#include<Servo.h> //using Servo library

#include<NewPing.h> //Ultrasonic library to make response simple, fast and powerful

//L298N Motor driver confuguration

const int RMF=4;  //RightMotorForward

const int RMB=5;  //RightMotorBackward

const int LMB=6;  //LeftMotorBackward

const int LMF=7;  //LeftMotorForward

//Ultrasonic sensor configurations

#define trigPin A1  //Analog Pin

#define echoPin A2  //Analog Pin

#define maximum\_distance 220  //Maximum distance(in cm) we want to measure(optional)(Ultrasonic sensor can measure 2 to 400 cm distance)

boolean goesForward = false;

int distance = 100;

NewPing sonar(trigPin, echoPin, maximum\_distance); //performs distance measurement using sonar

Servo servoMotor;  //My servo name

void setup()

{

  // put your setup code here, to run once:

  pinMode(RMF, OUTPUT);

  pinMode(RMB, OUTPUT);

  pinMode(LMB, OUTPUT);

  pinMode(LMF, OUTPUT);

  servoMotor.attach(10); //attach signal pin of servo to pin 10 of arduino

  servoMotor.write(110);   // 65 //initial position of servo  //set by absorbing positions of servo

  delay(2000);

  distance=readPing();

  delay(100);

  distance=readPing();

  delay(100);

  distance=readPing();

  delay(100);

  distance=readPing();

  delay(100);

  Serial.begin(9600); //start serial communication

}

void loop()

{

  // put your main code here, to run repeatedly:

  int distanceR=0;  //right distance

  int distanceL=0;  //left distance

  delay(50);

  Serial.print("distance = ");

  Serial.println(distance); //prints data on Serial monitor

//Functions calling

  if(distance <= 45)

  {

    moveStop();

    delay(300);

    moveBackward();

    delay(300);

    moveStop();

    delay(300);

    distanceR=lookRight();

    delay(300);

    distanceL=lookLeft();

    delay(300);

    if(distance >= distanceL)

    {

      turnRight();

      moveStop();

    }

    else

    {

      turnLeft();

      moveStop();

    }

  }

  else

  {

    moveForward();

  }

  distance = readPing();

}

//Functions definition

int lookRight()

{

  servoMotor.write(35);

  delay(500);

  int distance = readPing();

  delay(100);

  servoMotor.write(100);

  return distance;

}

int lookLeft()

{

  servoMotor.write(180);

  delay(500);

  int distance = readPing();

  delay(100);

  servoMotor.write(110);

  return distance;

  delay(100);

}

int readPing()

{

  delay(70);

  int cm=sonar.ping\_cm();

  if(cm == 0)

  {

    cm = 250;

  }

  return cm;

}

void moveStop()

{

  digitalWrite(RMF, LOW);

  digitalWrite(LMF, LOW);

  digitalWrite(RMB, LOW);

  digitalWrite(LMB, LOW);

}

void moveForward()

{

  if(!goesForward)

  {

    goesForward=true;

  digitalWrite(RMF, HIGH);

  digitalWrite(LMF, HIGH);

  digitalWrite(RMB, LOW);

  digitalWrite(LMB, LOW);

  }

}

void moveBackward()

{

  goesForward=false;

  digitalWrite(RMB, HIGH);

  digitalWrite(LMB, HIGH);

  digitalWrite(RMF, LOW);

  digitalWrite(LMF, LOW);

}

void turnRight()

{

  digitalWrite(LMF, HIGH);

  digitalWrite(RMB, HIGH);

  digitalWrite(LMB, LOW);

  digitalWrite(RMF, LOW);

  delay(250);

  digitalWrite(LMF, HIGH);

  digitalWrite(RMB, HIGH);

  digitalWrite(LMB, LOW);

  digitalWrite(RMF, LOW);

}

void turnLeft()

{

  digitalWrite(RMF, HIGH);

  digitalWrite(LMB, HIGH);

  digitalWrite(RMB, LOW);

  digitalWrite(LMF, LOW);

  delay(250);

  digitalWrite(RMF, HIGH);

  digitalWrite(LMB, HIGH);

  digitalWrite(RMB, LOW);

  digitalWrite(LMF, LOW);

}