

# Orientación a Objetos

## Estados

Verónica E. Arriola-Rios

Facultad de Ciencias, UNAM

19 de noviembre de 2025



Estados

- 1 Estados
  - 2 Autómatas
  - 3 Ejemplo: Autómatas Celulares

## Temas

1 Estados

- Definiciones
  - Estados en un programa orientado a objetos
  - Ejemplo

## Estado

## Definición

- El concepto de *estado* se utiliza para especificar cómo se encuentra un sistema en un tiempo dado.
  - Esta descripción se da en términos de *variables*, cada una de las cuales toma su valor de un *dominio*<sup>a</sup> que le es propio.
  - Al transcurrir el tiempo los valores de estas variables se pueden modificar, para describir esta dinámica se utilizan:
    - *Máquinas de estados finitas* o *Autómatas de estados finitos*
    - *Sistemas de transiciones*<sup>b</sup>

<sup>a</sup>El dominio es el conjunto de valores posibles para una variable.

<sup>b</sup> Transition system 2017.

## Temas

1 Estados

- Definiciones
  - Estados en un programa orientado a objetos
  - Ejemplo

## Estados en un programa orientado a objetos

- En un programa, el estado es el conjunto de valores asignados, en un tiempo dado, a todas las **variables vivas** del programa: atributos de objetos existentes y variables locales.
  - Para facilitar el análisis del comportamiento del programa, consideraremos los estados de objetos individuales y las transiciones entre ellos.
  - Cuando los objetos interactúan entre sí, necesitaremos considerar al estado como dado por la unión de los atributos de los objetos en comunicación.

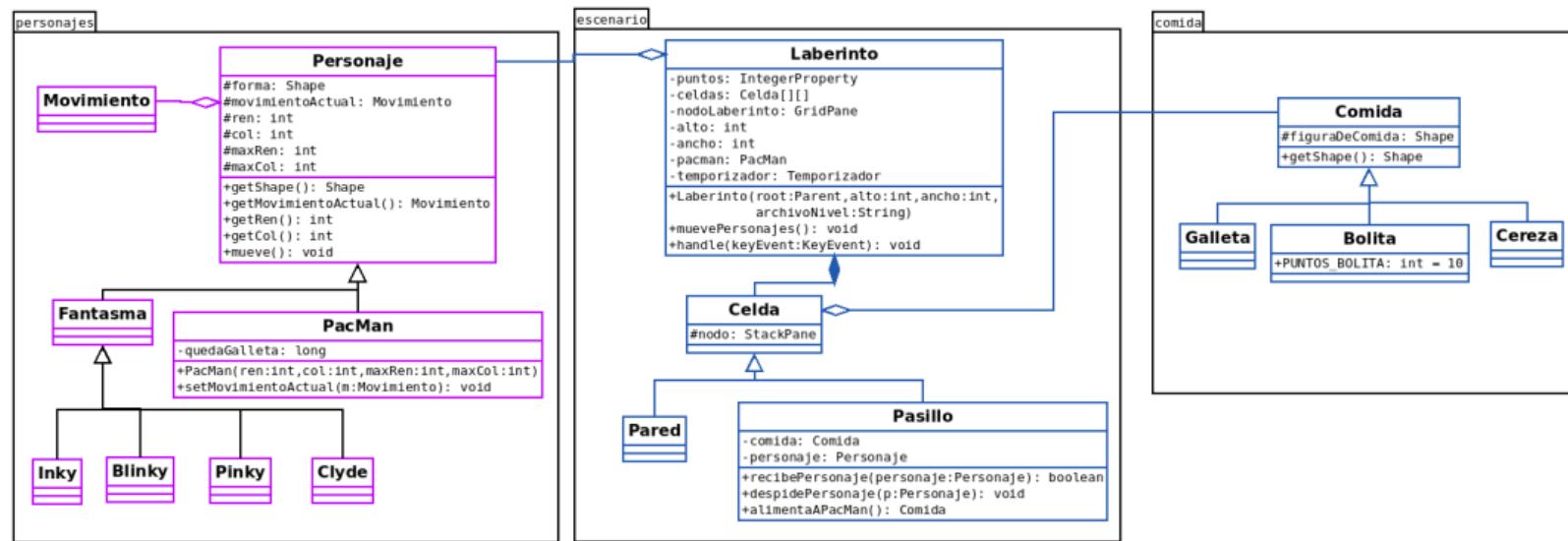
## Temas

## 1 Estados

- Definiciones
  - Estados en un programa orientado a objetos
  - **Ejemplo**

# PacMan

El siguiente diagrama de clases nos puede ayudar a visualizar las variables que definen el estado del programa para jugar PacMan.



# PacMan

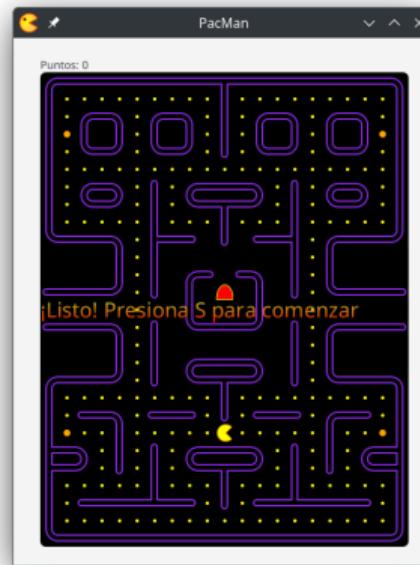


Figura: Juego implementado con PakuMan y un fantasma que lo persigue usando A\*

# Autómatas

1 Estados

2 Autómatas

3 Ejemplo: Autómatas Celulares

# Temas

## 2 Autómatas

- Autómatas finitos
- Ejemplo: Pacman
- Máquina de estados

# Autómatas finitos o máquinas de estados

“Un autómata finito contiene un conjunto de estados  $Q$  y un conjunto de transiciones de estado a estado que ocurren cuando al autómata se le alimentan símbolos de un alfabeto  $\Sigma$ .” Viso G. 2008

## Definición

Un *autómata finito M* es un quíntuplo  $M = (Q, \Sigma, \delta, q_0, F)$  donde:

- $\Sigma$ : Alfabeto finito de entrada
- $Q$ : Conjunto finito de estados
- $\delta$ : Función de transición que asigna un estado siguiente dados un estado y un símbolo.

$$\delta : Q \times \Sigma \rightarrow Q$$

- $q_0$ : Estado inicial  $q_0 \in Q$
- $F$ : Estados finales, que aceptan la cadena,  $F \subseteq Q$

# Temas

## 2 Autómatas

- Autómatas finitos
- Ejemplo: Pacman
- Máquina de estados

# Pacman

Podemos utilizar un autómata finito para expresar los estados del personaje PacMan.

- $\Sigma$ : Las acciones y eventos que pueden ocurrir: las teclas que presiona el usuario y si PacMan se come una galleta, en cuyo caso podrá comer fantasmas durante un intervalo finito de tiempo.

$$\Sigma = \{\rightarrow, \leftarrow, \uparrow, \downarrow, \text{comeGalleta}, \text{finTiempo}\}$$

- $Q$ : Estados de movimiento de PacMan. Mientras el usuario no presione una dirección, PacMan se sigue desplazando en la última dirección que se le dejó. Entonces, cada estado de PacMan está dado por las combinaciones de valores posibles para sus dos variables:

$$\text{estadoDeMovimiento} \in \{\text{Izquierda}, \text{Derecha}, \text{Abajo}, \text{Arriba}\}$$

$$\text{poder} \in \{\text{Normal}, \text{con Galleta}\}$$

- $\delta$ : Función que indica a qué dirección de movimiento cambia y si transita entre comer o no fantasmas.
- $q_0$ :  $q_0 = \text{Derecha Normal}$
- F: En este caso PacMan solo podría tomar cualquier estado como final. Pero, si consideraramos al estado del laberinto completo, los estados finales serían todos aquellos donde ya no queden bolitas por comer.

# Autómata finito para PacMan

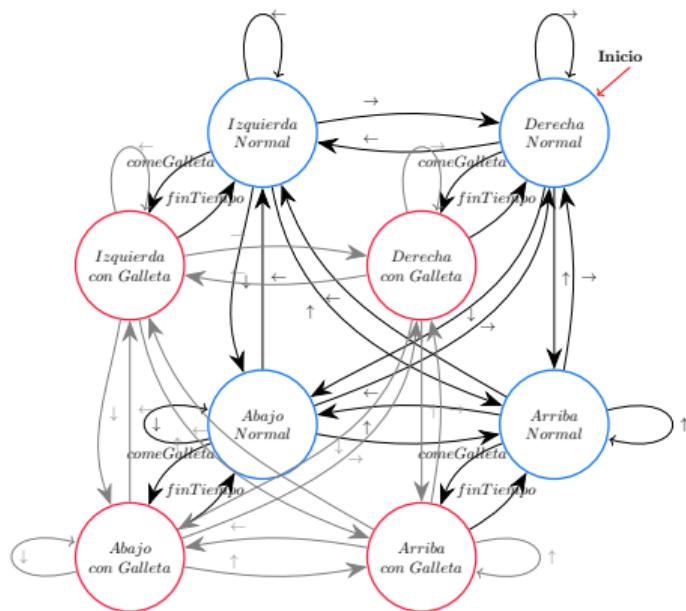


Figura: Estados de PacMan y sus transiciones.

El estado está dado por dos variables: movimientoActual y quedaGalleta.

# Diseño de una calculadora

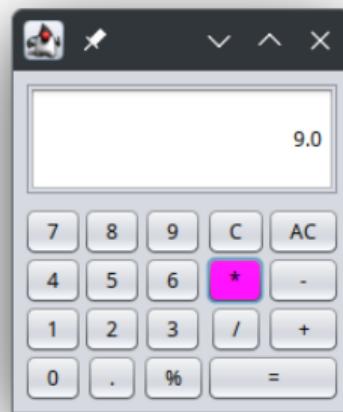


Figura: Aplicación gráfica

<https://github.com/veroarriola/calculadorasimple>

# Autómata

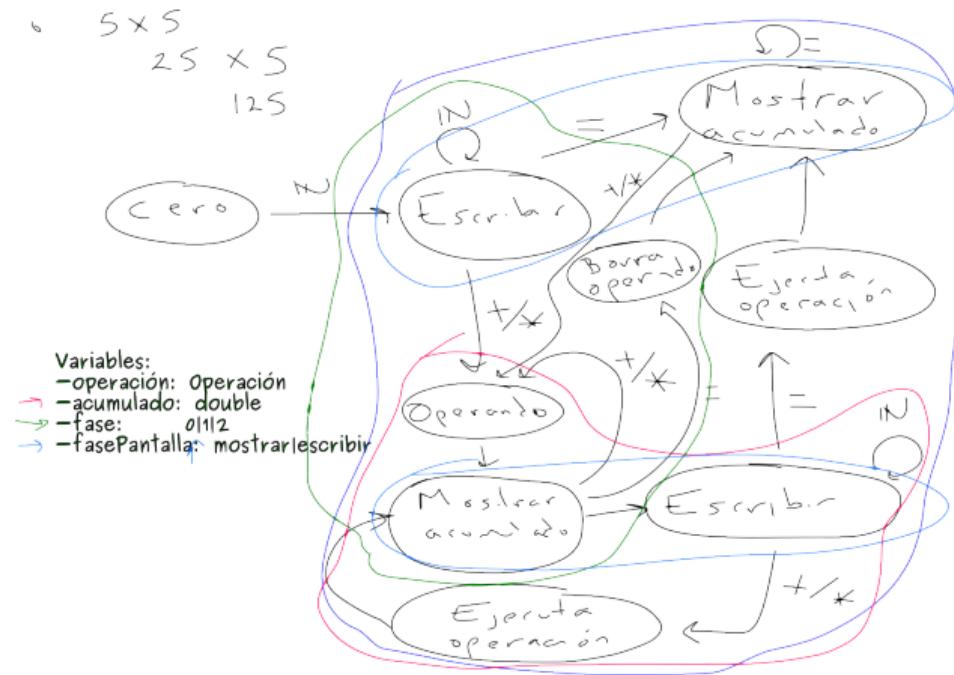


Figura: Fase de diseño

# Autómata en limpio

Variables:

- operación: Operación
- acumulado: double
- fase: 0|1|2
- fasePantalla: mostrar/escribir

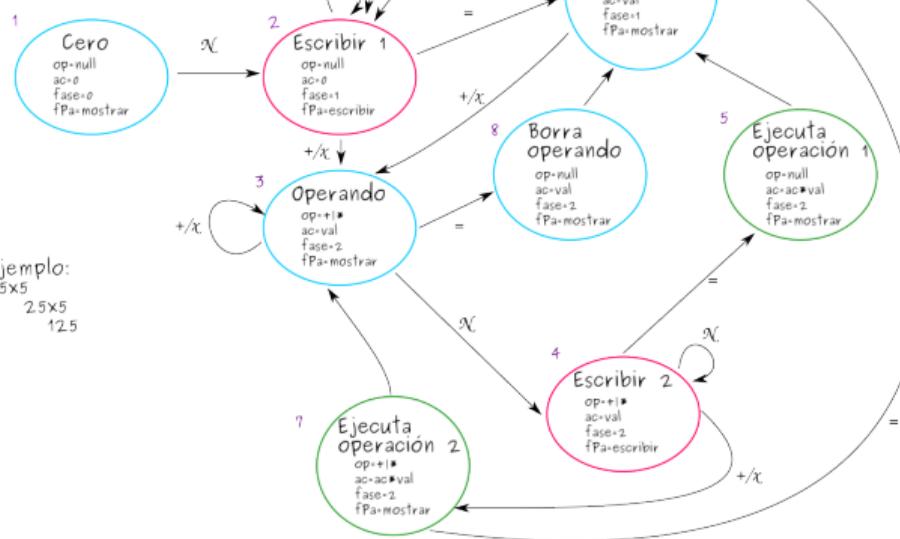


Figura: Autómata en limpio

# Temas

## 2 Autómatas

- Autómatas finitos
- Ejemplo: Pacman
- Máquina de estados

# Máquina de estados

- Es más común escuchar este término en el campo de la ingeniería.
- Es básicamente un autómata finito conocido como *Máquina de Mealy* y sólo agrega el hecho de que puede devolver un símbolo por cada transición que realiza.

# Máquina de estados

## Definición (Máquina de estados)

Una *máquina de estados* es un concepto utilizado para describir el comportamiento posible de un sistema discreto, que evoluciona en el tiempo. Está constituida por:

**Estados** Un conjunto  $S = \{s_1, s_2, \dots\}$  de estados, que representan la configuración actual del sistema. El sistema sólo se puede encontrar en un estado a la vez.

**Entradas** Un conjunto  $I = \{i_1, i_2, \dots\}$  de posibles valores con los cuales se alimentará a la máquina en cada paso temporal.

**Salidas** Se tiene un conjunto  $O = \{o_1, o_2, \dots\}$  de símbolos, que puede emitir la máquina en cada paso temporal. Qué símbolo emita puede depender del estado en el que se encuentre actualmente y del valor de entrada que acabe de recibir.

## Máquina de estados

**Función de transición** Una función  $\gamma$  que describe bajo qué condiciones puede el sistema transitar de un estado hacia otro.

**Función de salida** Una función  $\beta$  que indica de qué depende el símbolo que emitirá la máquina en cada paso temporal.

Fuentes:

- 1 [https://en.wikipedia.org/wiki/Transition%5C\\_system](https://en.wikipedia.org/wiki/Transition%5C_system)
- 2 <https://www.techopedia.com/definition/16447/state-machine>
- 3 [https://github.com/pluginaweek/state\\_machine](https://github.com/pluginaweek/state_machine)

## Ejemplo: Autómatas Celulares

1 Estados

2 Autómatas

3 Ejemplo: Autómatas Celulares

# Temas

## 3 Ejemplo: Autómatas Celulares

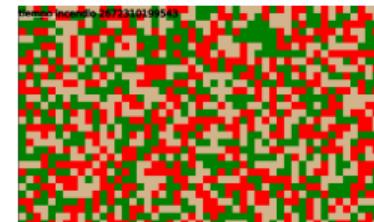
- Definiciones
- Simulación

# Autómata celular

## Definición (Autómata celular)

Un *autómata celular* es un sistema dinámico discreto en el cual el tiempo transcurre por pasos y el espacio se encuentra dividido en *celdas* regulares de igual tamaño y forma acomodadas en una *malla*.

- Cada celda puede tomar un valor de un dominio  $\mathbb{D}$ , común a todas las celdas del autómata.
- El *estado* del autómata al tiempo  $t$  es el conjunto de valores  $x^t$  de todas sus celdas.



## Definición

- El estado del autómata se actualiza del tiempo  $t$  al tiempo  $t + 1$  calculando los nuevos valores para todas sus celdas.

$$x^{t+1} = f(x^t)$$

- El valor de cada celda se calcula con reglas sencillas locales, donde el nuevo valor  $x_{ij}^{t+1}$  depende de los valores de sus celdas vecinas, de modo que:

$$x_{ij}^{t+1} = f(x_{ij}^t)$$

# Vecindad

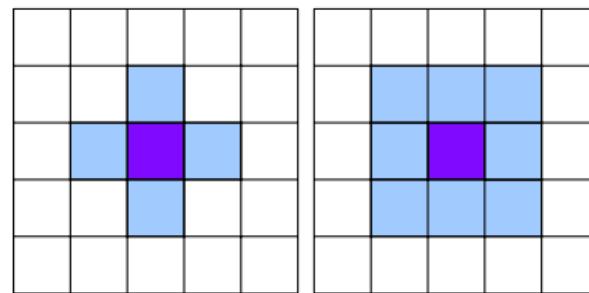


Figura: Izquierda: vecindad de Von Neumann. Derecha: vecindad de Moore. Los dos tipos de vecindades más usuales en autómatas celulares bidimensionales. El estado de la celda central es determinado, en el siguiente paso temporal, por su valor actual y el de las celdas coloreadas a su alrededor.

# Temas

## 3 Ejemplo: Autómatas Celulares

- Definiciones
- Simulación

# Comportamiento emergente

- El resultado que se obtiene al realizar simulaciones con estas reglas sencillas es que el sistema completo manifiesta propiedades que sólo se explican como consecuencia de las interacciones entre sus partes.
- A estos comportamientos complejos, donde se observa que *el todo es más que la suma de las partes* se le conoce como *comportamientos emergentes*.
- Los autómatas son un tipo de ecuaciones en diferencias y son utilizados frecuentemente para modelar fenómenos naturales.

# Ejemplo: simulaciones

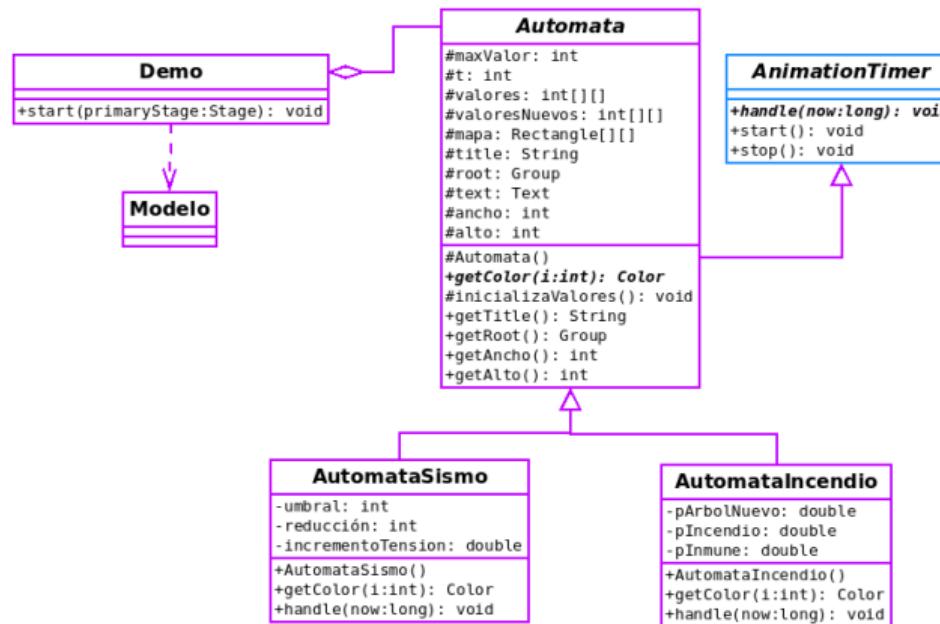


Figura: El arreglo con los estados de las celdas contiene el estado del autómata.

# Bibliografía I

-  *Transition system* (jul. de 2017). Wikimedia Foundation, Inc. URL:  
[https://en.wikipedia.org/wiki/Transition%5C\\_system](https://en.wikipedia.org/wiki/Transition%5C_system).
-  Viso G., Elisa (2008). *Introducción a la teoría de la computación*. Temas de computación. Las prensas de ciencias. 304 págs. ISBN: 978-970-32-54-15-6.

# Licencia

Creative Commons  
Atribución-No Comercial-Compartir Igual

