valor de la celda en la que está, y que puede efectuar dos acciones $\sigma = \{N, R\}$: N si no se mueve y R si se mueve a la derecha.

¿Qué es un Autómata de Pila?

Descripción informal de un AP

Un AP tiene varios elementos:

1) Una cinta dividida en **celdas**

- Cada celda contiene un **símbolo** de la palabra de entrada el cual pertenece a un **alfabeto** Σ.
- Al final de la cinta, tenemos un **símbolo nuevo** para representar el final de la palabra: $\square$. Este nuevo símbolo no pertenece al alfabeto de la palabra.

2) Un cabezal en la cinta que puede leer el valor de la celda en la que está, y que puede efectuar dos acciones $\sigma = \{N, R\}$: N si no se mueve y R si se mueve a la derecha.

¿Qué es un Autómata de Pila?

Descripción informal de un AP

Un AP tiene varios elementos:

1) Una cinta dividida en **celdas**

- Cada celda contiene un **símbolo** de la palabra de entrada el cual pertenece a un **alfabeto** Σ.
- Al final de la cinta, tenemos un **símbolo nuevo** para representar el final de la palabra: $\square$. Este nuevo símbolo no pertenece al alfabeto de la palabra.

2) Un cabezal en la cinta que puede leer el valor de la celda en la que está, y que puede efectuar dos acciones $\sigma = \{N, R\}$: N si no se mueve y R si se mueve a la derecha.

¿Qué es un Autómata de Pila?

Descripción informal de un AP

Un AP tiene varios elementos:

1) Una cinta dividida en **celdas**

- Cada celda contiene un **símbolo** de la palabra de entrada el cual pertenece a un **alfabeto** Σ.
- Al final de la cinta, tenemos un **símbolo nuevo** para representar el final de la palabra: $\square$. Este nuevo símbolo no pertenece al alfabeto de la palabra.

2) Un cabezal en la cinta que puede leer el valor de la celda en la que está, y que puede efectuar dos acciones $\sigma = \{N, R\}$: N si no se mueve y R si se mueve a la derecha.

¿Qué es un Autómata de Pila?

Descripción informal de un AP

Un AP tiene varios elementos:

1) Una cinta dividida en **celdas**

- Cada celda contiene un **símbolo** de la palabra de entrada el cual pertenece a un **alfabeto** Σ.
- Al final de la cinta, tenemos un **símbolo nuevo** para representar el final de la palabra: $\square$. Este nuevo símbolo no pertenece al alfabeto de la palabra.

2) Un cabezal en la cinta que puede leer el valor de la celda en la que está, y que puede efectuar dos acciones $\sigma = \{N, R\}$: N si no se mueve y R si se mueve a la derecha.

0	0	1	1	■

3) Una pila que puede guardar símbolos.

- Los símbolos que pueden almacenarse en la pila pertenecen a **otro alfabeto**: Γ.
- Uno de estos símbolos es $\$$, el cual sí está **dentro del alfabeto de la pila**.

4) Un cabezal en la pila el cual lee **el último símbolo** de la misma.

- El cabezal puede *apilar* (`push`) más símbolos, o bien *retirar* el símbolo de hasta arriba (`pop`).

5) Un conjunto de estados, unidos por transiciones que dependen de los símbolos leídos por **ambos cabezales**.

¿Qué es un Autómata de Pila?

Descripción informal de un AP

3) Una pila que puede guardar símbolos.

- Los símbolos que pueden almacenarse en la pila pertenecen a **otro alfabeto**: Γ.
- Uno de estos símbolos es $\$$, el cual sí está **dentro del alfabeto de la pila**.

4) Un cabezal en la pila el cual lee **el último símbolo** de la misma.

- El cabezal puede *apilar* (`push`) más símbolos, o bien *retirar* el símbolo de hasta arriba (`pop`).

5) Un conjunto de estados, unidos por transiciones que dependen de los símbolos leídos por **ambos cabezales**.

¿Qué es un Autómata de Pila?

Descripción informal de un AP

3) Una pila que puede guardar símbolos.

- Los símbolos que pueden almacenarse en la pila pertenecen a **otro alfabeto**: Γ.
- Uno de estos símbolos es $\$$, el cual sí está **dentro del alfabeto de la pila**.

4) Un cabezal en la pila el cual lee **el último símbolo** de la misma.

- El cabezal puede *apilar* (`push`) más símbolos, o bien *retirar* el símbolo de hasta arriba (`pop`).

5) Un conjunto de estados, unidos por transiciones que dependen de los símbolos leídos por **ambos cabezales**.

¿Qué es un Autómata de Pila?

Descripción informal de un AP

3) Una pila que puede guardar símbolos.

- Los símbolos que pueden almacenarse en la pila pertenecen a **otro alfabeto**: Γ.
- Uno de estos símbolos es $\$$, el cual sí está **dentro del alfabeto de la pila**.

4) Un cabezal en la pila el cual lee **el último símbolo** de la misma.

- El cabezal puede *apilar* (`push`) más símbolos, o bien *retirar* el símbolo de hasta arriba (`pop`).

5) Un conjunto de estados, unidos por transiciones que dependen de los símbolos leídos por **ambos cabezales**.

¿Qué es un Autómata de Pila?

Descripción informal de un AP

3) Una pila que puede guardar símbolos.

- Los símbolos que pueden almacenarse en la pila pertenecen a **otro alfabeto**: Γ.
- Uno de estos símbolos es $\$$, el cual sí está **dentro del alfabeto de la pila**.

4) Un cabezal en la pila el cual lee **el último símbolo** de la misma.

- El cabezal puede *apilar* (`push`) más símbolos, o bien *retirar* el símbolo de hasta arriba (`pop`).

5) Un conjunto de estados, unidos por transiciones que dependen de los símbolos leídos por **ambos cabezales**.

3) Una pila que puede guardar símbolos.

- Los símbolos que pueden almacenarse en la pila pertenecen a **otro alfabeto**: Γ.
- Uno de estos símbolos es $\$$, el cual sí está **dentro del alfabeto de la pila**.

4) Un cabezal en la pila el cual lee **el último símbolo** de la misma.

- El cabezal puede *apilar* (`push`) más símbolos, o bien *retirar* el símbolo de hasta arriba (`pop`).

5) Un conjunto de estados, unidos por transiciones que dependen de los símbolos leídos por **ambos cabezales**.

Una pila como estructura de datos

Descripción informal de un AP

Una pila (en inglés *stack*) funciona por medio del principio LIFO, que significa *Last In, First Out*—el último elemento en entrar es el primero en salir.

Ejemplo

Sean A una pila con los valores $A = \langle 1, 2, 3 \rangle$ y $F = \{\mathrm{pop}, \mathrm{push}\}$ un conjunto de funciones *in-place* aplicables a pilas.

Si aplicamos la función pop sobre A, entonces obtendremos el valor 3, y la pila pasará a ser $A = \langle 1, 2 \rangle$.

Si después metemos un valor más—digamos 4—a la pila (proceso para el cual usamos la función push), entonces la pila es ahora $A = \langle 1, 2, 4 \rangle$.

¿Qué pasa si aplicamos pop nuevamente?

Una pila como estructura de datos

Descripción informal de un AP

Una pila (en inglés *stack*) funciona por medio del principio LIFO, que significa *Last In, First Out*—el último elemento en entrar es el primero en salir.

Ejemplo

Sean A una pila con los valores $A = \langle 1, 2, 3 \rangle$ y $F = \{\texttt{pop}, \texttt{push}\}$ un conjunto de funciones *in-place* aplicables a pilas.

Si aplicamos la función pop sobre A, entonces obtendremos el valor 3, y la pila pasará a ser $A = \langle 1, 2 \rangle$.

Si después metemos un valor más—digamos 4—a la pila (proceso para el cual usamos la función $\texttt{push}$), entonces la pila es ahora $A = \langle 1, 2, 4 \rangle$.

¿Qué pasa si aplicamos pop nuevamente?

Una pila como estructura de datos

Descripción informal de un AP

Una pila (en inglés *stack*) funciona por medio del principio LIFO, que significa *Last In, First Out*—el último elemento en entrar es el primero en salir.

Ejemplo

Sean A una pila con los valores $A = \langle 1, 2, 3 \rangle$ y $F = \{\mathtt{pop}, \mathtt{push}\}$ un conjunto de funciones *in-place* aplicables a pilas.

Si aplicamos la función pop sobre A, entonces obtendremos el valor 3, y la pila pasará a ser $A = \langle 1, 2 \rangle$.

Si después metemos un valor más—digamos 4—a la pila (proceso para el cual usamos la función $\mathtt{push}$), entonces la pila es ahora $A = \langle 1, 2, 4 \rangle$.

¿Qué pasa si aplicamos pop nuevamente?

Una pila como estructura de datos

Descripción informal de un AP

Una pila (en inglés *stack*) funciona por medio del principio LIFO, que significa *Last In, First Out*—el último elemento en entrar es el primero en salir.

Ejemplo

Sean A una pila con los valores $A = \langle 1, 2, 3\rangle$ y $F = \{\texttt{pop}, \texttt{push}\}$ un conjunto de funciones *in-place* aplicables a pilas.

Si aplicamos la función pop sobre A, entonces obtendremos el valor 3, y la pila pasará a ser $A = \langle 1, 2\rangle$.

Si después metemos un valor más—digamos 4—a la pila (proceso para el cual usamos la función push), entonces la pila es ahora $A = \langle 1, 2, 4\rangle$.

¿Qué pasa si aplicamos pop nuevamente?

Una pila como estructura de datos

Descripción informal de un AP

Una pila (en inglés *stack*) funciona por medio del principio LIFO, que significa *Last In, First Out*—el último elemento en entrar es el primero en salir.

Ejemplo

Sean A una pila con los valores $A = \langle 1, 2, 3 \rangle$ y $F = \{\texttt{pop}, \texttt{push}\}$ un conjunto de funciones *in-place* aplicables a pilas.

Si aplicamos la función pop sobre A, entonces obtendremos el valor 3, y la pila pasará a ser $A = \langle 1, 2 \rangle$.

Si después metemos un valor más—digamos 4—a la pila (proceso para el cual usamos la función push), entonces la pila es ahora $A = \langle 1, 2, 4 \rangle$.

¿Qué pasa si aplicamos pop nuevamente?

Definiciones formales

Formalización y diseño de APs

Hay **muchas** maneras de expresar los APs y sus elementos:

- Brena (2003) utiliza transiciones expresadas en términos del input y `pop` y `push` de la pila, pertenecientes a una *relación de transición*, con estados finales y nuevos símbolos.
- Maheshwari y Smid (2017) utilizan una función de transición **completa** en términos del estado, el input, movimiento en la cinta ($\sigma = \{N, R\}$) y la función `replace` de la pila, sin estados finales.
- Tinelli (2016) utiliza transiciones expresadas en términos del input y la función `replace` de la pila, con estados finales.

Nosotros usaremos **una combinación de los tres**: diseñamos usando la notación de Maheshwari y Smid, refinamos usando la notación de Brena y con los símbolos de Tinelli because we're cool like that.

Definiciones formales

Formalización y diseño de APs

Hay **muchas** maneras de expresar los APs y sus elementos:

- Brena (2003) utiliza transiciones expresadas en términos del input y pop y `push` de la pila, pertenecientes a una *relación de transición*, con estados finales y nuevos símbolos.
- Maheshwari y Smid (2017) utilizan una función de transición **completa** en términos del estado, el input, movimiento en la cinta ($\sigma = \{N, R\}$) y la función `replace` de la pila, sin estados finales.
- Tinelli (2016) utiliza transiciones expresadas en términos del input y la función `replace` de la pila, con estados finales.

Nosotros usaremos **una combinación de los tres**: diseñamos usando la notación de Maheshwari y Smid, refinamos usando la notación de Brena y con los símbolos de Tinelli because we're cool like that.

Definiciones formales

Formalización y diseño de APs

Hay **muchas** maneras de expresar los APs y sus elementos:

- Brena (2003) utiliza transiciones expresadas en términos del input y pop y `push` de la pila, pertenecientes a una *relación de transición*, con estados finales y nuevos símbolos.

- Maheshwari y Smid (2017) utilizan una función de transición **completa** en términos del estado, el input, movimiento en la cinta ($\sigma = \{N, R\}$) y la función `replace` de la pila, sin estados finales.

- Tinelli (2016) utiliza transiciones expresadas en términos del input y la función `replace` de la pila, con estados finales.

Nosotros usaremos **una combinación de los tres**: diseñamos usando la notación de Maheshwari y Smid, refinamos usando la notación de Brena y con los símbolos de Tinelli because we're cool like that.

Hay **muchas** maneras de expresar los APs y sus elementos:

- Brena (2003) utiliza transiciones expresadas en términos del input y pop y `push` de la pila, pertenecientes a una *relación de transición*, con estados finales y nuevos símbolos.
- Maheshwari y Smid (2017) utilizan una función de transición **completa** en términos del estado, el input, movimiento en la cinta ($\sigma = \{N, R\}$) y la función `replace` de la pila, sin estados finales.
- Tinelli (2016) utiliza transiciones expresadas en términos del input y la función `replace` de la pila, con estados finales.

Nosotros usaremos **una combinación de los tres**: diseñamos usando la notación de Maheshwari y Smid, refinamos usando la notación de Brena y con los símbolos de Tinelli because we're cool like that.

Hay **muchas** maneras de expresar los APs y sus elementos:

- Brena (2003) utiliza transiciones expresadas en términos del input y `pop` y `push` de la pila, pertenecientes a una *relación de transición*, con estados finales y nuevos símbolos.

- Maheshwari y Smid (2017) utilizan una función de transición **completa** en términos del estado, el input, movimiento en la cinta ($\sigma = \{N, R\}$) y la función `replace` de la pila, sin estados finales.

- Tinelli (2016) utiliza transiciones expresadas en términos del input y la función `replace` de la pila, con estados finales.

Nosotros usaremos **una combinación de los tres**: diseñamos usando la notación de Maheshwari y Smid, refinamos usando la notación de Brena y con los símbolos de Tinelli because we're cool like that.

Definiciones formales

Formalización y diseño de APs

Hay **muchas** maneras de expresar los APs y sus elementos:

- Brena (2003) utiliza transiciones expresadas en términos del input y `pop` y `push` de la pila, pertenecientes a una *relación de transición*, con estados finales y nuevos símbolos.
- Maheshwari y Smid (2017) utilizan una función de transición **completa** en términos del estado, el input, movimiento en la cinta ($\sigma = \{N, R\}$) y la función `replace` de la pila, sin estados finales.
- Tinelli (2016) utiliza transiciones expresadas en términos del input y la función `replace` de la pila, con estados finales.

Nosotros usaremos **una combinación de los tres**: diseñamos usando la notación de Maheshwari y Smid, refinamos usando la notación de Brena y con los símbolos de Tinelli because we're cool like that.

Definición Formal

Formalización y diseño de APs

Definición de un Autómata de Pila

Un autómata de pila M es una tupla de la forma $M = (Q, \Sigma, \Gamma, \delta, q, F)$ donde:

- Q es un conjunto finito de **estados**,
- Σ es el **alfabeto de la cinta** (sin incluir $\square$),
- Γ es el **alfabeto de la pila** (incluyendo $\$$),
- $q \in Q$ es el **estado inicial**,
- $F \subseteq Q$ es un conjunto finito de **estados finales** y
- δ es la función de transición.

Definición Formal

Formalización y diseño de APs

Definición de un Autómata de Pila

Un autómata de pila M es una tupla de la forma $M = (Q, \Sigma, \Gamma, \delta, q, F)$ donde:

- Q es un conjunto finito de **estados**,
- Σ es el **alfabeto de la cinta** (sin incluir $\square$),
- Γ es el **alfabeto de la pila** (incluyendo $\$$),
- $q \in Q$ es el **estado inicial**,
- $F \subseteq Q$ es un conjunto finito de **estados finales** y
- δ es la función de transición.

Definición Formal

Formalización y diseño de APs

Definición de un Autómata de Pila

Un autómata de pila M es una tupla de la forma $M = (Q, \Sigma, \Gamma, \delta, q, F)$ donde:

- Q es un conjunto finito de **estados**,
- Σ es el **alfabeto de la cinta** (sin incluir $\Box$),
- Γ es el **alfabeto de la pila** (incluyendo $\$$),
- $q \in Q$ es el **estado inicial**,
- $F \subseteq Q$ es un conjunto finito de **estados finales** y
- δ es la función de transición.

Definición de un Autómata de Pila

Un autómata de pila M es una tupla de la forma $M = (Q, \Sigma, \Gamma, \delta, q, F)$ donde:

- Q es un conjunto finito de **estados**,
- Σ es el **alfabeto de la cinta** (sin incluir $\Box$),
- Γ es el **alfabeto de la pila** (incluyendo $\$$),
- $q \in Q$ es el **estado inicial**,
- $F \subseteq Q$ es un conjunto finito de **estados finales** y
- δ es la función de transición.

Definición Formal

Formalización y diseño de APs

Definición de un Autómata de Pila

Un autómata de pila M es una tupla de la forma $M = (Q, \Sigma, \Gamma, \delta, q, F)$ donde:

- Q es un conjunto finito de **estados**,
- Σ es el **alfabeto de la cinta** (sin incluir $\Box$),
- Γ es el **alfabeto de la pila** (incluyendo $\$$),
- $q \in Q$ es el **estado inicial**,
- $F \subseteq Q$ es un conjunto finito de **estados finales** y
- δ es la función de transición.

Definición Formal

Formalización y diseño de APs

Definición de un Autómata de Pila

Un autómata de pila M es una tupla de la forma $M = (Q, \Sigma, \Gamma, \delta, q, F)$ donde:

- Q es un conjunto finito de **estados**,
- Σ es el **alfabeto de la cinta** (sin incluir $\Box$),
- Γ es el **alfabeto de la pila** (incluyendo $\$$),
- $q \in Q$ es el **estado inicial**,
- $F \subseteq Q$ es un conjunto finito de **estados finales** y
- δ es la función de transición.

Definición Formal

Formalización y diseño de APs

Definición de un Autómata de Pila

Un autómata de pila M es una tupla de la forma $M = (Q, \Sigma, \Gamma, \delta, q, F)$ donde:

- Q es un conjunto finito de **estados**,
- Σ es el **alfabeto de la cinta** (sin incluir $\square$),
- Γ es el **alfabeto de la pila** (incluyendo $\$$),
- $q \in Q$ es el **estado inicial**,
- $F \subseteq Q$ es un conjunto finito de **estados finales** y
- δ es la función de transición.

Definición Formal

Formalización y diseño de APs

La función de transición d es una función de la forma:

$$\delta : Q \times (\Sigma \cup \{\square\}) \times \Gamma \to Q \times \{N, R\} \times \Gamma^*$$

Ejemplo

Podemos escribir $q_0 1 S \to q_1 RSS$ que significaría que:

- Estando en el estado q_0,
- al recibir un 1,
- y si el tope de la pila tiene S

entonces el AP

- cambia al estado q_1,
- mueve la cinta hacia la derecha (Right) y
- reemplaza el tope de la pila por SS.

Definición Formal

Formalización y diseño de APs

La función de transición d es una función de la forma:

$$\delta : Q \times (\Sigma \cup \{\square\}) \times \Gamma \to Q \times \{N, R\} \times \Gamma^*$$

Ejemplo

Podemos escribir $q_0 1 S \to q_1 RSS$ que significaría que:

- Estando en el estado q_0,
- al recibir un 1,
- y si el tope de la pila tiene S

entonces el AP

- cambia al estado q_1,
- mueve la cinta hacia la derecha (Right) y
- reemplaza el tope de la pila por SS.

Definición Formal

Formalización y diseño de APs

La función de transición d es una función de la forma:

$$\delta : Q \times (\Sigma \cup \{\square\}) \times \Gamma \to Q \times \{N, R\} \times \Gamma^*$$

Ejemplo

Podemos escribir $q_0 1S \to q_1 RSS$ que significaría que:

- Estando en el estado q_0,
- al recibir un 1,
- y si el tope de la pila tiene S

entonces el AP

- cambia al estado q_1,
- mueve la cinta hacia la derecha (Right) y
- reemplaza el tope de la pila por SS.

Definición Formal

Formalización y diseño de APs

La función de transición d es una función de la forma:

$$\delta : Q \times (\Sigma \cup \{\Box\}) \times \Gamma \to Q \times \{N, R\} \times \Gamma^*$$

Ejemplo

Podemos escribir $q_0 1 S \to q_1 RSS$ que significaría que:

- Estando en el estado q_0,
- al recibir un 1,
- y si el tope de la pila tiene S

entonces el AP

- cambia al estado q_1,
- mueve la cinta hacia la derecha (Right) y
- reemplaza el tope de la pila por SS.

Definición Formal

Formalización y diseño de APs

La función de transición d es una función de la forma:

$$\delta : Q \times (\Sigma \cup \{\square\}) \times \Gamma \to Q \times \{N, R\} \times \Gamma^*$$

Ejemplo

Podemos escribir $q_0 1 S \to q_1 RSS$ que significaría que:

- Estando en el estado q_0,
- al recibir un 1,
- y si el tope de la pila tiene S

entonces el AP

- cambia al estado q_1,
- mueve la cinta hacia la derecha (Right) y
- reemplaza el tope de la pila por SS.

Definición Formal

Formalización y diseño de APs

La función de transición d es una función de la forma:

$$\delta : Q \times (\Sigma \cup \{\Box\}) \times \Gamma \to Q \times \{N, R\} \times \Gamma^*$$

Ejemplo

Podemos escribir $q_0 1 S \to q_1 RSS$ que significaría que:

- Estando en el estado q_0,
- al recibir un 1,
- y si el tope de la pila tiene S

entonces el AP

- cambia al estado q_1,
- mueve la cinta hacia la derecha (Right) y
- reemplaza el tope de la pila por SS.

Definición Formal

Formalización y diseño de APs

La función de transición d es una función de la forma:

$$\delta : Q \times (\Sigma \cup \{\Box\}) \times \Gamma \to Q \times \{N, R\} \times \Gamma^*$$

Ejemplo

Podemos escribir $q_0 1 S \to q_1 RSS$ que significaría que:

- Estando en el estado q_0,
- al recibir un 1,
- y si el tope de la pila tiene S

entonces el AP

- cambia al estado q_1,
- mueve la cinta hacia la derecha (Right) y
- reemplaza el tope de la pila por SS.

Definición Formal

Formalización y diseño de APs

La función de transición d es una función de la forma:

$$\delta : Q \times (\Sigma \cup \{\square\}) \times \Gamma \to Q \times \{N, R\} \times \Gamma^*$$

Ejemplo

Podemos escribir $q_0 1 S \to q_1 R S S$ que significaría que:

- Estando en el estado q_0,
- al recibir un 1,
- y si el tope de la pila tiene S

entonces el AP

- cambia al estado q_1,
- mueve la cinta hacia la derecha (Right) y
- reemplaza el tope de la pila por SS.

Consideraciones adicionales

Formalización y diseño de APs

Configuración inicial

- El AP empieza en el estado q.
- El cabezal de la cinta empieza en el símbolo inicial de la palabra w.
- La pila empieza con un solo símbolo, $\$$.

Cómputo y terminación

El AP hace una serie de pasos de cómputo y *termina* en el momento en que la pila se vacía. Si la pila no se vacía, entonces el programa no termina (*loop* infinito).

Aceptación

El AP acepta la palabra w si se cumplen las condiciones siguientes:

1. el autómata termina, y
2. al momento de terminar, (i.e. la pila se vacía) el cabezal de la cinta está en el símbolo $\square$.

Consideraciones adicionales

Formalización y diseño de APs

Configuración inicial

- El AP empieza en el estado q.
- El cabezal de la cinta empieza en el símbolo inicial de la palabra w.
- La pila empieza con un solo símbolo, $\$$.

Cómputo y terminación

El AP hace una serie de pasos de cómputo y *termina* en el momento en que la pila se vacía. Si la pila no se vacía, entonces el programa no termina (*loop* infinito).

Aceptación

El AP acepta la palabra w si se cumplen las condiciones siguientes:

1. el autómata termina, y
2. al momento de terminar, (i.e. la pila se vacía) el cabezal de la cinta está en el símbolo $\Box$.

Consideraciones adicionales

Configuración inicial

- El AP empieza en el estado q.
- El cabezal de la cinta empieza en el símbolo inicial de la palabra w.
- La pila empieza con un solo símbolo, $\$$.

Cómputo y terminación

El AP hace una serie de pasos de cómputo y *termina* en el momento en que la pila se vacía. Si la pila no se vacía, entonces el programa no termina (*loop* infinito).

Aceptación

El AP acepta la palabra w si se cumplen las condiciones siguientes:

1. el autómata termina, y
2. al momento de terminar, (i.e. la pila se vacía) el cabezal de la cinta está en el símbolo $\square$.

Consideraciones adicionales

Formalización y diseño de APs

Configuración inicial

- El AP empieza en el estado q.
- El cabezal de la cinta empieza en el símbolo inicial de la palabra w.
- La pila empieza con un solo símbolo, $\$$.

Cómputo y terminación

El AP hace una serie de pasos de cómputo y *termina* en el momento en que la pila se vacía. Si la pila no se vacía, entonces el programa no termina (*loop* infinito).

Aceptación

El AP acepta la palabra w si se cumplen las condiciones siguientes:

1. el autómata termina, y
2. al momento de terminar, (i.e. la pila se vacía) el cabezal de la cinta está en el símbolo $\square$.

Consideraciones adicionales

Formalización y diseño de APs

Configuración inicial

- El AP empieza en el estado q.
- El cabezal de la cinta empieza en el símbolo inicial de la palabra w.
- La pila empieza con un solo símbolo, $\$$.

Cómputo y terminación

El AP hace una serie de pasos de cómputo y *termina* en el momento en que la pila se vacía. Si la pila no se vacía, entonces el programa no termina (*loop* infinito).

Aceptación

El AP acepta la palabra w si se cumplen las condiciones siguientes:

1. el autómata termina, y
2. al momento de terminar, (i.e. la pila se vacía) el cabezal de la cinta está en el símbolo $\square$.

Consideraciones adicionales

Formalización y diseño de APs

Configuración inicial

- El AP empieza en el estado q.
- El cabezal de la cinta empieza en el símbolo inicial de la palabra w.
- La pila empieza con un solo símbolo, $\$$.

Cómputo y terminación

El AP hace una serie de pasos de cómputo y *termina* en el momento en que la pila se vacía. Si la pila no se vacía, entonces el programa no termina (*loop* infinito).

Aceptación

El AP acepta la palabra w si se cumplen las condiciones siguientes:

1. el autómata termina, y
2. al momento de terminar, (i.e. la pila se vacía) el cabezal de la cinta está en el símbolo $\Box$.

Ejemplo: *Matching Parentheses*

Formalización y diseño de APs

Ejemplos de palabras aceptadas: $(), ((())), ()(), \ldots$

Estructura:

- Antes de terminar de leer toda la palabra, la cantidad de "(" debe ser mayor o igual que ")", y
- Al terminar de leer toda la palabra, el número de "(" debe ser igual al número de ")".

Para no complicarnos, usaremos a para representar "(" y b para representar ")" en la cinta.

Por cada a leída, metemos una S en la pila. Por cada b que se lea, sacamos entonces una S. ¿Cuántos estados necesitamos? ¿Cuáles son finales?

Es un proceso **complicado**, por lo que se sugiere primero definir el AP formalmente (e.g. la función de transición completa, y parte por parte), y luego pasarlo a un diagrama con una relación de transición más pequeña: $Q \times \Sigma \cup \{\Box\} \times \Gamma \to Q \times \Gamma^*$.

¿Por qué no consideramos si se mueve o no se mueve el cabezal de la cinta?

Ejemplo: *Matching Parentheses*

Formalización y diseño de APs

Ejemplos de palabras aceptadas: $(), ((())), ()(), \ldots$
Estructura:

- Antes de terminar de leer toda la palabra, la cantidad de "(" debe ser mayor o igual que ")", y
- Al terminar de leer toda la palabra, el número de "(" debe ser igual al número de ")".

Para no complicarnos, usaremos a para representar "(" y b para representar ")" en la cinta.

Por cada a leída, metemos una S en la pila. Por cada b que se lea, sacamos entonces una S. ¿Cuántos estados necesitamos? ¿Cuáles son finales?
Es un proceso **complicado**, por lo que se sugiere primero definir el AP formalmente (e.g. la función de transición completa, y parte por parte), y luego pasarlo a un diagrama con una relación de transición más pequeña: $Q \times \Sigma \cup \{\square\} \times \Gamma \to Q \times \Gamma^*$.
¿Por qué no consideramos si se mueve o no se mueve el cabezal de la cinta?

Ejemplo: *Matching Parentheses*

Formalización y diseño de APs

Ejemplos de palabras aceptadas: $(), ((())), ()(), \ldots$
Estructura:

- Antes de terminar de leer toda la palabra, la cantidad de "(" debe ser mayor o igual que ")", y
- Al terminar de leer toda la palabra, el número de "(" debe ser igual al número de ")".

Para no complicarnos, usaremos a para representar "(" y b para representar ")" en la cinta.

Por cada a leída, metemos una S en la pila. Por cada b que se lea, sacamos entonces una S. ¿Cuántos estados necesitamos? ¿Cuáles son finales?
Es un proceso **complicado**, por lo que se sugiere primero definir el AP formalmente (e.g. la función de transición completa, y parte por parte), y luego pasarlo a un diagrama con una relación de transición más pequeña: $Q \times \Sigma \cup \{\Box\} \times \Gamma \to Q \times \Gamma^*$.
¿Por qué no consideramos si se mueve o no se mueve el cabezal de la cinta?

Ejemplo: *Matching Parentheses*

Formalización y diseño de APs

Ejemplos de palabras aceptadas: $(), ((())), ()(), \ldots$
Estructura:

- Antes de terminar de leer toda la palabra, la cantidad de "(" debe ser mayor o igual que ")", y
- Al terminar de leer toda la palabra, el número de "(" debe ser igual al número de ")".

Para no complicarnos, usaremos a para representar "(" y b para representar ")" en la cinta.

Por cada a leída, metemos una S en la pila. Por cada b que se lea, sacamos entonces una S. ¿Cuántos estados necesitamos? ¿Cuáles son finales?
Es un proceso **complicado**, por lo que se sugiere primero definir el AP formalmente (e.g. la función de transición completa, y parte por parte), y luego pasarlo a un diagrama con una relación de transición más pequeña: $Q \times \Sigma \cup \{\Box\} \times \Gamma \to Q \times \Gamma^*$.
¿Por qué no consideramos si se mueve o no se mueve el cabezal de la cinta?

Ejemplo: *Matching Parentheses*

Formalización y diseño de APs

Ejemplos de palabras aceptadas: $(), ((())), ()(), \ldots$
Estructura:

- Antes de terminar de leer toda la palabra, la cantidad de "(" debe ser mayor o igual que ")", y
- Al terminar de leer toda la palabra, el número de "(" debe ser igual al número de ")".

Para no complicarnos, usaremos a para representar "(" y b para representar ")" en la cinta.

Por cada a leída, metemos una S en la pila. Por cada b que se lea, sacamos entonces una S. ¿Cuántos estados necesitamos? ¿Cuáles son finales?
Es un proceso **complicado**, por lo que se sugiere primero definir el AP formalmente (e.g. la función de transición completa, y parte por parte), y luego pasarlo a un diagrama con una relación de transición más pequeña: $Q \times \Sigma \cup \{\square\} \times \Gamma \to Q \times \Gamma^*$.
¿Por qué no consideramos si se mueve o no se mueve el cabezal de la cinta?

Ejemplo: *Matching Parentheses*

Formalización y diseño de APs

Ejemplos de palabras aceptadas: $(), ((())), ()(), \ldots$
Estructura:

- Antes de terminar de leer toda la palabra, la cantidad de "(" debe ser mayor o igual que ")", y
- Al terminar de leer toda la palabra, el número de "(" debe ser igual al número de ")".

Para no complicarnos, usaremos a para representar "(" y b para representar ")" en la cinta.

Por cada a leída, metemos una S en la pila. Por cada b que se lea, sacamos entonces una S. ¿Cuántos estados necesitamos? ¿Cuáles son finales?
Es un proceso **complicado**, por lo que se sugiere primero definir el AP formalmente (e.g. la función de transición completa, y parte por parte), y luego pasarlo a un diagrama con una relación de transición más pequeña: $Q \times \Sigma \cup \{\Box\} \times \Gamma \to Q \times \Gamma^*$.
¿Por qué no consideramos si se mueve o no se mueve el cabezal de la cinta?

Ejemplo: *Matching Parentheses*

Formalización y diseño de APs

Ejemplos de palabras aceptadas: $(), ((())), ()(), \ldots$
Estructura:

- Antes de terminar de leer toda la palabra, la cantidad de "(" debe ser mayor o igual que ")", y
- Al terminar de leer toda la palabra, el número de "(" debe ser igual al número de ")".

Para no complicarnos, usaremos a para representar "(" y b para representar ")" en la cinta.

Por cada a leída, metemos una S en la pila. Por cada b que se lea, sacamos entonces una S. ¿Cuántos estados necesitamos? ¿Cuáles son finales?
Es un proceso **complicado**, por lo que se sugiere primero definir el AP formalmente (e.g. la función de transición completa, y parte por parte), y luego pasarlo a un diagrama con una relación de transición más pequeña: $Q \times \Sigma \cup \{\square\} \times \Gamma \to Q \times \Gamma^*$.
¿Por qué no consideramos si se mueve o no se mueve el cabezal de la cinta?

Ejemplo: *Matching Parentheses*

Formalización y diseño de APs

Ejemplos de palabras aceptadas: $(), ((())), ()(), \ldots$
Estructura:

- Antes de terminar de leer toda la palabra, la cantidad de "(" debe ser mayor o igual que ")", y
- Al terminar de leer toda la palabra, el número de "(" debe ser igual al número de ")".

Para no complicarnos, usaremos a para representar "(" y b para representar ")" en la cinta.

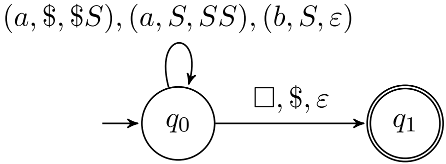
Por cada a leída, metemos una S en la pila. Por cada b que se lea, sacamos entonces una S. ¿Cuántos estados necesitamos? ¿Cuáles son finales?
Es un proceso **complicado**, por lo que se sugiere primero definir el AP formalmente (e.g. la función de transición completa, y parte por parte), y luego pasarlo a un diagrama con una relación de transición más pequeña: $Q \times \Sigma \cup \{\Box\} \times \Gamma \to Q \times \Gamma^*$.
¿Por qué no consideramos si se mueve o no se mueve el cabezal de la cinta?

Ejemplo: *Matching Parentheses*

Formalización y diseño de APs

$M = (Q, \Sigma, \Gamma, \delta, q, F)$:

- $Q = \{q_0, q_1\}$
- $\Sigma = \{a, b\}$
- $\Gamma = \{\$, S\}$
- $\delta =$
  - $((q_0, a, \$)(q_0, \$S))$
  - $((q_0, a, S)(q_0, SS))$
  - $((q_0, b, S)(q_0, \varepsilon))$
  - $((q_0, \square, \$), (q_1, \varepsilon))$
- $q = q_0$
- $F = \{q_1\}$

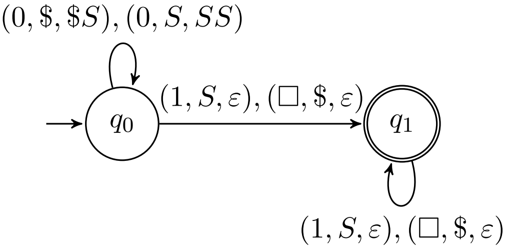


Ejemplo: $\{0^n 1^n\}$

Formalización y diseño de APs

$M = (Q, \Sigma, \Gamma, \delta, q, F)$:

- $Q = \{q_0, q_1\}$
- $\Sigma = \{0, 1\}$
- $\Gamma = \{\$, S\}$
- $\delta =$
  - $((q_0, 0, \$)(q_0, \$S))$
  - $((q_0, 0, S)(q_0, SS))$
  - $((q_0, 1, S)(q_1, \varepsilon))$
  - $((q_0, \square, \$), (q_1, \varepsilon))$
  - $((q_1, 1, S)(q_1, \varepsilon))$
  - $((q_1, \square, \$)(q_1, \varepsilon))$
- $q = q_0$
- $F = \{q_1\}$



Combinación y concatenación

Formalización y diseño de APs

Combinación de APs

De manera muy similar a unir dos AFNs, la idea es hacer un estado inicial **previo** que una los estados iniciales de los APs usando transiciones vacías $(\varepsilon, \varepsilon, \varepsilon)$.

Concatenación de APs

La concatenación funciona de manera muy similar que como era en AFNs, sin embargo hay que garantizar que la pila se encuentra en *ciertas condiciones* antes de pasar al siguiente AP. La solución es utilizar un símbolo especial antes de iniciar con el primer AP, y sacarlo antes de iniciar la operación con el segundo.

Combinación y concatenación

Formalización y diseño de APs

Combinación de APs

De manera muy similar a unir dos AFNs, la idea es hacer un estado inicial **previo** que una los estados iniciales de los APs usando transiciones vacías $(\varepsilon, \varepsilon, \varepsilon)$.

Concatenación de APs

La concatenación funciona de manera muy similar que como era en AFNs, sin embargo hay que garantizar que la pila se encuentra en *ciertas condiciones* antes de pasar al siguiente AP. La solución es utilizar un símbolo especial antes de iniciar con el primer AP, y sacarlo antes de iniciar la operación con el segundo.