**Line Follower Algorithm: Overview**

A **Line Follower Algorithm** is designed for robots, especially line-following bots, to navigate paths and solve mazes by following lines on the ground (usually black lines on a white surface). The primary goal is for the bot to detect and follow the line using sensors, making decisions at intersections or junctions to reach its destination.

These algorithms utilize a set of **IR sensors** (usually three or more) placed at the front of the bot to detect the line and navigate based on sensor readings. The behavior of the bot is determined by predefined rules or priorities, such as whether to turn left or right at junctions.

### Types of Line Follower Algorithms

There are mainly two types of line follower algorithms based on how they prioritize turns at junctions:

**1-Left Prioritized Algorithm**

The Line Follower Bot uses a **Left Prioritized Algorithm**, which is based on inputs from three sensors—left, middle, and right. The bot navigates by detecting a black line on a white surface, making decisions based on sensor readings to move forward, turn, or stop.

#### **Algorithm Cases**:

1. **Middle sensor detects line**: The bot moves forward.
2. **Left sensor detects line**: The bot turns left.
3. **Right sensor detects line**: The bot turns right.
4. **Left and middle sensors detect line**: The bot turns left.
5. **Left and right sensors detect line**:This is condition is invalid and will never arise.
6. **All sensors detect line** (e.g., T-junction):The bot takes a left turn as we are dealing left prioritized algorithm.
7. **No sensor detects line**:The bot makes a left turn until the any sensor detects the line, then moves according to above cases.
8. **Right and middle sensors detect line**: Instead of sudden right turn , the bot moves forward ,if all the sensors detect white surface then bot turns right.

This LFRS algorithm ensures that the bot follows a line and navigates through different junctions by prioritizing left turns, enhancing exploration and maneuverability.

**2-Right Prioritized Algorithm**

The **Line Follower Bot** uses a **Right Prioritized Algorithm**, which is based on inputs from three sensors—**left, middle, and right**. The bot navigates by detecting a black line on a white surface, making decisions based on sensor readings to move forward, turn, or stop. The algorithm prioritizes right turns at junctions to enhance exploration and maneuverability.

#### Algorithm Cases:

1. **Middle sensor detects line**:The bot **moves forward**.
2. **Right sensor detects line**:The bot **turns right**.
3. **Left sensor detects line**:The bot **turns left**.
4. **Right and middle sensors detect line**:The bot **turns right**.
5. **Left and right sensors detect line**:This condition is **invalid** and will never arise.
6. **All sensors detect line (e.g., T-junction)**:The bot takes a **right turn**, as this is a right-prioritized algorithm.
7. **No sensor detects line**:The bot makes a **right turn** until any sensor detects the line, then moves according to the above cases.
8. **Left and middle sensors detect line**:Instead of a sudden left turn, by providing a delay we allow bot to **move forward** and if all sensors detect a white surface, the bot then **turns left**.

This **Right prioritized** algorithm ensures that the bot follows a line and navigates through different junctions by prioritizing **right turns**, enhancing exploration and maneuverability in mazes or track-based challenges.