



UNIVERSIDADE DA CORUÑA

# Robobo

Ejercicios con  python™



# Pre-requisitos

Para realizar los ejercicios, recomendamos leer y entender:

- Diapositivas de la asignatura “Robobo: Introducción a la programación de Robobo con Python”
- Wiki de Robobo: Robobo and Python

<https://github.com/mintforpeople/robobo-programming/wiki/python-doc>



# Ejercicio 1

## INTRODUCCIÓN

- Los robots autónomos pueden realizar tareas en las que exploren su entorno, pero deben hacerlo con cuidado para evitar dañarse.
- Pueden encontrarse con obstáculos en su camino o lugares donde podrían sufrir caídas a distinto nivel.
- Aprender a detectar esos peligros y parar a tiempo evitará averías.



# Ejercicio 1

## OBJETIVO DEL EJERCICIO

Crea un programa que haga que Robobo avance hacia delante hasta que detecte un obstáculo:

- A través de los sensores de infrarrojos delanteros, detectará el obstáculo y, a medida que avanza, cambiará el color de los LEDs para indicar la distancia al obstáculo (verde si está lejos, amarillo a distancia media y rojo cuando esté cerca).
- Cuando esté muy cerca del obstáculo, se detendrá para evitar la colisión.



# Ejercicio 1

## SENSORES/ACTUADORES NECESARIOS

- Sensores

- Infrarrojos: `robobo.readIRSensor(IR.FrontC)`

- Actuadores

- LEDs: `robobo.setLedColorTo(LED.All, Color.RED)`
- Ruedas:

```
robobo.moveWheels(speed, speed)
```

```
robobo.stopMotors()
```



# Ejercicio 1

## UN POCO DE AYUDA

- Define variables y ubica su definición al inicio del programa:

```
farIRValue = 30 #Recuerda calibrarlo para tu experimento antes de empezar
```

- Ejemplo de cambio de leds al detectar la distancia establecida como lejos:

```
while (robobo.readIRSensor(IR.FrontC) < farIRValue)and \
      (robobo.readIRSensor(IR.FrontRR) < farIRValue)and \
      (robobo.readIRSensor(IR.FrontLL) < farIRValue):
    robobo.wait(0.01)
```

```
robobo.setLedColorTo(LED.All, Color.MAGENTA)
```

Permite hacer salto de línea

Recuerda importar todos los módulos que utilices en los ejercicios: Color, LED, IR, ...



# Ejercicio 1

## RETO ADICIONAL

Cuando hayas completado el ejercicio, te proponemos este otro reto que utiliza los mismos sensores, pero de una forma un poco distinta:

- Pon a caminar a Robobo sobre una mesa y evita que se caiga. **Realiza este reto con cuidado, si Robobo se cae se puede romper.** Utiliza una **velocidad lenta (speed = 5)** para probar tu programa y mantente atento por si tienes que cogerlo para evitar que se caiga.
- Ten en cuenta que **los sensores infrarrojos Front-L, Front-R, Back-R y Back-L están inclinados hacia el suelo**, por lo que son estos los que deberás utilizar en este nuevo reto.



## Ejercicio 2

### INTRODUCCIÓN

- Los robots autónomos en ocasiones tienen que interactuar con las personas.
- Aunque no será su función principal en el entorno, es importante trabajar un poco la expresividad porque puede que en algunos casos tengan que interactuar personas durante el desarrollo de sus tareas.





## Ejercicio 2

### OBJETIVO DEL EJERCICIO

Basándote en el ejercicio anterior, añade expresión a Robobo de la siguiente forma:

- Cuando se encuentre cerca del obstáculo, haz que se muestre asustado.
- Haz que eche la cabeza hacia atrás y a continuación que retroceda con las ruedas durante unos segundos.
- Para terminar, ahora que Robobo se ha alejado del obstáculo, haz que diga una frase que indique que ya no hay peligro de choque y haz que su cara vuelva a su expresión normal (ya no está asustado).



## Ejercicio 2

### SENSORES/ACTUADORES NECESARIOS

Además de los utilizados en el ejercicio 1, utilizaremos:

- Actuadores

- Síntesis de voz: `robobo.sayText("Ui, casi choco!")`
- Expresiones con la cara: `robobo.setEmotionTo(Emotions.SURPRISED)`
- Expresiones con sonido: `robobo.playSound(Sounds.DISCOMFORT)`
- Ruedas: `robobo.moveWheelsByTime(speed, speed, 2)`
- Mover cabeza: `robobo.moveTiltTo(50, 15)`



# Ejercicio 3

## INTRODUCCIÓN

- Durante las pruebas de los ejercicios con Robobo, puede ser útil parar sus actuadores a través de la captura de un evento en el propio robot, en lugar de tener que ir a pararlos desde el ordenador.



## Ejercicio 3

### OBJETIVO DEL EJERCICIO

Haz un programa que haga que Robobo se mueva indefinidamente:

- Mientras se mueve, haz que diga cada dos segundos “Si me tocas la cara, dejaré de moverme”.
- Cuando Robobo detecte un TAP en la pantalla, haz que se pare.

El objetivo de este ejercicio es aprender a utilizar eventos (aunque otras soluciones son posibles).



## Ejercicio 3

### SENSORES/ACTUADORES NECESARIOS

Además de algunos de los utilizados en los ejercicios 1 y 2, utilizaremos:

- Sensores

- Evento de detección de TAP:

```
robobo.whenATapIsDetected(tapDetectedCallback)
```



## Ejercicio 3

### UN POCO DE AYUDA

- Define una función con la acción que realizará Robobo cuando se detecte el TAP:

```
def tapDetectedCallback():  
    robobo.
```

→ Aquí llamarás al sensor/actuador de Robobo que vayas a utilizar

- Si quieres que Robobo diga algo cada tres segundos, puedes hacer algo como:

```
while True:  
    robobo.wait(3)  
    robobo.sayText("Hola, me llamo Robobo")
```

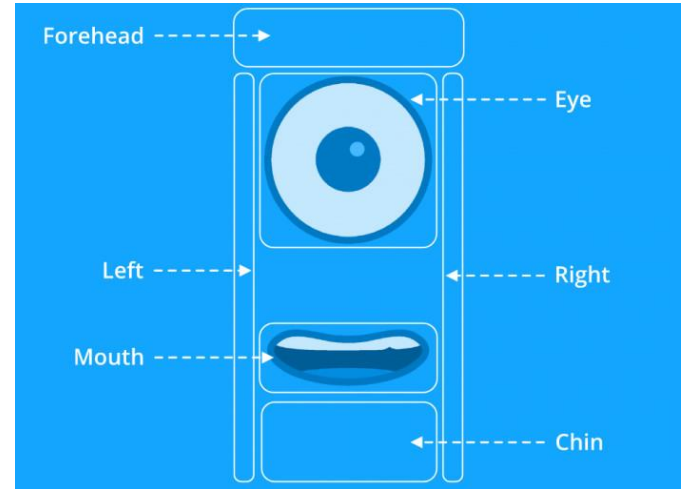


## Ejercicio 3

### RETO ADICIONAL

Cuando hayas completado el ejercicio, te proponemos este otro reto que utiliza los mismos sensores:

- Haz que Robobo se pare cuando le tocas el ojo y que vuelva a moverse cuando le tocas la boca u otra de las posibles zonas que se muestran en la figura.





## Ejercicio 4

### INTRODUCCIÓN

- En los entornos reales existen numerosos objetos con los que el robot debe interactuar y/o tener en cuenta. Para identificarlos, podemos utilizar códigos QR que el robot puede ver a través de una cámara e interpretar.





# Ejercicio 4

## OBJETIVO DEL EJERCICIO

Crea un programa que haga que Robobo avance hacia delante hasta que detecte una señal de tráfico:

- A través de la cámara, Robobo detectará la señal con un código QR y, en función del tipo de señal que detecte, tendrá que realizar distintas acciones.
- Ten en cuenta que deberás ajustar la altura del TILT de Robobo para poder ver el código QR, ya que sólo lo detectará si lo ve completo.





# Ejercicio 4

## SENSORES/ACTUADORES NECESARIOS

Además de algunos de los utilizados en los ejercicios anteriores, utilizaremos:

- Sensores

- Cámara

- Detección de QRs:

```
robobo.whenAQRCodeIsDetected(qrDetectedCallback)
```

- Lectura del QR:

```
robobo.readQR()
```



## Ejercicio 4

### UN POCO DE AYUDA

- `readQR()` devuelve un objeto con distintos atributos. Por ejemplo, para leer la distancia a la que está el QR, haríamos lo siguiente:

```
robobo.readQR().distance
```



# Ejercicio 5

## INTRODUCCIÓN

- El uso de colores en robótica autónoma es una de las formas de etiquetar zonas/elementos. En función del ámbito en el que se utilicen, el significado de los colores puede cambiar. Uno de los ámbitos en los que se suelen utilizar es la prevención de riesgos, dónde a través de colores se pueden indicar distintas zonas de peligro.



# Ejercicio 5

## OBJETIVO DEL EJERCICIO

El objetivo de este ejercicio es familiarizarse con la detección de colores. Crea un programa en el que Robobo se mueva hasta que detecte un objeto de color (configurado previamente):

- Al verlo se parará, dirá la distancia al objeto y la posición del objeto respecto a Robobo: izquierda, derecha o de frente.
- Si no está de frente, Robobo girará hasta situarse frente al objeto de color.
- Para terminar, avanzará recto hasta situarse lo más cerca posible, pero sin llegar a chocar.



## Ejercicio 5

### SENSORES/ACTUADORES NECESARIOS

Además de algunos de los utilizados en los ejercicios anteriores, utilizaremos:

- Sensores: cámara

- Detección de Colores:

```
robobo.whenANewColorBlobIsDetected(blobDetectedCallback)
```

- Lectura de un color: `robobo.readColorBlob(color)`

- Activación de la detección de uno o varios colores:

```
robobo.setActiveBlobs(red, green, blue, custom)
```



# Ejercicio 5

## UN POCO DE AYUDA

- `readColorBlob()` devuelve un objeto con distintos atributos:

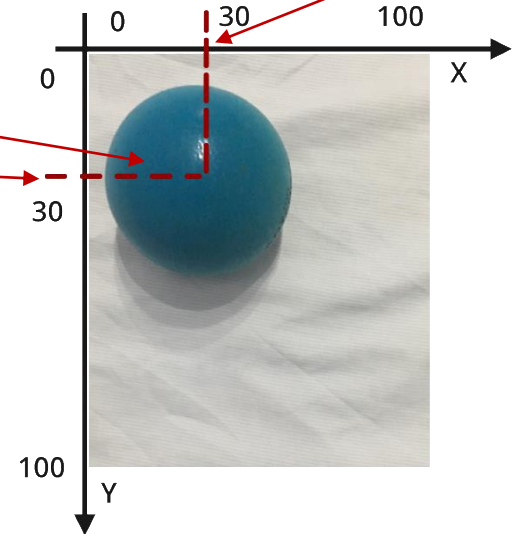
```
robobo.readColorBlob().size
```

```
robobo.readColorBlob().posy
```

- El área del color que ve Robobo es inversamente proporcional a la distancia a la que se encuentra el color.
- Antes de comenzar, calibraremos los colores a través de la configuración de la app Robobo.

```
robobo.readColorBlob().posx
```

Área del color detectado con la cámara



Ejemplo de imagen detectada por la cámara



## Ejercicio 5

### RETO ADICIONAL

Cuando hayas completado el ejercicio, te proponemos este otro reto que utiliza los mismos sensores:

- Haz que Robobo se comporte de forma distinta al ver distintos colores, por ejemplo, que se aleje de un color que identifica como peligro y que, al detectar cualquier otro color, diga el color detectado.

#### UN POCO DE AYUDA

Puedes leer todos los colores con el método: `robobo.readAllColorBlobs()`

*No incluido en las diapositivas de la asignatura, consulta la referencia de Robobo.py:*

<https://mintforpeople.github.io/robobo.py/robobo.html#Robobo.Robobo.readAllColorBlobs>





# Ejercicio 6

## INTRODUCCIÓN

- Una de las funciones de los robots autónomos es el transporte y organización de elementos en un almacén o línea de producción.
- En algunos casos los robots transportan los objetos a un lugar y en otros casos, asisten a otros robots.





# Ejercicio 6

## OBJETIVO DEL EJERCICIO

El objetivo de este ejercicio es que Robobo se acerque a un cilindro de color hasta que llegue a cogerlo con el “pusher” y después pare:

- Utilizaremos el accesorio “pusher” para coger el cilindro y lo fijaremos a la base de Robobo en la parte delantera.
- Para realizar este ejercicio tendréis utilizar los sensores cámara e infrarrojos.

