

Fundamentos de Programación y Robótica

Módulo 2 - Práctico 2

Ejercicio 1 (leer_boton_nxt)

En este ejercicio vamos a realizar un programa que reaccione al presionar el sensor de contacto (sensor tipo botón). Lo que se debe realizar es un programa que ubique la tortuga en la misma posición que está el cursor del ratón cada vez que se presiona el botón.

Es un programa que consiste en una iteración que ejecuta por siempre. Dentro de esta iteración se debe chequear el valor del sensor y si el mismo está presionado, se debe establecer la posición x e y de la tortuga con los valores x e y del cursor del ratón. Además, puede ser conveniente realizar una espera de 0,3 segundos en cada iteración (esté presionado el botón o no).

Algunos de los bloques que deben estar obligatoriamente en la solución son:

- fijar xy
- cursor x
- cursor y

Ejercicio 2 (pista_carrera)

En este ejercicio vamos a construir un programa en donde podamos controlar el trazo de la tortuga con la posición de un motor, de tal forma que se pueda conducir a la tortuga como un auto en el lienzo. En este "auto" la velocidad será constante y el volante será una rueda conectada al motor.

Para esto, se debe realizar un programa que periódicamente (cada 0,3 segundos) establezca el rumbo de la tortuga según la posición del motor y además la haga avanzar hacia adelante una distancia a definir por el usuario (10 pasos son suficientes).

Algunos de los bloques que deben estar obligatoriamente en la solución son:

- posición del motor
- fijar rumbo.
- esperar.

El programa debe ejecutar por siempre y para terminarlo se debe utilizar la opción *Parar* del menú Tortuga.

Ejercicio 3 (pista_carrera_con_freno)

Para este ejercicio nos basaremos en el ejercicio anterior. El comportamiento de la tortuga debe ser el mismo, exceptuando que ahora contará con un freno. Este freno debe implementarse con un botón del kit NXT.

Entonces, si el botón no está presionado el comportamiento es el mismo que para el **Ejercicio 2**. La tortuga avanza a una velocidad constante y su rumbo se determina en base a la posición del motor.

Ahora, si el botón se encuentra presionado la tortuga no debe avanzar. Al soltar el botón, debe comportarse normalmente y avanzar.

La velocidad a utilizar es la misma que para el **Ejercicio 2**.

Algunos de los bloques que deben estar obligatoriamente en la solución son:

- posición del motor

- fijar rumbo.
- esperar.

El programa debe ejecutar por siempre y para terminarlo se debe utilizar la opción *Parar* del menú Tortuga.

Ejercicio 4 (boton_tortuga)

En este ejercicio vamos a trabajar con el sensor de contacto para manipular el avance de la tortuga en la pantalla, tanto para hacerla avanzar como para detenerla.

Para esto, realizaremos un programa que dibuje un espiral desde dentro hacia afuera (como se vió en el **Práctico 1**). El programa contará con una iteración “por siempre” y realizará un paso del espiral únicamente si el botón está presionado.

Si no se presiona el botón no se debe realizar nada. Al presionar el botón por primera vez la tortuga comienza a dibujar el espiral y mientras el botón se mantiene presionado, la tortuga debe continuar dibujando el espiral. Si se suelta el botón, la tortuga debe detenerse.

Algunos de los bloques que deben estar obligatoriamente en la solución son:

- leer
- hablar

Ejercicio 5 (decir_distancia)

El objetivo de este ejercicio es vincular los bloques de la paleta de medios con los sensores y sus valores.

Entonces, vamos a construir un programa en donde la PC nos indique si tiene un objeto cerca, lejos o si la medición se corresponde con el valor máximo (no detecta nada). Para esto se debe utilizar el bloque *hablar* de la paleta de medios.

Se deben determinar franjas de valores para el sensor de distancia para saber si la medida es cercana, lejana o no ve nada. Estos valores dependen del escenario y en consecuencia, debe definirlos usted.

Entonces, debe ejecutarse *para siempre* un programa que lea la distancia medida y evalúe que situación es. Según la situación, debe decir “*Hay un objeto cerca*” o “*Hay un objeto lejos*” o si “*No logro leer nada*”, según el caso. Para conocer la medida máxima del sensor, debe estudiarse el mismo.

Algunos de los bloques que deben estar obligatoriamente en la solución son:

- leer
- hablar

Ejercicio 6 (sensor_motor)

El objetivo de este ejercicio es manipular la velocidad del motor con los valores obtenidos para los sensores de distancia y de grises.

Para esto deben evaluarse los valores máximos y mínimos que devuelve un sensor en el ambiente que se trabaje. Con estos datos y el rango de potencias posibles para el motor, es necesario realizar una operación (como una regla de tres) a la hora de indicarle al motor cual es la potencia con la que debe girar.

Ejercicio 7 (motor_brillo)

El objetivo de este ejercicio es operar un motor según el brillo capturado por la cámara de la computadora.

Entonces, se debe realizar un programa que ponga en marcha el motor con una cierta velocidad determinada por el brillo de la cámara. Cuanto más brillo observe la cámara, más rápido debe ir el motor.

Debe tenerse en cuenta que el motor solo acepta valores entre -127 y 127 y en consecuencia modificar el valor del sensor para que con brillo máximo, devuelva la máxima velocidad y lo opuesto con brillo mínimo. Puede trabajar también entre 0 y 127.

El programa debe ejecutarse por siempre y continuamente establecer la velocidad del motor.

Algunos de los bloques que deben estar obligatoriamente en la solución son:

- iniciar motor
- brillo

Ejercicio 8 (boton_luz)

Para este ejercicio, vamos a utilizar el sensor de grises, el de contacto y el programa desarrollado en el **Ejercicio 4** que dibuja espirales desde dentro hacia afuera.

El funcionamiento general del programa debe ser similar, pero el color del trazo de la tortuga depende del valor leído por el sensor de grises. A menor valor de luz leído, más rojo el color y a mayor valor de luz más azul.

Recomendamos para esto investigar el rango de valores que devuelve el sensor de grises y el rango de valores aceptable por los colores.

Algunos de los bloques que deben estar obligatoriamente en la solución son:

- por siempre
- fijar color
- leer

Nota: Pruebe lo mismo asignando al color del trazo el valor leído por el sensor distancia.

Ejercicio 9 (alarma)

En este ejercicio se debe construir un sistema de alarma rudimentario. Para esto, se utilizarán el sensor de distancia y el sensor de contacto.

El sistema debe implementarse de forma tal que ejecute por siempre. Entonces, al leer el sensor de distancia un valor menor a un umbral dado (por ejemplo, 30) debe reproducirse un tono en el brick del kit lego, que representa la alarma. Esta alarma debe sonar hasta que se presione el botón.

Entonces, el sistema tiene dos estados: la alarma suena o la alarma no suena. Si el estado es "la alarma no suena" se debe consultar el valor del sensor distancia y si éste está por debajo del valor de la alarma, se debe cambiar de estado.

Si el estado es "la alarma suena" se debe reproducir un tono en el brick y leer el valor del botón. Si el mismo está presionado, se debe cambiar de estado.



Algunos de los bloques que deben estar obligatoriamente en la solución son:

- leer
- reproducir tono
- Bloques de variables (para almacenar el estado)

2. *Observación:* Recomendamos repasar el concepto de estado presentado al final del módulo