

# Programas Interactivos

Dante Zanarini

mayo 2020

# Computación Interactiva

- Un **programa interactivo** se comunica mediante eventos de *entrada/salida* **durante** su ejecución.
- Ejemplos:
  - Juegos
  - Sistemas operativos
  - Procesadores de texto
  - Sistemas de mensajería instantánea
- Parece más difícil encontrar algún programa que no lo sea...

# Formas de interactuar con programas

- Una persona puede interactuar mediante:
  - Teclado
  - Mouse
  - Touchscreen

# Formas de interactuar con programas

- Una persona puede interactuar mediante:
  - Teclado
  - Mouse
  - Touchscreen
- Un programa también puede responder a eventos generados por otros programas:
  - Paso del tiempo (ticks) del reloj interno.
  - Arribo de mensajes provenientes de otra computadora.

# Formas de interactuar con programas

- Una persona puede interactuar mediante:
  - Teclado
  - Mouse
  - Touchscreen
- Un programa también puede responder a eventos generados por otros programas:
  - Paso del tiempo (ticks) del reloj interno.
  - Arribo de mensajes provenientes de otra computadora.
- Las salidas pueden ser a través de pantallas, impresoras, parlantes, etc.

# Eventos y manejadores de eventos

- En una computadora, el sistema operativo se encarga de manejar todos los dispositivos.
- Cuando uno **presiona una tecla**, el sistema operativo invoca a una **función que procesa entradas del teclado**.

# Eventos y manejadores de eventos

- En una computadora, el sistema operativo se encarga de manejar todos los dispositivos.
- Cuando uno **presiona una tecla**, el sistema operativo invoca a una **función que procesa entradas del teclado**.
  - **Presionar una tecla** es un **evento**.
  - **La función que lo procesa** es un **manejador de eventos**.

# Programando interacciones

- Diseñar un sistema reactivo requiere diseñar:
  - **Manejadores de eventos** para los que se necesita estar preparados (ticks, teclas, mouse, etc.)
  - Una **función principal** que comunique estas asociaciones al sistema operativo



# Programando interacciones

- Diseñar un sistema reactivo requiere diseñar:
  - **Manejadores de eventos** para los que se necesita estar preparados (ticks, teclas, mouse, etc.)
  - Una **función principal** que comunique estas asociaciones al sistema operativo
- *DrRacket* nos brinda un entorno de ejecución similar a un sistema operativo, y la biblioteca *2htdp/universe* nos provee este mecanismo de comunicación
- Las expresiones **big-bang** nos permiten asociar eventos a manejadores de eventos.

# Programando interacciones

- Diseñar un sistema reactivo requiere diseñar:
  - **Manejadores de eventos** para los que se necesita estar preparados (ticks, teclas, mouse, etc.)
  - Una **función principal** que comunique estas asociaciones al sistema operativo
- *DrRacket* nos brinda un entorno de ejecución similar a un sistema operativo, y la biblioteca *2htdp/universe* nos provee este mecanismo de comunicación
- Las expresiones **big-bang** nos permiten asociar eventos a manejadores de eventos.
- **Démosle un vistazo**