Learning Computing With Robots

in C++

Edited by

Deepak Kumar

This version was produced by Bruce MacLennan in January 2011 based on the Fall '08 Python text, revised June 2009.

Contributors

IPRE was created to explore the use of personal robots in education with primary funding from Microsoft Research, Georgia Institute of Technology, and Bryn Mawr College. This text would not have been possible without the hardware, software, and course materials developed by members of the IPRE team.

Ben Axelrod, Georgia Institute of Technology
Tucker Balch, Georgia Institute of Technology
Douglas Blank, Bryn Mawr College
Natasha Eilbert, Bryn Mawr College
Ashley Gavin, Bryn Mawr College
Gaurav Gupta, Georgia Institute of Technology
Mansi Gupta, Bryn Mawr College
Mark Guzdial, Georgia Institute of Technology
Jared Jackson, Microsoft Research
Ben Johnson, Georgia Institute of Technology
Deepak Kumar, Bryn Mawr College
Bruce MacLennan, University of Tennessee, Knoxville
Marwa Nur Muhammad, Bryn Mawr College
Keith O'Hara, Georgia Institute of Technology
Shikha Prashad, Bryn Mawr College
Richard Roberts, Georgia Institute of Technology
Jay Summet, Georgia Institute of Technology
Monica Sweat, Georgia Institute of Technology
Stewart Tansley, Microsoft Research
Daniel Walker, Georgia Institute of Technology

ii

Contents

Preface

What is a computer? A personal computer?
What is a robot? A personal robot?
What is computing? What is personal computing?

This book will introduce you to the world of computers, robots, and computing. Computers, as we know them today have been around less than 75 years. To our young audience for whom this book is designed, that may seem like an eternity. While computers are all around us, the world of computing is just in its infancy. It is a baby!

In this book, you will learn that computing is no more about computers than astronomy is about telescopes. In that respect, the roots of computing could be traced back more than 2000 years to Euclid in 300 BC. Could it be that computing is actually as old as or even older than some of the other disciplines? Prior to the development of modern computers, the word *computing* mostly implied numerical calculation. Today, computing encompasses all forms of electronic manipulation of objects: numbers, text, sound, music, photographs, movies, genes, motors, etc., and even ideas! Every day we are discovering new things that can be computed.

Robots are not that much older than computers. We will learn that, contrary to popular belief, a robot is any mechanism that is guided by automated control.

A robot can be controlled as a result of computation. In fact, in today's robots computation is a key component.

Advances in the past few decades have brought about many changes in the way we use and live with technology. Computers have become smaller, cheaper, faster, and more personal. The same is also true for robots. In this book we have created a confluence of these two technologies to use as a basis for introducing the world of computing to students.

The word *personal* is key to this new confluence. Computers and robots are no longer the realm of large corporate offices and industrial manufacturing scenarios. They have become personal in many ways: they help you write your term papers, store and organize your photographs, your music collection, your recipes, and keep you in touch with your friends and family. Many people might even dismiss the computer as an appliance: like a toaster oven, or a car. However, as you will learn in this book, a computer is far more versatile than most appliances. It can be used as an extension of your mind. We were not kidding above when we mentioned that computers can be used to process ideas. It is in this form, that a computer becomes an enormously powerful device regardless of your field of interest. How personal you can make a computer or a robot is entirely up to you and your abilities to control these devices. This is where this book comes in. You will learn the basics of how a computer and a robot is controlled and how you can use these ideas to personalize your computers even further, limited only by your imagination.

How to use this book

We have strived in creating this book to make the learning experience engaging, entertaining, and enlightening. This book comes with a robot kit and you will make extensive use of your personal robot as you go through the book. As a part of your learning process, you will be bringing all of the material in this book to life in the form of computer programs and robot behaviors. You will create new and interesting behaviors for your robots. In order to get the most out of this book, you will need the following:

- This book.
- A robot. This book uses the Scribbler robot.
- A Fluke Lancet dongle.
- A Bluetooth enabled computer. If your computer does not have built-in Bluetooth, you can purchase a USB Bluetooth dongle to add that functionality. You can use any of the following operating systems: MACOS (Version 10.4 or later), Windows (XP or Vista), or Linux.
- Software: the C++ version of Myro. This book is based on C++, but other editions use other languages, such as Python.
- Some accessories: A basic gamepad controller, Sharpie permanent color markers, a flashlight, spare AA batteries, poster or any other large drawing paper, a carrying case, some craft materials for decorating the robot, etc.

You can order the robot, the Fluke dongle, the Bluetooth dongle, as well as a software CD, and the Python-based book from Georgia Robotics Inc. (georgiarobotics.com). For more information on additional sources, documentation (including a copy of this book), and latest updates visit the IPRE web site at roboteducation.org.

While our approach is tightly integrated with the use of a personal robot we will also take frequent excursions into other forms of computing: numbers, text, sounds, music, images, etc. The thing to remember is that all the concepts of computing introduced in this book can be used to do computing on all kinds of objects.

To get the most out of this book we would recommend reading each chapter first and then bringing all the ideas discussed in the chapter to life by actually creating computer programs and robot behaviors. Before proceeding to the next chapter, try and do most of the exercises at the end of each chapter and think about further applications that come to mind. Try out a few of those and then proceed to the next chapter. We would also like to encourage you to work in an environment where you can share your work and ideas with others around you. After all, you are going to be creating several robot behaviors. Find the opportunity to demonstrate your robot's newly programmed

capabilities to your friends, classmates, and even the wider world (through web sites like youtube.com). Make learning from this book a social activity and you will find yourself learning more. The bottom-line keyword here is *doing*. Computing is an abstract concept, but the only way to grasp it is by *doing it*. Like that famous Nike commercial, *Just do it!*

What you will learn

By using this book as described above you will learn a number of fundamental concepts in computing and robotics. We have written the text for entry-level students into computing. Other than being able to use a computer to do something, we do not expect our readers to have any additional background in computing. Some of the material in the book does rely on basic high school level mathematics. It is ideally suited for a college-level introductory computing course which is widely recognized as CS1 in the computing education community. By the time a reader works through the entire book they will have covered the following topics:

- Fundamentals of computing: problem solving, program design, basic syntax and semantics of the programming language C++; the role of names; data values: numbers, strings, and booleans; data structures: vectors, lists, structures, dictionaries, objects; functions, parameters, basic scoping rules, assignment, expressions, control (sequential, conditional and iteration constructs), modules and libraries (APIs); Simple input-output, file I/O, and other forms of I/O devices; testing and error checking; interactive design.
- Program development process: using an IDE, interactive style computing, debugging, and documentation.
- Basic robotics: robot sensing and control; sensor types, synchronous and asynchronous operations; control paradigms: direct or reactive, Braitenberg style, subsumption style, neural network; behavior design; real-time/embedded control.
- Media computation: numbers, sound and music, image: generation, processing, and understanding; gamepad controllers and game playing.
- Social implications of computing and robotics.

- Computation: history and essentials of algorithms; efficiency; limits of computing; other issues and open problems in computing.

For instructors, it should be clear that the above set of concepts provide sufficient coverage of the CS1 topics recommended by the latest ACM/IEEE Curricular guidelines[1] as well as the guidelines developed for liberal arts degrees in computer science[2]. In many respects the content of the book goes well beyond the recommended CS1 guidelines. It does not explicitly mention recursion or sorting algorithms. We have deliberately chosen to omit these topics. Sorting of lists and other sequential structures is available in standard C++ libraries and students will use these to do sorting. We often find that students discover recursion in the course of doing exercises from the text. Rather than provide a separate chapter or a section on recursion, we have left it as a topic to be introduced by the instructor if she/he so chooses. We do use recursion is example programs in Chapter 10. It would also be fairly trivial for instructors to include class discussions of sorting algorithms if they feel so inclined.

Further, for instructors, the coverage and sequence of topics in this book will vastly differ from that found in most traditional CS1 texts. We have deliberately taken a *contextual approach* to introduce ideas. The nature of the context, *robots and computers*, has driven the nature, ordering, and sequencing of topics. We have found that this is a more engaging, exciting, and enlightening framework. We hope that you will be able to share this excitement while you are teaching a course based on this material. Further details on how to use this book and additional materials is provided in the Instructor's manual accompanying this text. Much of this work is an outcome of research conducted under the banner of IPRE specifically for developing a fresh new approach to CS1. We invite you to refer to our project web site (roboteducation.org) and read some of the papers and articles that elaborate much more on the underlying research and its goals.

---

[1] Computing Curricula 2001, ACM Journal on Educational Resources in Computing (JERIC) Vol. 1 , No. 3, Fall 2001.

[2] A Model Curriculum for a Liberal Arts Degree in Computer Science, LACS, ACM JERIC, Vol. 7, No. 2, June 2007.

To students, we have taken efforts to make the material engaging and exciting for you. We hope that we have succeeded somewhat in this. This is an open source development project, so we hope that you will take the time to provide valuable feedback, as well as ideas and content for further improving this book. We do hope that you will exploit all the fun learning opportunities provided in this book.