ARTIFICIAL INTELLIGENCE AND PROGRAMMING ROBOT

PRACTICAL NO 6

027_Abhishek_Ojha

Practical No. 6

Aim:

Write a program to create a path following robot.

Theory:

Light sensor is a transducer used for detecting light and creates a voltage difference equivalent to the light intensity fall on a light sensor.

The two main light sensors used in robots are **Photovoltaic cells and Photo resistor.** Other kind of light sensors like phototransistors, phototubes are rarely used.

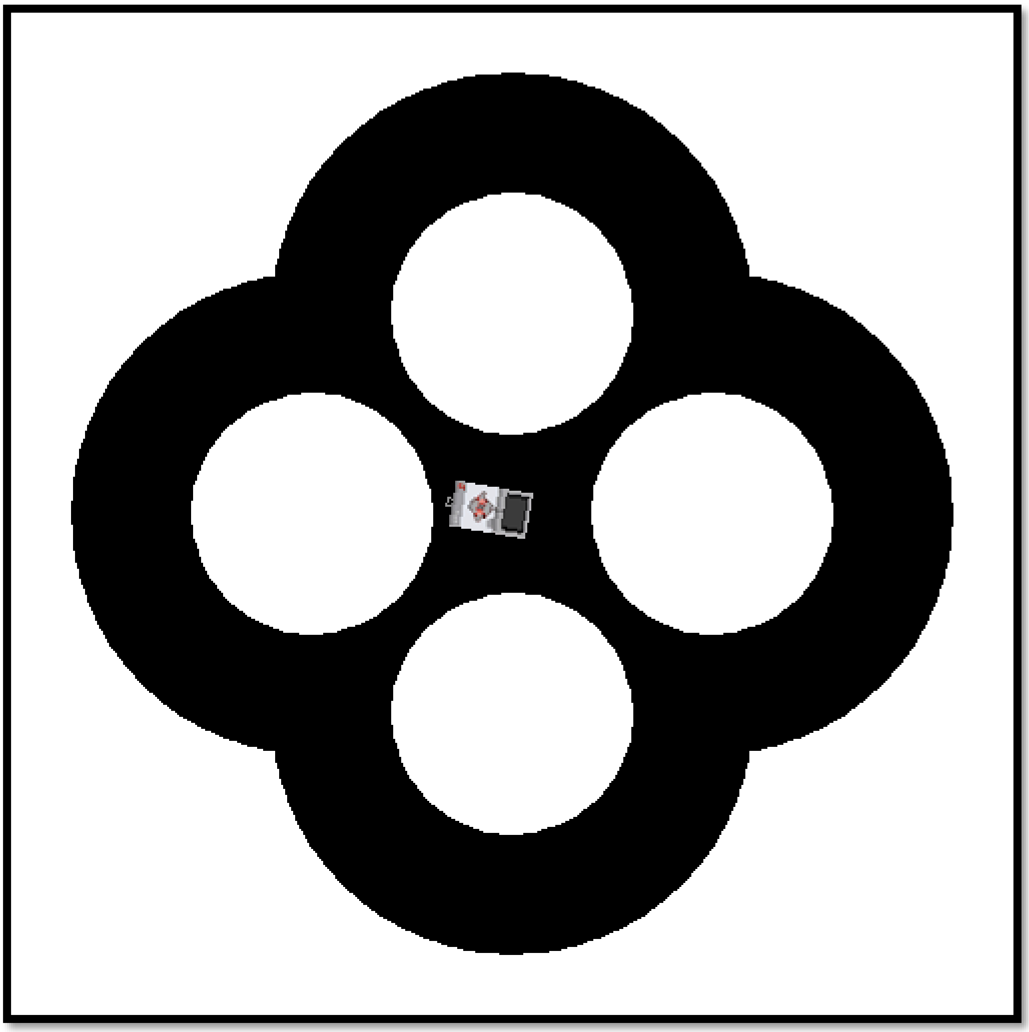
We make a path using game grid where the path is denoted with a dark color and anything outside is not the path, following this assumption we use two light sensor to make the robot stay on the dark path only and turn it towards the dark path when the senor does not sense the path.

Code:

```
package pathfollowingrobot;

import ch.aplu.robotsim.*;
import ch.aplu.jgamegrid.*;
import java.awt.*;

public class PathFollowingRobot {

    static {
        RobotContext.setStartDirection(10);
    }

    public PathFollowingRobot() {

        LegoRobot robot = new LegoRobot();
        Gear gear = new Gear();
        LightSensor ls1 = new LightSensor(SensorPort.S1);
        LightSensor ls2 = new LightSensor(SensorPort.S2);
   try{
            Thread.sleep(5000);
        } catch (InterruptedException ex) {
            ex.printStackTrace();
        }
//      Adding Parts
        robot.addPart(gear);
        robot.addPart(ls1);
        robot.addPart(ls2);
```

```
//     initial movement
        gear.forward();

//     Trigger Level
        int intensity = 500;
//     turning arch
        double arch = 0.1;

        while (true) {
//          get values of light sensors
            int ls1v = ls1.getValue();
            int ls2v = ls2.getValue();

//          Driver Logic for movement
//          if lighsensor1 and lighsensor2 are on track move forward
            if (ls1v < intensity && ls2v < intensity) {
                gear.forward();
            }
//          if lighsensor1 is on track and lighsensor2 is not on track turn right by arch
            if (ls1v < intensity && ls2v > intensity) {
                gear.rightArc(arch);
            }
//          if  lighsensor1 is not on track and lighsensor2 is on track turn left by arch
            if (ls1v > intensity && ls2v < intensity) {
                gear.leftArc(arch);
            }
//          if lighsensor1 and lighsensor2 are not on track move backward
            if (ls1v > intensity && ls2v > intensity) {
                gear.backward();
            }
        }
    }

//    Creating a track
    private static void _init(GameGrid gg) {
        GGBackground bg = gg.getBg();
//    Tracks
        bg.setPaintColor(Color.black);
        bg.fillArc(new Point(250, 150), 120, 0, 360);
        bg.fillArc(new Point(250, 350), 120, 0, 360);
        bg.fillArc(new Point(150, 250), 120, 0, 360);
        bg.fillArc(new Point(350, 250), 120, 0, 360);
//    Gaps
        bg.setPaintColor(Color.white);
        bg.fillArc(new Point(250, 350), 60, 0, 360);
        bg.fillArc(new Point(250, 150), 60, 0, 360);
        bg.fillArc(new Point(150, 250), 60, 0, 360);
        bg.fillArc(new Point(350, 250), 60, 0, 360);
    }

    public static void main(String[] args) {
        new PathFollowingRobot();
    }

}
```

Output:



Conclusion: We successfully made a Lego robot follow a path using light sensors