#pragma config(Sensor, S2, rightBumper, sensorTouch)

#pragma config(Sensor, S3, leftBumper, sensorTouch)

//\*!!Code automatically generated by 'ROBOTC' configuration wizard !!\*//

void stopMotors() {

motor[motorA] = 0;

motor[motorB] = 0;

}

// moveBackwards moves the robot backwards for a given amount of time.

// int power the power of the motors

// int time the amount of time the motors will run.

void moveBackwards(int power, int time) {

motor[motorA] = -1 \* power;

motor[motorB] = -1 \* power;

sleep(time);

}

// turnLeft turns the robot left for a given amount of time.

void turnLeft(int power, int time) {

motor[motorA] = -1 \* power;

motor[motorB] = 1 \* power;

sleep(time);

}

// turnRight turns the robot right for a given amount of time.

void turnRight(int power, int time) {

motor[motorA] = 1 \* power;

motor[motorB] = -1 \* power;

sleep(time);

}

task dontTouchMe() {

while(1)

{

nxtDisplayBigTextLine(0, "LB: %d", SensorValue[leftBumper]);

nxtDisplayBigTextLine(2, "RB: %d", SensorValue[rightBumper]);

if(SensorValue[leftBumper] && SensorValue[rightBumper])

{

moveBackwards(50, 500);

playSound(soundBeepBeep);

if (random(1))

{

turnLeft(50, 500);

} else {

turnRight(50, 500);

}

} else if (SensorValue[leftBumper])

{

moveBackwards(50, 500);

playSound(soundBeepBeep);

turnRight(50, 500);

} else if (SensorValue[rightBumper])

{

moveBackwards(50, 500);

playSound(soundBeepBeep);

turnLeft(50, 500);

}

stopMotors();

}

}

task main()

{

startTask(dontTouchMe);

while(1) {

sleep(1000);

}

}