

Tarea 03

Secuencia de comandos

1. Secuencias en un robot

¿Que es una secuencia?

Una secuencia es un conjunto de instrucciones ordenadas que el robot ejecuta una tras otra.

Hacer secuencias en un robot sirve para ordenar y controlar paso a paso las acciones que debe realizar, de forma lógica y estructurada. En lugar de que el rover haga todo al mismo tiempo o sin control, las secuencias permiten que actúe según un plan, reaccionando a sensores, tiempos o condiciones específicas.

Ejemplo de una secuencia para el rover:

1. Esperar 2 segundos
2. Avanzar derecho durante 5 segundos
3. Detener el rover
4. Girar a la izquierda 90 grados
5. Avanza derecho 3 segundos
6. Detener el rover

2. Problema para el rover

El objetivo de esta tarea es resolver problemas que se apeguen a la vida real mediante secuencias, es decir, plantearemos una situación ficticia donde le daremos al rover pasos a seguir para completar la misión con ayuda del simulador.

Planteamiento del problema

Eres un ingeniero de la NASA que trabaja en el ROVER-PAPIME-25, el robot en el que estan trabajando será enviado a Marte para la exploración de la zona, tu tarea es asegurar que el robot funcione correctamente después del aterrizaje, para lograr eso tu superior quiere la siguiente secuencia implementada en el simulador del robot:

”Después de que el rover aterrice en la superficie marciana, deberá esperar 5 segundos antes de avanzar, simulando la preparación de los motores. A continuación, deberá desplazarse en línea recta durante 5 segundos. Luego, desde el punto donde se detuvo, esperará 2 segundos y girará sobre su propio eje, con el fin de prepararse para la siguiente maniobra. En esta fase, el rover deberá trazar una trayectoria con forma de cuadrado, en la dirección que se prefiera. Finalmente, una vez completado el cuadrado, el rover deberá trazar un círculo hacia el lado opuesto al del recorrido anterior.”

3. Solución del problema

1. Dirigirnos a PAPIME_PE103825/ros2_ws/src/hardware/rover_sequence/rover_sequence

Abrir el archivo rover_sequence.py

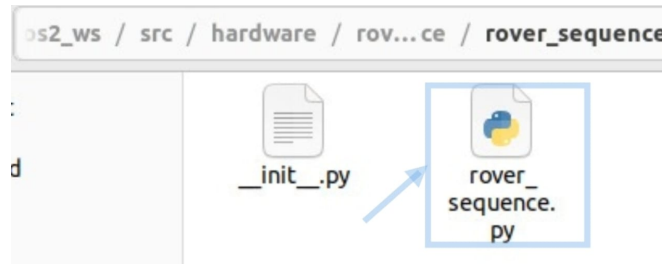


Figura 1: Archivo para editar

2. Copia y pega

Debes de copiar el código que aparece en esta sección, este representa la lista de movimientos para la solución del problema, pegalo en la línea 13 del archivo. **Al estar en python recuerda tener cuidado con la tabulación.**

```
self.sequence = [
    {"x": 0.0, "z": 0.0, "duration": 5.0},
    {"x": 0.0, "z": 0.5, "duration": 5.0},
    {"x": 0.0, "z": 0.0, "duration": 2.0},
    {"x": -0.5, "z": 0.0, "duration": 1.5},
    {"x": 0.0, "z": 0.5, "duration": 10.0},
    {"x": -0.5, "z": 0.0, "duration": 2.5},
    {"x": 0.0, "z": 0.5, "duration": 10.0},
    {"x": -0.5, "z": 0.0, "duration": 2.5},
    {"x": 0.0, "z": 0.5, "duration": 10.0},
    {"x": -0.5, "z": 0.0, "duration": 2.5},
    {"x": 0.0, "z": 0.5, "duration": 10.0},
    {"x": -0.5, "z": 0.0, "duration": 2.5},
    {"x": 0.5, "z": 2.0, "duration": 10.0},
]
```

Nota: 'x' representa el movimiento angular, 'z' define el movimiento lineal y 'duración' es el tiempo de ejecución dado en segundos.

3. Ejecución

- Abrir una terminal (cntrl + alt+ t)
- cd PAPIME_PE103825/ros2_ws
- colcon build
- source install/setup.bash
- ros2 launch rover_sequence rover_launch.py

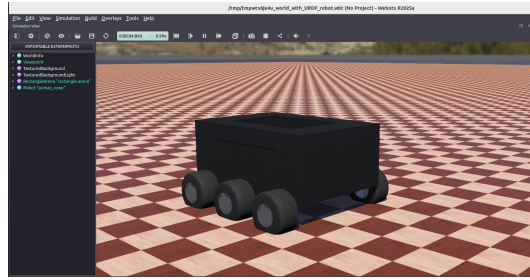


Figura 2: Inicio de la simulación

4. Experimentación

Una vez observado el funcionamiento de la simulación de la secuencia del rover, experimenta creando tus propias combinaciones y probando diferentes valores. Añade o elimina etapas para comprender de manera más profunda cómo opera el sistema.

Tarea: A partir de la secuencia presentada en este documento, realiza la simulación de forma inversa. Es decir, si en la secuencia original el rover avanzaba, ahora deberá retroceder; y si giraba hacia la derecha, deberá hacerlo hacia la izquierda, y viceversa.