

```
#define leftCenterSensor 34
```

```
#define leftNearSensor 32
```

```
#define leftFarSensor 30
```

```
#define rightCenterSensor 36
```

```
#define rightNearSensor 38
```

```
#define rightFarSensor 40
```

```
int leftCenterReading;
```

```
int leftNearReading;
```

```
int leftFarReading;
```

```
int rightCenterReading;
```

```
int rightNearReading;
```

```
int rightFarReading;
```

```
int leftNudge;
```

```
int replaystage;
```

```
int rightNudge;
```

```
#define leapTime 200
```

```
#define leftMotor1 2
```

```
#define leftMotor2 3
```

```
#define rightMotor1 5
```

```
#define rightMotor2 4
```

```
#define led 13
```

```
char path[30] = {};
```

```
int pathLength;
```

```
int readLength;
```

```
void setup(){
```

```
    pinMode(leftCenterSensor, INPUT);
```

```
    pinMode(leftNearSensor, INPUT);
```

```
    pinMode(leftFarSensor, INPUT);
```

```
    pinMode(rightCenterSensor, INPUT);
```

```
    pinMode(rightNearSensor, INPUT);
```

```
    pinMode(rightFarSensor, INPUT);
```

```
    pinMode(leftMotor1, OUTPUT);
```

```
    pinMode(leftMotor2, OUTPUT);
```

```
    pinMode(rightMotor1, OUTPUT);
```

```
    pinMode(rightMotor2, OUTPUT);
```

```
    pinMode(led, OUTPUT);
```

```
    digitalWrite(led, HIGH);
```

```
    delay(1000);
```

```
}
```

```
void loop(){
```

```
readSensors();
```

```
if(leftFarReading==1 && rightFarReading==1 &&  
   (leftCenterReading==0 || rightCenterReading==0) ){  
    straight();  
}  
else{  
    leftHandWall();  
}  
  
}
```

```
void readSensors(){  
  
    leftCenterReading = digitalRead(leftCenterSensor);  
    leftNearReading   = digitalRead(leftNearSensor);  
    leftFarReading    = digitalRead(leftFarSensor);  
    rightCenterReading = digitalRead(rightCenterSensor);  
    rightNearReading   = digitalRead(rightNearSensor);  
    rightFarReading    = digitalRead(rightFarSensor);  
  
}
```

```
void leftHandWall(){
```

```
if( leftFarReading==0 && rightFarReading==0){  
    digitalWrite(leftMotor1, HIGH);  
    digitalWrite(leftMotor2, LOW);  
    digitalWrite(rightMotor1, HIGH);  
    digitalWrite(rightMotor2, LOW);  
    delay(leapTime);  
    readSensors();
```

```
  
    if(leftFarReading==0 || rightFarReading==0){  
        done();  
    }  
    if(leftFarReading==1 && rightFarReading==1){  
        turnLeft();  
    }  
}
```

```
  
if(leftFarReading==0){  
    digitalWrite(leftMotor1, HIGH);  
    digitalWrite(leftMotor2, LOW);  
    digitalWrite(rightMotor1, HIGH);  
    digitalWrite(rightMotor2, LOW);  
    delay(leapTime);  
    readSensors();  
  
    if(leftFarReading==1 && rightFarReading==1){
```

```
    turnLeft();  
}  
else{  
    done();  
}  
}
```

```
if(rightFarReading==0){  
    digitalWrite(leftMotor1, HIGH);  
    digitalWrite(leftMotor2, LOW);  
    digitalWrite(rightMotor1, HIGH);  
    digitalWrite(rightMotor2, LOW);  
    delay(30);  
    readSensors();
```

```
    if(leftFarReading==0){  
        delay(leapTime-30);  
        readSensors();
```

```
    if(rightFarReading==0 && leftFarReading==0){  
        done();  
    }  
    else{  
        turnLeft();  
        return;  
    }  
}
```

```

delay(leapTime-30);

readSensors();

if(leftFarReading==1 && leftCenterReading==1 &&
    rightCenterReading==1 && rightFarReading==1){
    turnRight();

    return;
}

path[pathLength]='S';
pathLength++;

if(path[pathLength-2]=='B'){

    shortPath();
}

straight();
}

readSensors();

if(leftFarReading==1 && leftCenterReading==1 && rightCenterReading==1
    && rightFarReading==1 && leftNearReading==1 && rightNearReading==1){
    turnAround();
}

}

void done(){

    digitalWrite(leftMotor1, LOW);

    digitalWrite(leftMotor2, LOW);

    digitalWrite(rightMotor1, LOW);

    digitalWrite(rightMotor2, LOW);

```

```
replaystage=1;

path[pathLength]='D';

pathLength++;
```

```
delay(500);

replay();

}
```

```
void turnLeft(){

    while(digitalRead(rightCenterSensor)==0 | digitalRead(leftCenterSensor)==0){

        digitalWrite(leftMotor1, LOW);

        digitalWrite(leftMotor2, HIGH);

        digitalWrite(rightMotor1, HIGH);

        digitalWrite(rightMotor2, LOW);

        delay(20);

        digitalWrite(leftMotor1, LOW);

        digitalWrite(leftMotor2, LOW);

        digitalWrite(rightMotor1, LOW);

        digitalWrite(rightMotor2, LOW);

        delay(10);

    }
```

```
while(digitalRead(rightCenterSensor)==1){

    digitalWrite(leftMotor1, LOW);

    digitalWrite(leftMotor2, HIGH);

    digitalWrite(rightMotor1, HIGH);

    digitalWrite(rightMotor2, LOW);
```

```
    delay(20);  
    digitalWrite(leftMotor1, LOW);  
    digitalWrite(leftMotor2, LOW);  
    digitalWrite(rightMotor1, LOW);  
    digitalWrite(rightMotor2, LOW);  
    delay(10);  
}
```

```
if(replaystage==0){  
    path[pathLength]='L';  
  
    pathLength++;  
    if(path[pathLength-2]=='B'){  
  
        shortPath();  
    }  
}  
}
```

```
void turnRight(){
```

```
    while(digitalRead(rightCenterSensor)==0){  
        digitalWrite(leftMotor1, HIGH);  
        digitalWrite(leftMotor2, LOW);  
        digitalWrite(rightMotor1, LOW);  
        digitalWrite(rightMotor2, HIGH);
```



```
    delay(20);

    digitalWrite(leftMotor1, LOW);
    digitalWrite(leftMotor2, LOW);
    digitalWrite(rightMotor1, LOW);
    digitalWrite(rightMotor2, LOW);
    delay(10);
}

while(digitalRead(rightCenterSensor)==1){

    digitalWrite(leftMotor1, HIGH);
    digitalWrite(leftMotor2, LOW);
    digitalWrite(rightMotor1, LOW);
    digitalWrite(rightMotor2, HIGH);
    delay(20);

    digitalWrite(leftMotor1, LOW);
    digitalWrite(leftMotor2, LOW);
    digitalWrite(rightMotor1, LOW);
    digitalWrite(rightMotor2, LOW);
    delay(10);
}

while(digitalRead(leftCenterSensor)==1){

    digitalWrite(leftMotor1, HIGH);
    digitalWrite(leftMotor2, LOW);
    digitalWrite(rightMotor1, LOW);
    digitalWrite(rightMotor2, HIGH);
    delay(20);

    digitalWrite(leftMotor1, LOW);
    digitalWrite(leftMotor2, LOW);
```

```
digitalWrite(rightMotor1, LOW);  
digitalWrite(rightMotor2, LOW);  
delay(10);  
}
```

```
if(replaystage==0){  
path[pathLength]='R';  
Serial.println("r");  
pathLength++;  
Serial.print("Path length: ");  
Serial.println(pathLength);  
if(path[pathLength-2]=='B'){  
Serial.println("shortening path");  
shortPath();  
}  
}
```

```
}
```

```
void straight(){  
if( digitalRead(leftCenterSensor)==1){  
digitalWrite(leftMotor1, HIGH);  
digitalWrite(leftMotor2, LOW);  
digitalWrite(rightMotor1, HIGH);  
digitalWrite(rightMotor2, LOW);  
delay(1);  
digitalWrite(leftMotor1, HIGH);
```

```
digitalWrite(leftMotor2, LOW);  
digitalWrite(rightMotor1, LOW);  
digitalWrite(rightMotor2, LOW);  
delay(2);  
return;  
}  
  
if(digitalRead(rightCenterSensor)==1){  
    digitalWrite(leftMotor1, HIGH);  
    digitalWrite(leftMotor2, LOW);  
    digitalWrite(rightMotor1, HIGH);  
    digitalWrite(rightMotor2, LOW);  
    delay(1);  
    digitalWrite(leftMotor1, LOW);  
    digitalWrite(leftMotor2, LOW);  
    digitalWrite(rightMotor1, HIGH);  
    digitalWrite(rightMotor2, LOW);  
    delay(2);  
    return;  
}
```

```
digitalWrite(leftMotor1, HIGH);  
digitalWrite(leftMotor2, LOW);  
digitalWrite(rightMotor1, HIGH);  
digitalWrite(rightMotor2, LOW);  
delay(2);  
digitalWrite(leftMotor1, LOW);  
digitalWrite(leftMotor2, LOW);
```

```
digitalWrite(rightMotor1, LOW);

digitalWrite(rightMotor2, LOW);

delay(1);

}

void turnAround(){

    digitalWrite(leftMotor1, HIGH);

    digitalWrite(leftMotor2, LOW);

    digitalWrite(rightMotor1, HIGH);

    digitalWrite(rightMotor2, LOW);

    delay(150);

    while(digitalRead(leftCenterSensor)==1){

        digitalWrite(leftMotor1, LOW);

        digitalWrite(leftMotor2, HIGH);

        digitalWrite(rightMotor1, HIGH);

        digitalWrite(rightMotor2, LOW);

        delay(2);

        digitalWrite(leftMotor1, LOW);

        digitalWrite(leftMotor2, LOW);

        digitalWrite(rightMotor1, LOW);

        digitalWrite(rightMotor2, LOW);

        delay(1);

    }

    path[pathLength]='B';

    pathLength++;

    straight();
```

```
}
```

```
void shortPath(){
```

```
int shortDone=0;
```

```
if(path[pathLength-3]=='L' && path[pathLength-1]=='R'){
```

```
    pathLength-=3;
```

```
    path[pathLength]='B';
```

```
    //Serial.println("test1");
```

```
    shortDone=1;
```

```
}
```

```
if(path[pathLength-3]=='L' && path[pathLength-1]=='S' && shortDone==0){
```

```
    pathLength-=3;
```

```
    path[pathLength]='R';
```

```
    //Serial.println("test2");
```

```
    shortDone=1;
```

```
}
```

```
if(path[pathLength-3]=='R' && path[pathLength-1]=='L' && shortDone==0){
```

```
    pathLength-=3;
```

```
    path[pathLength]='B';
```

```
    //Serial.println("test3");
```

```
    shortDone=1;
```

```
}
```

```
if(path[pathLength-3]=='S' && path[pathLength-1]=='L' && shortDone==0){  
    pathLength-=3;  
    path[pathLength]='R';  
    //Serial.println("test4");  
    shortDone=1;  
}
```

```
if(path[pathLength-3]=='S' && path[pathLength-1]=='S' && shortDone==0){  
    pathLength-=3;  
    path[pathLength]='B';  
    //Serial.println("test5");  
    shortDone=1;  
}
```

```
if(path[pathLength-3]=='L' && path[pathLength-1]=='L' && shortDone==0){  
    pathLength-=3;  
    path[pathLength]='S';  
    //Serial.println("test6");  
    shortDone=1;  
}
```

```
path[pathLength+1]='D';  
path[pathLength+2]='D';  
pathLength++;  
//Serial.print("Path length: ");  
//Serial.println(pathLength);  
//printPath();  
}
```

```
void printPath(){  
    Serial.println("+++++++");  
    int x;  
    while(x<=pathLength){  
        Serial.println(path[x]);  
        x++;  
    }  
    Serial.println("+++++++");  
}
```

```
void replay(){  
    readSensors();  
    if(leftFarReading==1 && rightFarReading==1){  
        straight();  
    }  
    else{  
        if(path[readLength]=='D'){  
            digitalWrite(leftMotor1, HIGH);  
            digitalWrite(leftMotor2, LOW);  
            digitalWrite(rightMotor1, HIGH);  
            digitalWrite(rightMotor2, LOW);  
            delay(100);  
        }  
    }  
}
```

```
digitalWrite(leftMotor1, LOW);  
digitalWrite(leftMotor2, LOW);  
digitalWrite(rightMotor1, LOW);  
digitalWrite(rightMotor2, LOW);  
    endMotion();  
}  
  
if(path[readLength]=='L'){  
    digitalWrite(leftMotor1, HIGH);  
    digitalWrite(leftMotor2, LOW);  
    digitalWrite(rightMotor1, HIGH);  
    digitalWrite(rightMotor2, LOW);  
    delay(leapTime);  
    turnLeft();  
}  
  
if(path[readLength]=='R'){  
    digitalWrite(leftMotor1, HIGH);  
    digitalWrite(leftMotor2, LOW);  
    digitalWrite(rightMotor1, HIGH);  
    digitalWrite(rightMotor2, LOW);  
    delay(leapTime);  
    turnRight();  
}  
  
if(path[readLength]=='S'){  
    digitalWrite(leftMotor1, HIGH);  
    digitalWrite(leftMotor2, LOW);  
    digitalWrite(rightMotor1, HIGH);  
    digitalWrite(rightMotor2, LOW);
```



```
    delay(leapTime);
```

```
    straight();
```

```
  }
```

```
    readLength++;
```

```
  }
```

```
    replay();
```

```
  }
```

```
void endMotion(){
```

```
    digitalWrite(led, LOW);
```

```
    delay(500);
```

```
    digitalWrite(led, HIGH);
```

```
    delay(200);
```

```
    digitalWrite(led, LOW);
```

```
    delay(200);
```

```
    digitalWrite(led, HIGH);
```

```
    delay(500);
```

```
    endMotion();
```

```
}
```