

FUNDAMENTOS DE PROGRAMACION EN VIDEOJUEGOS

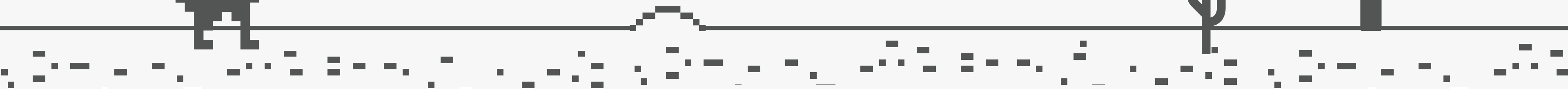
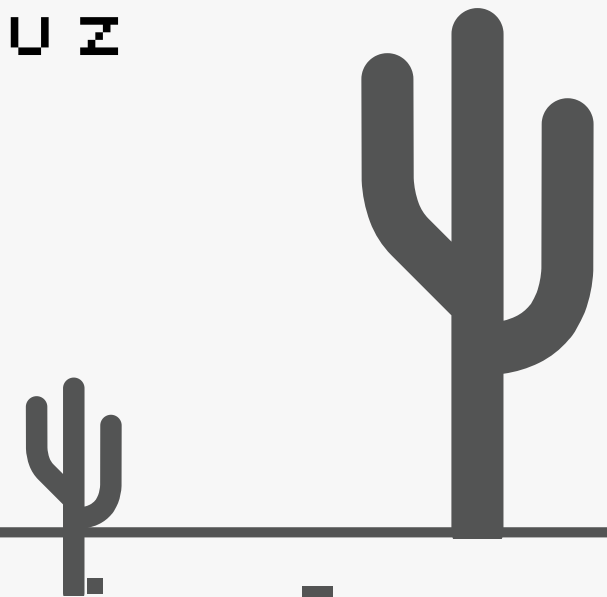
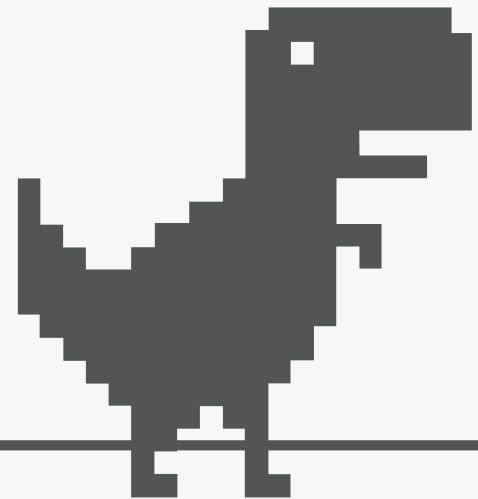


UPSO



# APUNTE\_5

ALUMNA: OPAZO MARIA LUZ





## 1.Estado Actual y Sensores

- Condición: Situación que puede evaluarse como verdadera o falsa. Se utiliza para tomar decisiones en el código.
- Sensores: Primitivas que permiten hacer preguntas sobre el estado del escenario en tiempo real. En Pilas Bloques, todos los sensores son condiciones.

## 2.Alternativa Condicional

- Forma Completa:
  - Estructura que permite elegir entre dos acciones (ramas: si y sino) según el valor de verdad de una condición.

### ■ Ejemplo:

 Copy  Download

```
Si (¿Hay una pelota de fútbol acá?)  
  Patear pelota de fútbol  
Sino  
  Rebotar pelota de ping pong
```

### ○ Forma Simplificada:

- Solo ejecuta una acción si la condición es verdadera. No tiene rama **sino**.

### ■ Ejemplo:

 Copy  Download

```
Si (¿Hay una pelota de fútbol acá?)  
  Patear pelota de fútbol
```



- ***Buenas Prácticas:***

- Evitar duplicación de código: No repetir comandos en ambas ramas de una alternativa condicional si deben ejecutarse siempre.
- ***Dividir en subtareas:*** Separar la lógica de movimiento de la lógica de procesamiento para mejorar la claridad.
- ***No anidar bloques naranjas:*** En Pilas Bloques, los bloques de condiciones no deben anidarse directamente.

- **Ejercicios Analizados**

"La pelota indecisa":

- El escenario puede variar (hay o no una pelota).
- La solución debe funcionar para cualquier escenario posible.

"¿Pelota o paleta?":

- El escenario puede tener una pelota de fútbol o de ping pong, pero no ambas ni ninguna.
- Se utiliza la alternativa condicional completa para manejar las dos posibilidades.

- ***Nuevos Problemas y Soluciones***

Escenarios con múltiples posibilidades:

- Ejemplo: Puede haber ambas pelotas, ninguna, o solo una.
- Soluciones deben contemplar todos los casos posibles.

- **Importancia de los escenarios:**

- La solución adecuada depende de las restricciones del problema.