Программа

```cpp
// Setup the servo motor
#include <Servo.h>
Servo myservo;              // инициализировать объект сервопривода

int servposnum = 0;
int servpos = 0;

// Setup Motor A (front and rear) pins
int enableA = 1;
int pinA1 = 3;
int pinA2 = 2;

// Setup Motor B (front and rear) pins
int enableB = 6;
int pinB1 = 5;
int pinB2 = 4;

// Setup Ultrasonic Sensor pins
#define trigPin 8
#define echoPin 9


void setup() {
  // Configure the pin modes for each drive motor
  pinMode (enableA, OUTPUT);
  pinMode (pinA1, OUTPUT);
  pinMode (pinA2, OUTPUT);

  pinMode (enableB, OUTPUT);
  pinMode (pinB1, OUTPUT);
  pinMode (pinB2, OUTPUT);

  // Configure the pin modes for the Ultrasonic Sensor
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
```

```
    // Turn pin into servo driver.
    myservo.attach(7);
}

void loop() {
  // Main code goes here and will run repeatedly:

    car(); // function keeps moving car forward while distance > 25 cm
    avoid(); // function makes car go back, turn slightly right to move forward
in new direction


}

// Create motor functions
void motorAforward() {
digitalWrite (pinA1, HIGH);
digitalWrite (pinA2, LOW);
}
void motorBforward() {
digitalWrite (pinB1, LOW);
digitalWrite (pinB2, HIGH);
}
void motorAbackward() {
digitalWrite (pinA1, LOW);
digitalWrite (pinA2, HIGH);
}
void motorBbackward() {
digitalWrite (pinB1, HIGH);
digitalWrite (pinB2, LOW);
}
void motorAstop() {
digitalWrite (pinA1, HIGH);
digitalWrite (pinA2, HIGH);
}
void motorBstop() {
digitalWrite (pinB1, HIGH);
digitalWrite (pinB2, HIGH);
}
void motorAcoast() {
digitalWrite (pinA1, LOW);
digitalWrite (pinA2, LOW);
}
void motorBcoast() {
digitalWrite (pinB1, LOW);
digitalWrite (pinB2, LOW);
}
void motorAon() {
digitalWrite (enableA, HIGH);
}
void motorBon() {
```

```arduino
digitalWrite (enableB, HIGH);
}
void motorAoff() {
digitalWrite (enableA, LOW);
}
void motorBoff() {
digitalWrite (enableB, LOW);
}

// Setup movement functions
void forward (int duration) {
motorAforward();
motorBforward();
delay (duration);
}
void backward (int duration) {
motorAbackward();
motorBbackward();
delay (duration);
}
void right (int duration) {
motorAbackward();
motorBforward();
delay (duration);
}
void left (int duration) {
motorAforward();
motorBbackward();
delay (duration);
}
void coast (int duration) {
motorAcoast();
motorBcoast();
delay (duration);
}
void breakRobot (int duration) {
motorAstop();
motorBstop();
delay (duration);
}
void disableMotors() {
motorAoff();
motorBoff();
}
void enableMotors() {
motorAon();
motorBon();
}

// Setup Ultrasonic Sensor distance measuring
int distance() {
```

```cpp
  int duration, distance;
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(1000);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  distance = (duration/2) / 29.1;
  return distance;
}

// Setup the main car function
void car() {
int distance_0;
distance_0 = distance();
  // Keep moving forward in a straight line while distance of objects > 25cm
  while(distance_0 > 25)
  {
    // Keep moving servo motor back and forth to scan surroundings
    // This allows the ultrasonic sensor to see more to its left and right
    if(servposnum == 0)
    {
     myservo.writeMicroseconds (1500);   //  задаем среднюю точку, тем самым
останавливая вал сервомотора
     servposnum = 1;
     delay(100);
    }
    else if(servposnum == 1)
    {
     myservo.writeMicroseconds (1800);   // задаем поворот сервопривода по ходу
движения на лево "+300" от средней точки 1500
     servposnum = 2;
    delay(100);
    }
    else if(servposnum  == 2)
    {
     myservo.writeMicroseconds (1500);   //  задаем среднюю точку
     servposnum = 3;
     delay(100);
    }
    else if(servposnum == 3)
    {
     myservo.writeMicroseconds (1200);   // задаем поворот сервопривода по ходу
движения на право "-300" от средней точки 1500
     servposnum = 1;
     delay(100);
    }
    motorAon();
    motorBon();
    forward(1);
    distance_0 = distance();

  }
```

```
    breakRobot(0);


}
void avoid()
{
    // Go back and turn slightly right to move car in new direction if object
detected < 25cm away
    backward(320);
    right(100);
}
```