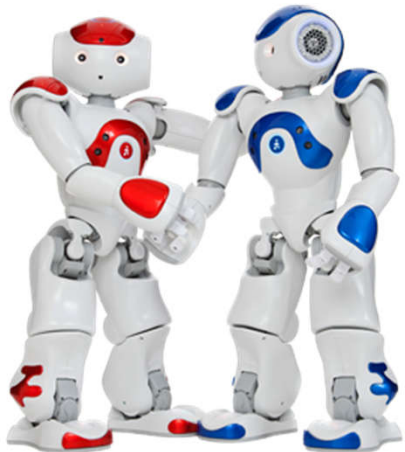


- Día 2: Parte 2 - **Control de Flujo y Reconocimiento**

Prof. Oscar E. Ramos, Ph.D.

*Universidad de Ingeniería y Tecnología (UTECH)
Departamento de Ingeniería Electrónica
14 de febrero del 2018*



Control de Flujo

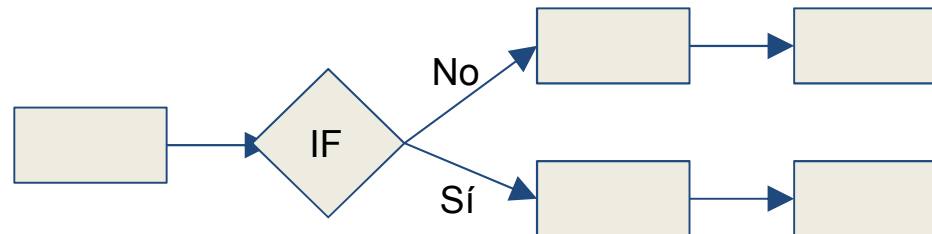
Control de Flujo

- Flujo: “secuencia” de un programa

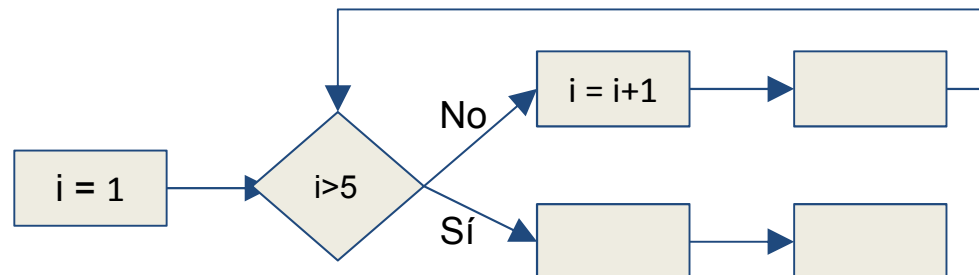
Secuencial:



Condicional (IF):

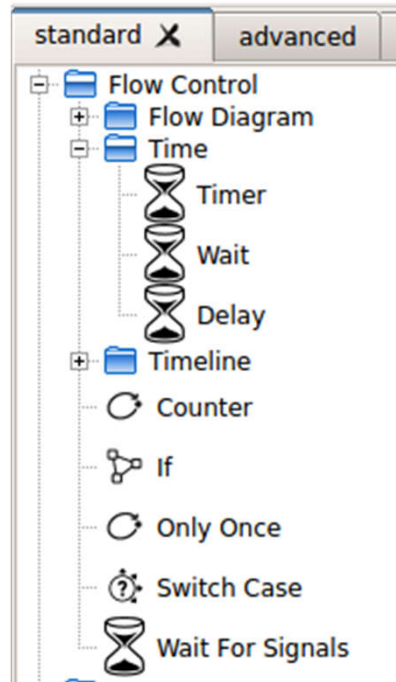


Bucle (“loop”):



Control de Flujo

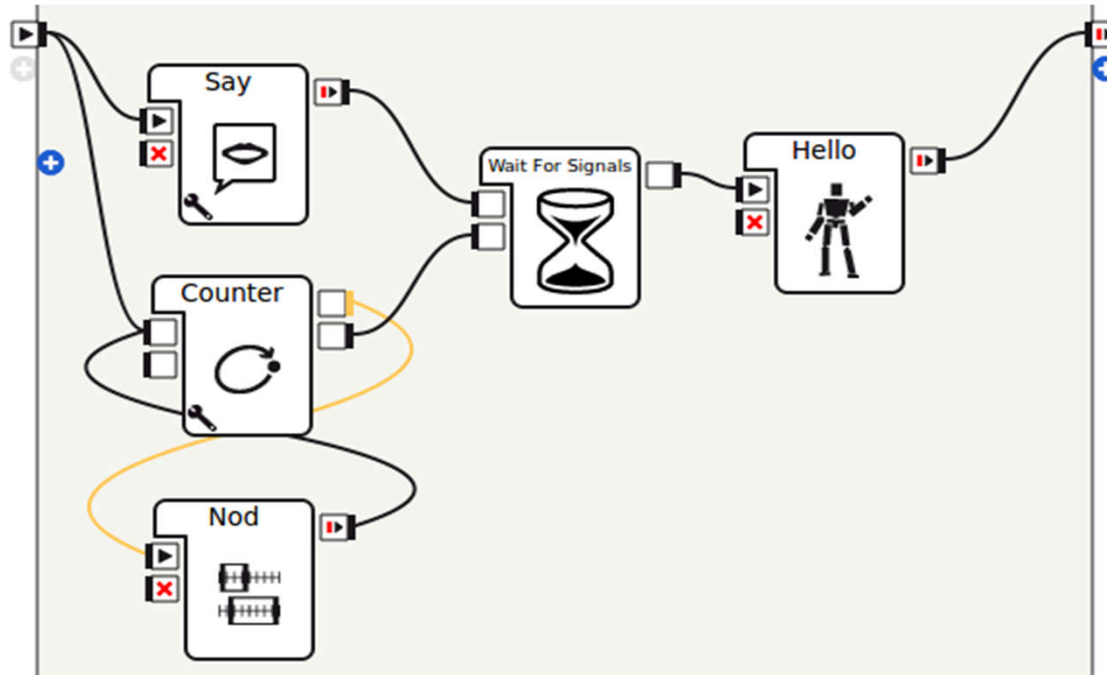
- Para el NAO se usan los siguientes bloques



- **If**: “decisión”
- **Switch case**: selecciona entre sus entradas
- **Counter**: ejecuta una acción un número de veces

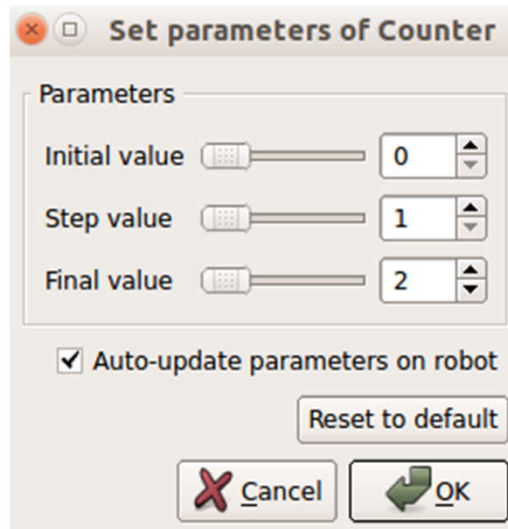
Control de Flujo

- **Objetivo:**
El robot NAO mueve su cabeza 2 veces mientras saluda y luego (de que ambas acciones terminan) mueve la mano
 - Construir el siguiente diagrama



Control de Flujo

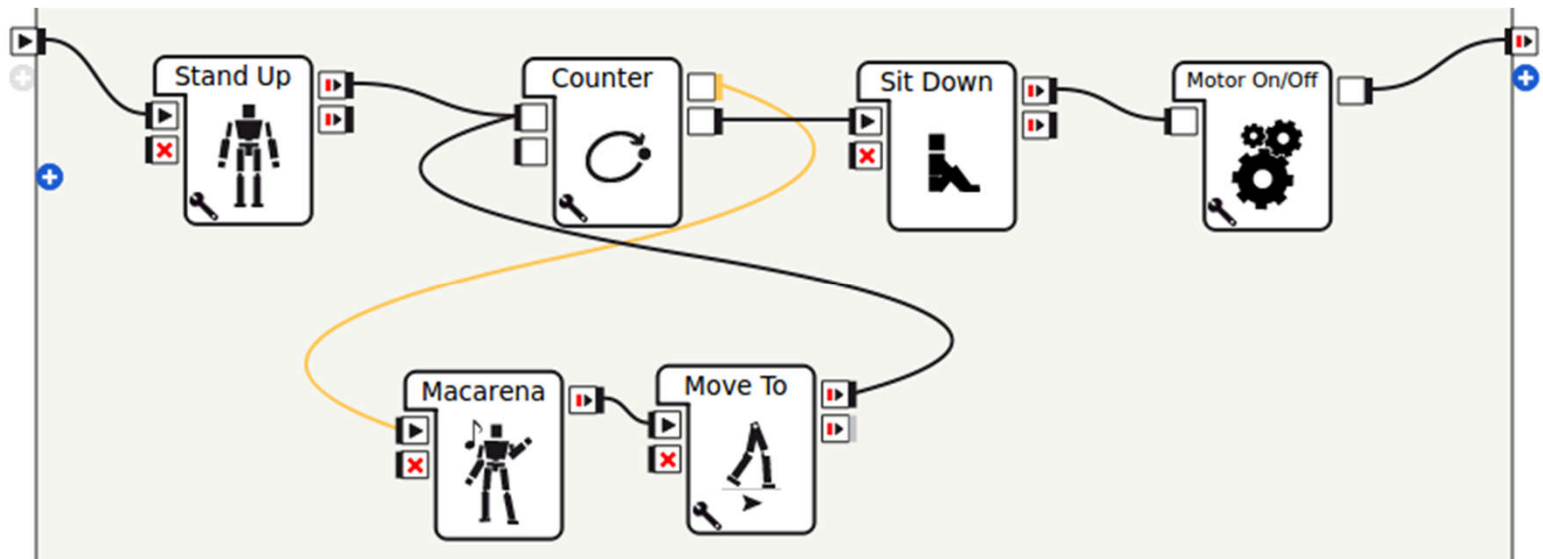
- El bloque “Nod” es de tipo “timeline”
 - Realizar un movimiento de cabeza: subir y bajar
 - No es necesario usar el robot real, se puede usar el robot virtual
- En el bloque “Counter”, escribir 2 en “Final Value”



- Ejecutar el programa

Macarena - Full

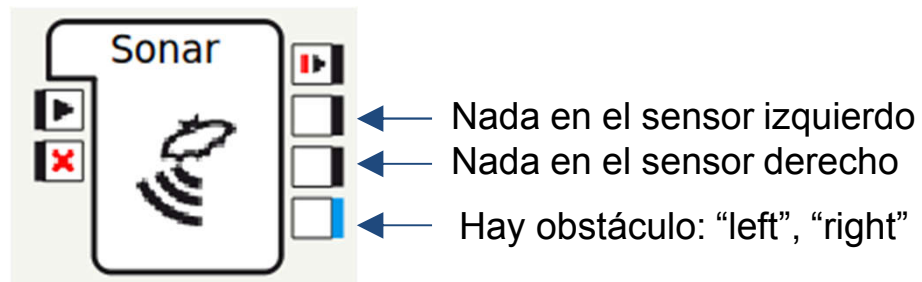
- **Objetivo:**
Repetir el movimiento de Macarena 4 veces, con giro
- Realizar el diagrama (abrir el archivo macarena-full.pml)



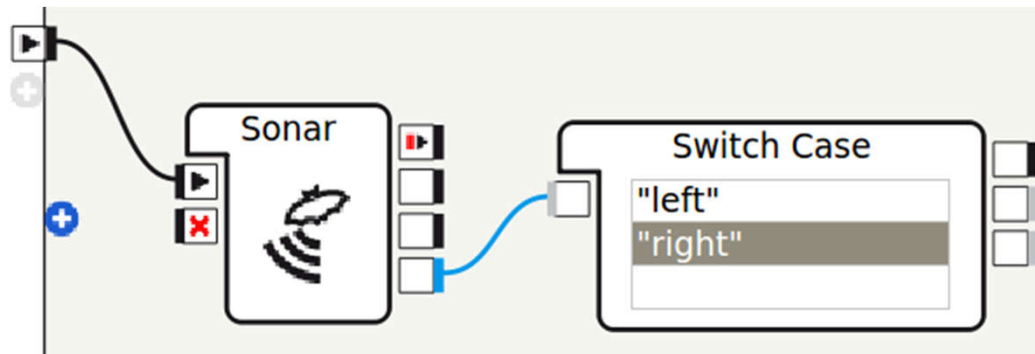
- En “Move To” y “Counter” usar parámetros adecuados
- Al final del movimiento, apagar los motores

Detenerse Cuando hay Obstáculo

- ¿Cómo saber si hay un obstáculo?
 - Una alternativa: con el sensor de ultrasonido
- Sensor de ultrasonido en el NAO:



- Conexión del sensor:



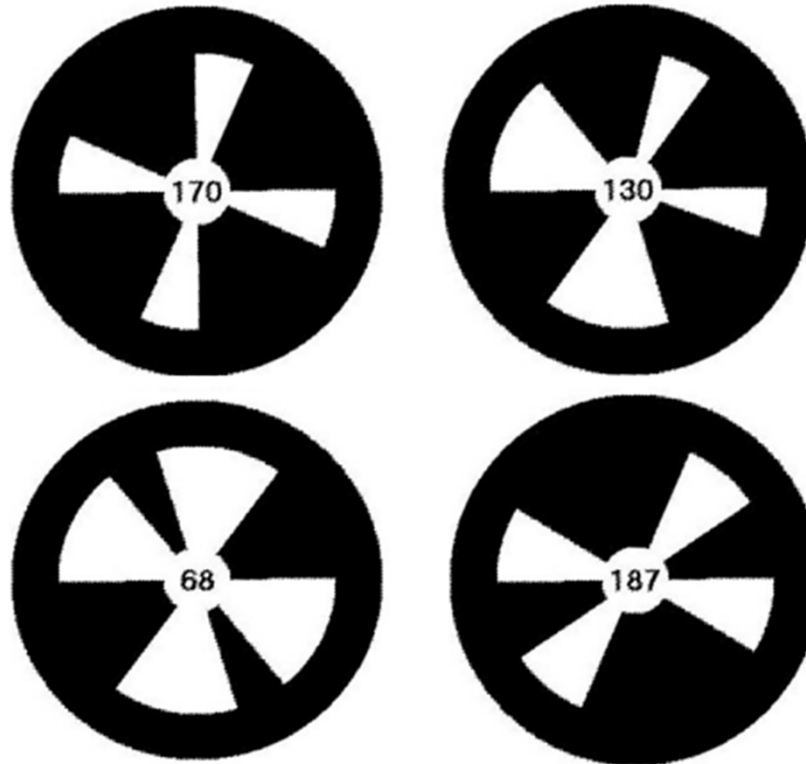
Detenerse Cuando hay Obstáculo

Hacer que el NAO camine indefinidamente (usar el bloque “**Move Toward**”). Cuando el sensor de ultrasonido detecta un obstáculo en frente, el robot se debe detener y debe decir un mensaje que detectó un obstáculo.

Nota: el bloque Move Toward hace que el robot se mueva de manera indefinida en la dirección especificada. Para detenerlo se requiere que su entrada “X” sea activada y que haya un bloque Move Toward que tenga todos los parámetros en cero.

Reconocimiento de marcadores

¿Qué son los marcadores?

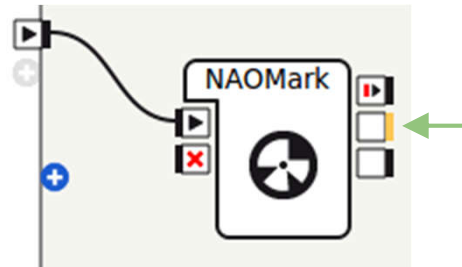


También
llamados
“naomarks”

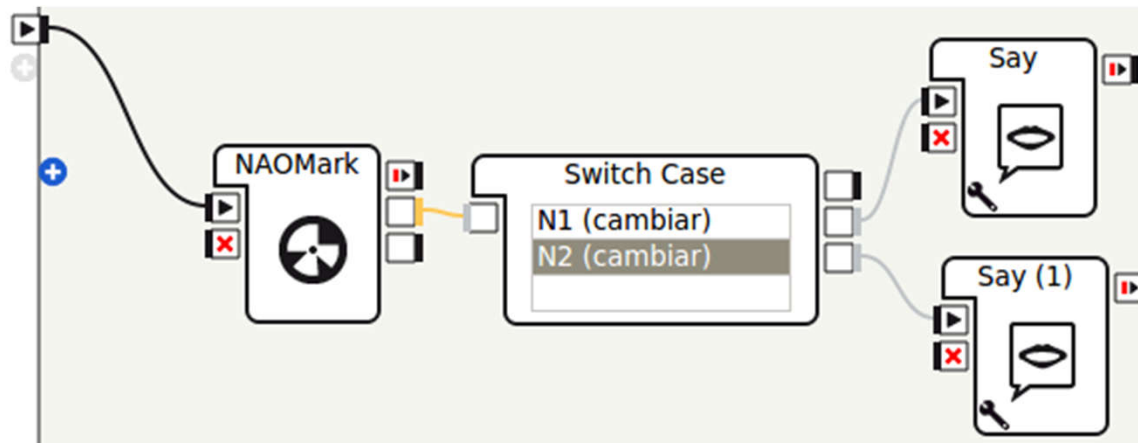
- Son elementos de referencia (“landmarks”) con un patrón específico que el robot puede reconocer (usando su cámara + procesamiento de la imagen/visión computacional)

Reconocimiento de Marcadores

- Conectar un bloque de “NAOMark”, mostrar diferentes marcadores y observar las salidas



- Luego seleccionar algunos marcadores y conectar un bloque “Switch” con diferentes bloques “Say”



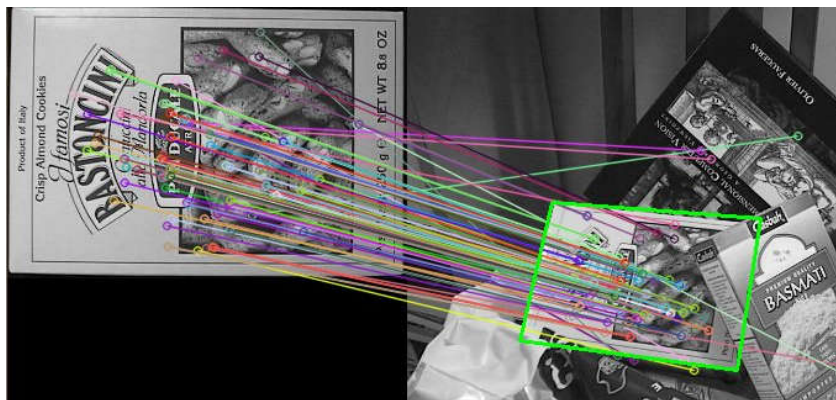
Ejercicios

- Hacer que el robot camine diferente cuando se le muestran diferentes marcadores
- Usar marcadores para hacer para dar las siguientes instrucciones al robot: caminar, detenerse, girar a la derecha, girar a la izquierda (y que diga lo que está haciendo)

Reconocimiento de objetos

Reconocimiento de Objetos

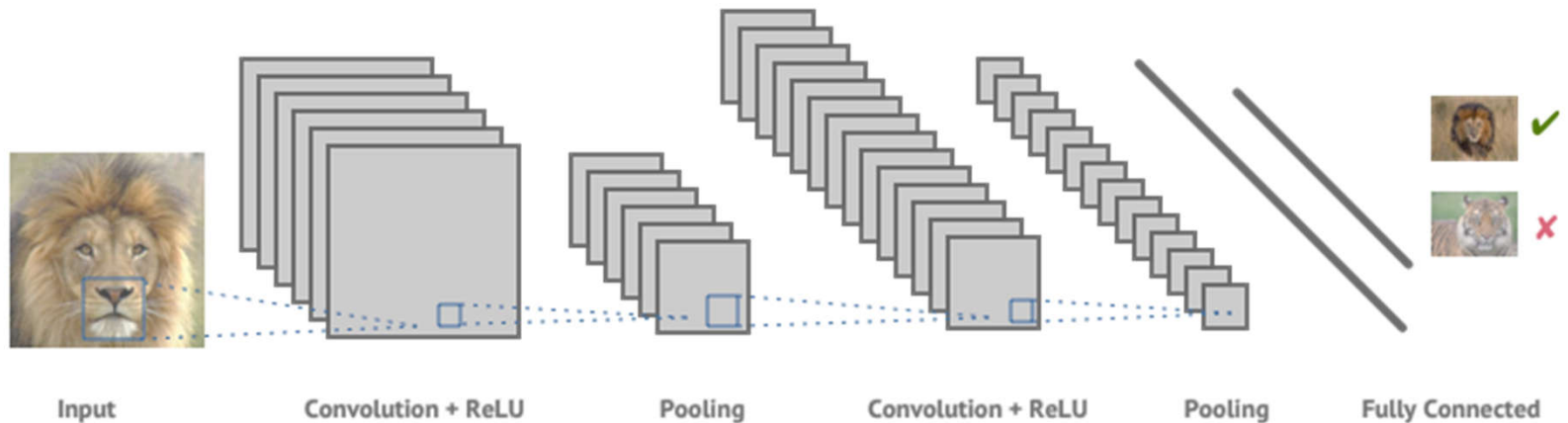
- Para reconocer objetos se requiere una biblioteca (“base de datos”) de objetos conocidos:
 - **Entrenamiento** (aprendizaje, *training*): mostrar un objeto al robot para que lo aprenda
 - **Prueba** (*testing*): mostrar el objeto aprendido al robot para que lo reconozca
- ¿Cómo reconoce el robot?
 - Basado en puntos característicos de la imagen (“features”) que son invariantes (ejemplo: SIFT, SURF, ...)



[Imagen de OpenCV: coldvision.io]

Reconocimiento de Objetos

- ¿Cómo reconoce el robot?
 - Alternativamente se podría usar otros métodos de “deep learning” (ejemplo: redes neuronales convolucionales)



[Imagen de shafeentejani.github.io]

Reconocimiento de Objetos

- Entrenamiento

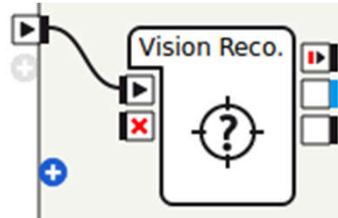
- Ir al monitor de video (“video monitor”)
- Click en “new vision recognition database”



- Click en play (de video monitor) y colocar un objeto frente a la cámara (algo plano con imágenes)
- Click en el botón “learn”
 - Hacer click en los bordes del objeto hasta segmentarlo por completo
 - Ingresar un nombre para el objeto
- Hacer click en el botón “send” para agregarlo a la biblioteca

Reconocimiento de Objetos

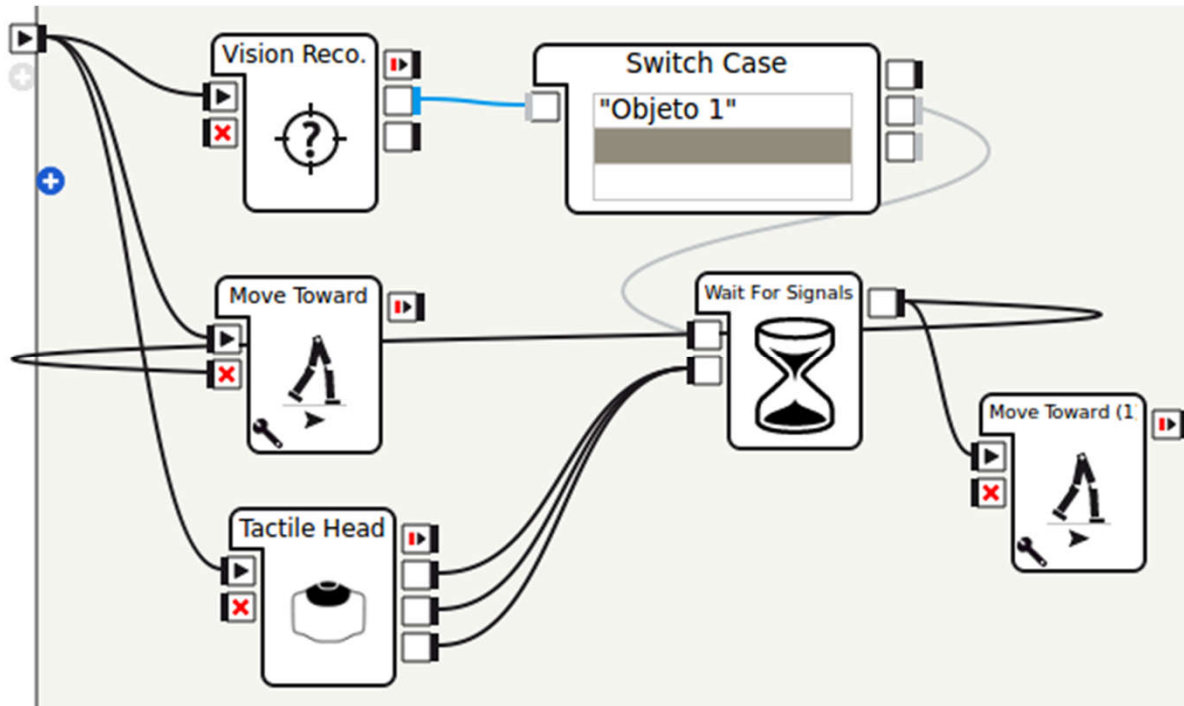
- Prueba
 - Insertar un bloque “Vision Reco”
 - Conectarlo y observar que el objeto sea detectado a su salida al mostrarse el objeto



Reconocimiento de Objetos

- Objetivo

Hacer que el robot camine hasta que vea un objeto que reconoce, y se le toque la cabeza



Ejercicio

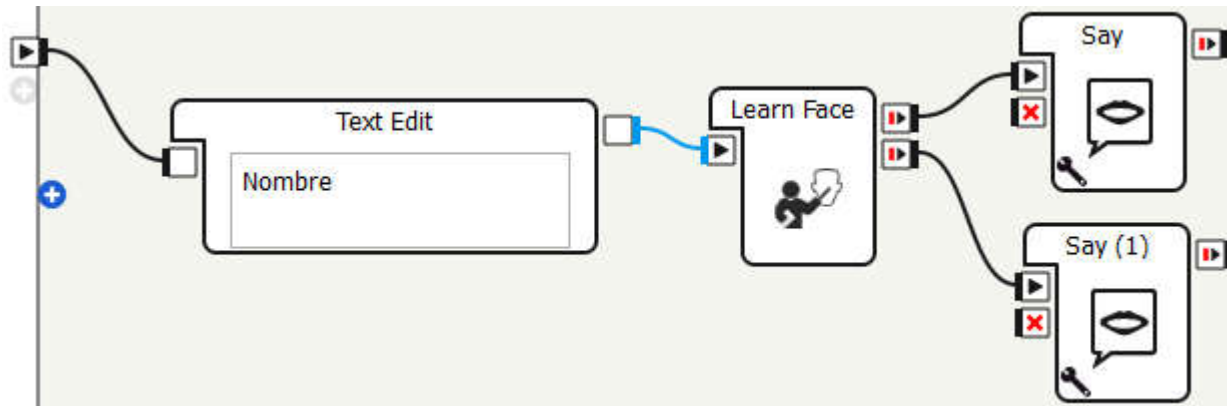
→ Hacer que el robot mueva su cabeza hasta encontrar un objeto determinado

Break

Reconocimiento de Rostros

Reconocimiento de Rostros

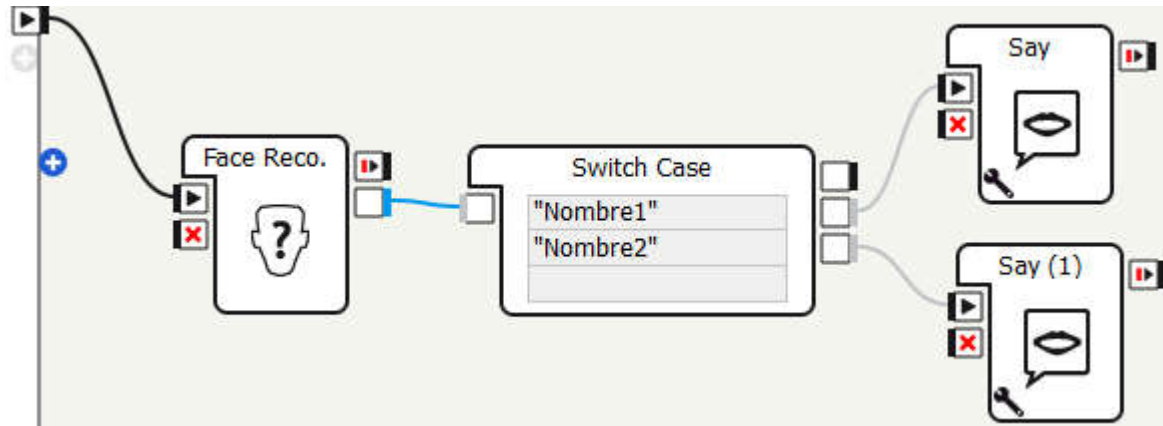
- **Entrenamiento:** Insertar un bloque **Learn Face** (de vision), y **Text Edit** (de Data Edit).
 - Escribir un nombre en “**Text Edit**”
 - En el bloque **Say** superior hacer que el robot diga que aprendió; en el inferior, que hubo un problema



- Luego cambiar y escribir otro nombre y mostrarle otro rostro

Reconocimiento de Rostros

- Prueba de reconocimiento
 - Reemplazar `nombre1` y `nombre2` por los nombres reconocidos
 - El bloque de arriba debe decir que vio a la persona 1, y el de abajo que vio a la persona 2.



Ejercicio

Hacer que el robot reconozca a tres personas y las salude por su nombre