Learning Roomba: Teacher’s Guide
Module 3 Controlling Movement

Drew Housten (dhousten@gmail.com)
1 Introduction

The first step of working with mobile robots is to make them move in the environment. There are several different ways to control a differential drive robot such as the Roomba. This Module will discuss the types of movement and how to make the Roomba perform those actions. The Module will introduce the movement application programming interface that allows the student to send commands to the Roomba. All of the following Modules build on the ability to make the Roomba move.

2 Educational Merit

Making a robot move is the first opportunity for the students to influence how the robot behaves using their own code. It also is a way to tie some of the mathematical and physics concepts to a tangible, working system. The students will use some of the concepts discussed in the second module to control the Roomba.

3 Topics Covered

The following topics are covered by this Module:

- Why is movement important?
- Types of movement
- How to control the Roomba
- Movement code support
- Exercises

4 Exercise Solutions

Sample programs that meet the requirements for each of the exercises are provided here.

Exercise 1 Program: This program will have the Roomba move in the different possible ways.

```java
import roomba.roombanetwork.services.userservice.*;

public class MyRoombaProgram{
    public static void main(String [] args){
```
UserService.setServerAddress("localhost");
UserService.setName("Your Name");

Roomba roomba = new Roomba();

roomba.drive(1);
roomba.turn(90);
roomba.forwardForTime(.3, 5);
roomba.turnForTime(.3, 3);
roomba.forwardSpeed(.3);
roomba.turnSpeed(.3);
roomba.pause(3);

UserService.disconnect();
System.exit(1);

Exercise 2 Program: This program will have the Roomba go forward 1 meter, turn around, and come back.

import roomba.roombanetwork.services.userservice.*;

public class MyRoombaProgram{

    public static void main(String [] args){
        UserService.setServerAddress("localhost");
        UserService.setName("Your Name");

        Roomba roomba = new Roomba();

        roomba.drive(1);
        roomba.turn(180);
        roomba.drive(1);

        UserService.disconnect();
        System.exit(1);
    }
}

3
Exercise 3 Program: This program will have the Roomba go forward 1 meter, turn left for 5 seconds, and drive backwards for 0.5 meters.

```java
import roomba.roombanetwork.services.userservice.*;

public class MyRoombaProgram{
    public static void main(String [] args){
        UserService.setServerAddress("localhost");
        UserService.setName("Your Name");

        Roomba roomba = new Roomba();
        roomba.drive(1);
        roomba.turnForTime(-.3,5);
        roomba.drive(-0.5);

        UserService.disconnect();
        System.exit(1);
    }
}
```

5 Homework Solutions

The homework for this Module asks the student to write some pseudocode for a program that will make the Roomba drive in a square and return to the original location. Here is an example solution:

1. Initialize new Roomba object
2. Command the roomba to go forward 1 meter
3. Command the roomba to turn right 90 degrees
4. Command the roomba to go forward 1 meter
5. Command the roomba to turn right 90 degrees
6. Command the roomba to go forward 1 meter
7. Command the roomba to turn right 90 degrees
8. Command the roomba to go forward 1 meter
9. Disconnect from the roomba
6 Related Resources


Chapter 5 of *Hacking Roomba: ExtremeTech* covers driving the Roomba and the different ways that the Roomba can be controlled. It is an excellent resource to use in parallel with the materials presented in this Module.


Some of the associated exercises to the *Robotics Primer* are related to this Module. If more experience is needed or if a different approach is desired, looking at the supported materials would be helpful.