

# Robots in the Classroom

## Table Of Contents

---

### **Chapter 1: What is a Program? \_\_\_\_\_ 1**

- 1.1 Computer Languages
- 1.2 RobotBASIC
- 1.3 The Output or Terminal Screen
- 1.4 Drawing Lines
- 1.5 Running Programs
- 1.6 Saving and Retrieving Programs
- 1.7 Errors in a Program
- 1.8 Adding to the Program
- 1.9 Connecting Lines (the easy way)
- 1.10 The Robot Simulator
- 1.11 Summary
- 1.12 Exercises

### **Chapter 2: Variables \_\_\_\_\_ 13**

- 2.1 What is a Variable?
- 2.2 Case Sensitivity
- 2.3 Rectangles
- 2.4 Using Variables
- 2.5 User Input
- 2.6 Circles and Ellipses
- 2.7 Using the Robot
- 2.8 Exercises

### **Chapter 3: Loops \_\_\_\_\_ 23**

- 3.1 Efficiency
- 3.2 The WHILE-WEND Structure
- 3.3 The REPEAT-UNTIL Structure
- 3.4 A Counting-Structure (FOR-NEXT)
- 3.5 Implementing the FOR-NEXT Structure
- 3.6 Implementing a WHILE-WEND Loop
- 3.7 Implementing a REPEAT-UNTIL Loop
- 3.8 Summary
- 3.9 Exercises

## **Chapter 4: Decisions** \_\_\_\_\_ **35**

- 4.1 Bumper Switches
- 4.2 The IF-THEN Structure
- 4.3 The IF-ENDIF Structure
- 4.4 The IF-ELSE-ENDIF Structure
- 4.5 The IF-ELSEIF Structure
- 4.6 Designing a Program
- 4.7 Explaining the Program
- 4.8 Debugging the Program
- 4.9 Improving the Program
- 4.10 Adding Randomness
- 4.11 Summary
- 4.12 Exercises

## **Chapter 5: Modules** \_\_\_\_\_ **49**

- 5.1 Subroutines and Labels
- 5.2 The Gosub Command
- 5.3 Advantages of Modular Programming
- 5.4 Organizing a Program
- 5.5 Analyzing the Program
- 5.6 The HELP System
- 5.7 Summary
- 5.8 Exercises

## **Chapter 6: Using the Mouse** \_\_\_\_\_ **57**

- 6.1 Moving Objects with the Mouse
- 6.2 The Main Program
- 6.3 The Power of Modularity
- 6.4 Module Definitions
- 6.5 Finding if the Mouse is Over an Object
- 6.6 Drawing and Erasing Objects
- 6.7 Principles of Animation
- 6.8 Summary
- 6.9 Exercises

## **Chapter 7: Following a Line** \_\_\_\_\_ **67**

- 7.1 Line Following
- 7.2 Line Sensors
- 7.3 Binary Numbers
- 7.4 The rSense() Function
- 7.5 Developing an Algorithm
- 7.6 A Simple Approach
- 7.7 Another Debugging Tool
- 7.8 Making the Robot Remember
- 7.9 The Value of Simulation
- 7.10 Finding the Line
- 7.11 Compromises
- 7.12 Summary
- 7.13 Exercises

## **Chapter 8: Finding an Object** \_\_\_\_\_ **83**

- 8.1 Vision Commands
- 8.2 The rLook() Function
- 8.3 The rBeacon() Function
- 8.4 Using Simulated Sensors
- 8.5 Finding an Object
- 8.6 Moving to the Object After it is Found
- 8.7 Knowing the Object Has Been Found
- 8.8 Printing Anywhere You Want
- 8.9 Testing the Program
- 8.10 Enhancing the Program
- 8.11 Summary
- 8.12 Exercises

## **Chapter 9: Navigating a Maze** \_\_\_\_\_ **93**

- 9.1 Creating a Maze
- 9.2 Moving Into the Maze
- 9.3 The rFeel() Function
- 9.4 Proceeding Through the Maze
- 9.5 Improving the Program
- 9.6 Array Variables
- 9.7 A Practical Example
- 9.8 Real World Robots
- 9.9 Correcting for Wheel Slip
- 9.10 Summary
- 9.11 Exercises

**Chapter 10: Beacon Navigation** 107

- 10.1 Celestial Navigation
- 10.2 What is a Beacon
- 10.3 Math is Essential
- 10.4 Using the Derived Equations
- 10.5 Your Future in Robotics
- 10.6 Summary
- 10.7 Exercises

**Chapter 11: Robotic Arms** 117

- 11.1 Real Robotic Arms
- 11.2 Choosing the Motors
- 11.3 Joint Options
- 11.4 End-effectors and Grippers
- 11.5 Arm Mathematics
- 11.6 A Simulated Arm
- 11.7 Programming the Arm
- 11.8 Summary
- 11.9 Exercises

**Chapter 12: Walking Robots** 133

- 12.1 Programming with Poses
- 12.2 Minimal Requirements
- 12.3 Compromises
- 12.4 Sensors
- 12.5 Controlling the Joints
- 12.6 The DrawRobot Subroutine
- 12.7 Moving Several Joints Simultaneously
- 12.8 The Robot Can Fall
- 12.9 Finding the Balance Point
- 12.10 Autonomous Decisions
- 12.11 Maintaining Balance
- 12.12 A YouTube Video
- 12.13 Summary
- 12.14 Exercises

**Appendix A: Solutions to Exercises** 145

**Index** 166