1. Overview

The primary goal of this project is to create a ROS node that mimics the behavior of the standard **turtle_teleop_key** node but uses the keyboard characters **‘w’**, **‘a’**, **‘s’**, and **‘d’** for controlling the turtle's movement. The node reads input from the keyboard, translates it into movement commands, and publishes these commands using a **Twist** message.

2. Implementation Details

Node Setup

- **ROS Node Initialization:**
  The node is initialized with rospy.init_node('my_teleop_node') which sets up the node within the ROS network.
- **Publisher Declaration:**
  The node creates a publisher for the **/turtle1/cmd_vel** topic using messages of type **Twist** (imported from geometry_msgs.msg).

Keyboard Input with Curses

- The **curses** module is used for non-blocking keyboard input. The terminal is set to cbreak mode and configured to read keystrokes immediately without waiting for the Enter key.
- The node monitors for key presses such as:
  - **‘w’:** Publishes a Twist message with a positive linear velocity to move the turtle forward.
  - **‘s’:** Publishes a Twist message with a negative linear velocity for moving the turtle backwards.
  - **‘a’:** Publishes a Twist message with a positive angular velocity for turning left (counterclockwise).
  - **‘d’:** Publishes a Twist message with a negative angular velocity for turning right (clockwise).

Command Publishing Loop

- The main loop continuously checks for key presses.
- A new **Twist** message is created for every key press, and the appropriate linear and angular velocities are set.

- The publisher then sends the message on the **/turtle1/cmd_vel** topic.
- The loop runs at a controlled rate (e.g., 10 Hz) to provide smooth movement control.

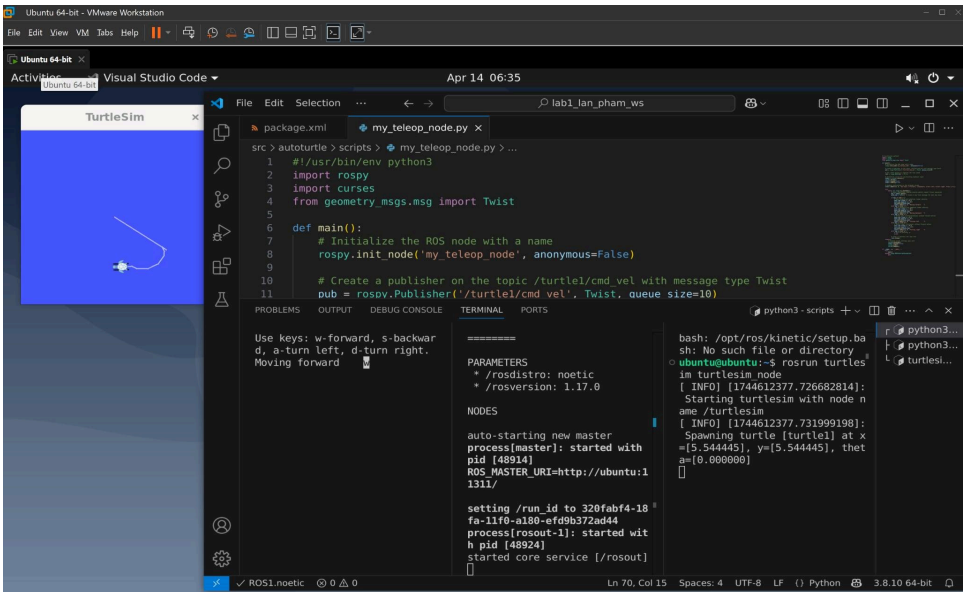
Graceful Shutdown

- The code is designed to clean up the terminal settings by resetting the curses configurations (restoring echo, disabling cbreak mode) when the node is shut down or when the user exits by pressing **'q'**.

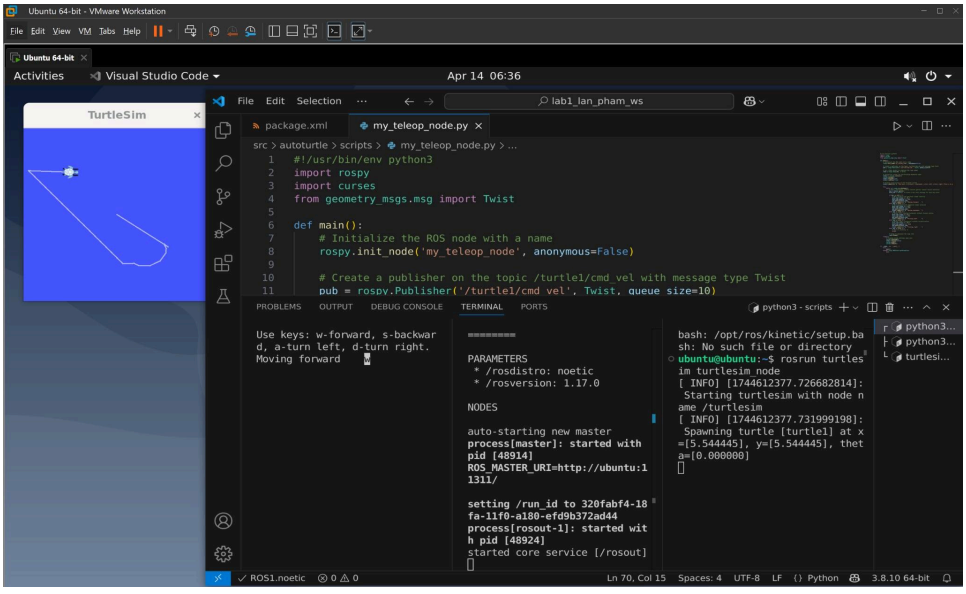
3. Results

The following screenshots capture the trajectories of the turtle under different command inputs:

- **Figure 1:**
  Description: The turtle is observed moving forward as a result of pressing **'w'**.



- **Figure 2:**
  Description: The turtle turns right and moves forward.



- **Figure 3:**

  Description: The turtle moves backward when **'s'** is pressed