

Learning Java Programming in Clara's World

Anton Bogdanovych • Tomas Trescak

Learning Java Programming in Clara's World

 Springer

Anton Bogdanovych
School of Computer, Data
and Mathematical Sciences
Western Sydney University (WSU)
Parramatta, NSW, Australia

Tomas Trescak
School of Computer, Data
and Mathematical Sciences
Western Sydney University (WSU)
Parramatta, NSW, Australia

ISBN 978-3-030-75541-6 ISBN 978-3-030-75542-3 (eBook)
<https://doi.org/10.1007/978-3-030-75542-3>

© Springer Nature Switzerland AG 2021

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors, and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Cover illustration: The book cover image features the “Leaf” 3D model by Thunk3D scanner available under Creative Commons CC BY 4.0.

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

*With gratitude to our families
for encouraging us to complete this work
and providing comfort and support,
while receiving much less attention than they deserve.*

Preface

Welcome to *Learning Java Programming in Clara's World*. This book is intended for a one-semester university or an equivalent high school programming course. It is equally suitable for just about anyone else who wants to start learning programming. While this text is written for students with little or no prior programming experience, experienced programmers would also benefit from the in-depth explanations, visual teaching style of this book and interesting programming challenges supplied with it.

Why Did We Write This Book?

The story of this book began ten years ago from overhearing a group of computer science students loudly talking about how much they hate programming. Hearing something like this while being deeply in love with programming made a significant impact on us and resulted in a strong feeling that something had to be done about this. Back then we were two young university academics with background in video games development and virtual reality with no involvement or experience in teaching programming. Programming was our passion and the key instrument in our daily lives. We felt a strong urge to share this passion and explain why programming is such an important tool in the twenty-first century.

After months of research into motivational techniques and existing novel methods of teaching programming, we have produced a proposal for teaching programming fundamentals that relied on similar principles (such as flow [1]) that game developers use for making players attracted to their games. We are grateful to our dean, Simeon Simoff, for trusting our passion and expertise and for allowing us to implement this proposal at Western Sydney University. Five years later, we received two national and one international awards for the teaching methodology and the visual programming framework that forms the basis for this book. The most important achievement, however, is that we have not heard students expressing their hate of programming ever since we started teaching it at our university.

Teaching Programming in a Fun Way

In this book, we are introducing the key fundamental concepts of Java programming such as loops, conditional statements, variables, expressions, methods and arrays through the eyes of a small ladybug called Clara. Clara is a fun and extremely obedient insect, who initially has very limited intelligence. Readers learn programming by making Clara move around and manipulate objects in her world. As the book progresses, Clara becomes smarter and smarter and develops her sense of touch and vision. She progressively acquires new programming skills and (together with readers) learns by tackling some of the world's greatest challenges.

Unlike many other programming textbooks that rely on solving toy artificial problems, this book explains programming concepts through real-world problems such as developing a vacuum cleaner robot, automatically patching potholes, launching rockets into space, simulating projectile motion, dynamically avoiding obstacles, delivering mail, etc. Every chapter of the book starts by presenting a challenge and then continues to explain new programming concepts with the focus on tackling this challenge. Focusing the new material explanation on these challenges helps to remind the readers of how this material is connected with the problems that they may encounter in the real world and makes it easier to relate to.

Our Teaching Approach

In order to develop the teaching methodology for this book we have reviewed and partially borrowed some of the best teaching metaphors and approaches from programming courses such as SwissEduc's "Informatik" [2] and Stanford's "Programming Methodology Course" [3]. We utilised concepts from motivational psychology [1] and applied techniques from gamification of education [4] and utilised relevant game design principles [5] to ensure that we can maintain a high level of student interest while working with programming exercises.

Our book relies on a visual teaching framework called "Clara's World". This framework uses similar principles of teaching programming in a visual way to those employed by introductory programming learning tools such as Karel [6], Scratch [7], Greenfoot [8] and Code.org [9]. Most existing frameworks and tools (i.e. Code.org and Scratch) have been developed with primary school students in mind. Our framework, however, targets university and high school students. Some frameworks (such as Greenfoot) are suitable for high school students, but they require students to memorise a large number of framework-specific commands, while in Clara's World the number of additional commands is very small, so students can concentrate on learning Java. Karel is probably the most suitable piece of software for the high school and university context, but it is underdeveloped. This is why Karel is usually employed in the first couple of introductory lectures, after which students have to switch to one of the traditional Java frameworks and abandon the highly visual approach of Karel. In contrast to Karel, this book in combination with

Clara's World helps to complete the entire first semester of a university programming fundamentals course without abandoning the visual teaching style, while allowing to maintain the required level of depth and complexity of explaining the key programming concepts.

The emphasis in this book is on making readers understand the very basic control structures that are common to all programming languages rather than immediately diving into the complexity of working with classes and objects as many other books do. The text gradually introduces readers to conditional statements, loops, methods, variables and arrays and then provides a quick overview of classes and objects towards the end. With the help of this book you will understand foundations of programming that are common for most modern programming languages, with a specific emphasis on Java (one the most in-demand languages on the market today).

We hope you enjoy learning Java in Clara's World and become as passionate about programming as we are.

Additional Online Resources

This book is supplied with digital online resources for students and educators. All programming problems presented in this book can be explored on the Clara's World website at <https://claraworld.net>.

Every programming problem covered in this book has a corresponding link to a problem template (for those readers willing to attempt the problem themselves), the link to the solution of this problem and a video recording of us solving this problem step-by-step. In addition, at the end of each chapter there is a link to exercises that readers are recommended to complete.

Instructors can be provided with a wide range of educational resources after providing their instructor status.

Digital Resources for Readers and Educators

This textbook provides free unrestricted access to online digital resources. These resources include templates and solutions for all problems presented in the book, as well as additional exercises for self study.

Students and other book readers should follow the instructions below to register on the Clara's World website and access these resources.

1. Go to <https://www.claraworld.net>
2. Select "Enter Clara's World"
3. Click "Register" and follow the on-screen instructions
4. Enter your registration details and make sure that you provide a valid email address
5. Select "Book" as the Organisation
6. Type "book2021" (without quotation marks) in the "Organisation Password" field
7. Check your inbox for the supplied email address and complete registration following the instructions you receive via email
8. Sign in with your email address and password and start using digital resources¹

Educators can be granted access to additional resources such as lecture slides, educational videos, a large pool of exercises for every topic covered in the book, the content management and marking system, automatic plagiarism detection of submitted code and other specialised tools, facilities for registering your organisation on Clara's World, facilities for setting up your own teaching resources and exercises, admin access to your organisation, video tutorials for educators and much more.

If you are interested in using Clara's World for your programming course as an educator please send us an introductory email to educator@claraworld.net, so that we can verify your educator status.

¹ See Appendix A for further information about creating an account on Clara's World and using the digital resources.

Contents

1	Introduction to Clara's World	1
1.1	Meet Clara	2
1.1.1	Clara's World	2
1.1.2	Programming Clara	3
1.2	Chapter Problem: Leaf Delivery	5
1.2.1	Introducing Clara's World Framework	6
1.2.2	Changing Clara's World Appearance	7
1.2.3	Working with Multiple Worlds	8
1.2.4	Solving the Chapter Problem	8
1.3	Revised Problem: Leaf Delivery and Reset	13
1.4	Summary	16
1.5	Exercises	17
2	Flow of Control	19
2.1	Chapter Story: Patching Roads	20
2.2	Chapter Problem: Clara Patches Roads	20
2.3	New Clara Commands	21
2.4	The Difference between <code>act()</code> and <code>run()</code>	22
2.5	Conditional Statements	23
2.6	Solving the Chapter Problem: Clara Patches Roads	25
2.7	Logical Operators	28
2.7.1	Logical NOT	28
2.7.2	Logical AND	31
2.7.3	Logical OR	32
2.8	Short-Circuited Operators	33
2.9	The Precedence of Logical Operators	33
2.10	Statement Blocks and Nested <code>if</code> -statements	34
2.11	Road Patching Example with Logical Operators	35
2.12	Conditional Operator	37

2.13	Common Errors	37
2.13.1	Indentation and the Use of Curly Brackets	37
2.13.2	More Formatting Errors	38
2.13.3	Using Semicolons	39
2.13.4	Issues with Explicit Mentioning of <code>true</code> and <code>false</code>	40
2.14	Summary	40
2.15	Exercises	41
3	Loops	43
3.1	Chapter Story: Vacuum Cleaner Robots	44
3.2	Chapter Problem: Clara the Vacuum Cleaner Robot	45
3.3	Loops	46
3.4	The <code>for</code> Loop	46
3.4.1	Examples of Using the <code>for</code> Loop	47
3.4.2	Solving the Chapter Problem with <code>for</code> Loops	48
3.4.3	Elements of the <code>for</code> Loop	56
3.4.4	Nested Loops	59
3.4.5	The <code>break</code> Statement	60
3.4.6	The <code>continue</code> Statement	61
3.5	Conditional Loops	62
3.5.1	Chapter Problem 2: Clara the General-Purpose Cleaner	63
3.5.2	The <code>while</code> Loop	64
3.5.3	Solving the Chapter Problem with <code>while</code> Loops	66
3.5.4	The <code>do-while</code> Loop	77
3.6	Common Errors	78
3.7	Summary	80
3.8	Exercises	81
4	Coding Style and Decomposition	83
4.1	Chapter Story: Consequences of Errors in Code	84
4.1.1	What Is Bad Code?	85
4.2	Key Principles of How to Write Good Code	86
4.2.1	Comments in Code	87
4.2.2	Code Indentation	92
4.2.3	Decomposition	93
4.2.4	Naming Conventions	99
4.3	Summary	99
4.4	Exercises	100
5	Variables	101
5.1	Chapter Story: The Friendship Algorithm	102
5.2	Chapter Problem: Selecting the Least Objectionable Activity	104
5.3	Variables	104
5.3.1	Variable Names	105
5.3.2	Variable Types	106
5.3.3	Variable Values	108
5.3.4	Declaring Variables	108

5.3.5	Assigning Variable Values	109
5.3.6	Variable Visibility	110
5.4	Receiving User Input from Keyboard	111
5.5	Console Output	112
5.6	Friendship Algorithm: Memorising Interests	113
5.7	Mid Point Example	115
5.8	Friendship Algorithm: Finding LOA	118
5.9	Summary	123
5.10	Exercises	123
6	Expressions	125
6.1	Chapter Story: Letterboxes with Flags	126
6.2	Chapter Problem: Using Flags	126
6.3	Boolean Variables	127
6.4	Solving the Chapter Problem	128
6.5	Variables and Literals	132
6.6	Working with Keyboard Input	133
6.7	Arithmetic Operators and Expressions with Variables	134
6.7.1	Subtraction	135
6.7.2	Addition	136
6.7.3	Remainder	136
6.7.4	Division	137
6.7.5	Variable Casting	138
6.8	Revised Chapter Problem: Aussie Postie	139
6.9	More on Expressions	146
6.9.1	Operator Precedence	146
6.9.2	The “+” Operator	147
6.9.3	Combined Assignment Operators	149
6.9.4	Relational Operators	149
6.9.5	Summary Example: Line in the Middle	150
6.10	Summary	155
6.11	Exercises	155
7	Debugging and Random Numbers	157
7.1	Chapter Story: Pac-man	158
7.2	Chapter Problem 1: Ghosts Performing Random Walk	158
7.3	Random Numbers	159
7.3.1	Solving the Chapter Problem: Finding the Leaf via Random Walk	161
7.4	Debugging	166
7.4.1	Chapter Problem 2: Repairing Clara’s World after an Earthquake	166
7.4.2	Desk-Checking	168
7.4.3	Console Output	169
7.4.4	Embedded Debugger	171

7.5	Advanced Keyboard Input	173
7.6	Creating Constants	174
7.7	The <code>switch-case</code> Construct.....	175
7.8	Obstacle Avoidance Bug Style.....	178
7.8.1	Chapter Problem 3: Bug Navigation.....	179
7.9	Summary	190
7.10	Exercises	190
8	Methods	191
8.1	Chapter Story: Simulating Projectile Motion in Video Games	192
8.1.1	Simplified Projectile Motion Simulation	192
8.2	Chapter Problem: Jumping Like a Flea	194
8.3	Methods	195
8.3.1	The <code>void</code> Returning Methods	197
8.3.2	Methods Returning a Value	198
8.3.3	Using <code>return</code> with <code>void</code> Methods	199
8.3.4	Methods Receiving Input.....	200
8.4	Information Hiding	202
8.5	Solving the Jumping Clara Problem	203
8.6	Passing Parameters by Value	208
8.7	Passing Parameters by Reference	210
8.8	Summary	210
8.9	Exercises	210
9	Arrays	211
9.1	Chapter Story: Game of Life	212
9.2	Chapter Problem 1: Game of Life in Clara's World	213
9.3	Arrays.....	213
9.3.1	Creating an Array	213
9.3.2	Working with Array Elements.....	215
9.3.3	Processing Arrays.....	216
9.3.4	Alternative Array Declaration	216
9.3.5	Array Length.....	217
9.3.6	Array Initialisation	218
9.4	Passing Arrays to Methods	219
9.5	Chapter Problem 2: Friendship Algorithm with Arrays	221
9.5.1	Sequential Search Algorithm	221
9.5.2	Solving Chapter Problem 2: Finding the Common Interest ..	222
9.6	Array Variables as References	225
9.6.1	Copying Arrays	226
9.6.2	Comparing Arrays	227
9.7	Multidimensional Arrays	228
9.7.1	Two-Dimensional Arrays.....	228
9.7.2	Initialising Two-Dimensional Arrays	229
9.7.3	Processing Two-Dimensional Arrays	229

9.7.4	Ragged Arrays	230
9.7.5	More than Two Dimensions	230
9.8	Solving Chapter Problem 1: Game of Life	231
9.9	Summary	236
9.10	Exercises	237
10	Classes and Objects	239
10.1	Chapter Story: The Amazing Ants	240
10.2	Chapter Problem: Clara Imitates Ant Behaviour	244
10.3	Introduction to Classes and Objects	245
10.4	Creating Objects	247
10.5	Creating a Class Example	247
10.6	Access Specifiers <code>private</code> and <code>public</code>	249
10.7	The <code>static</code> Keyword	249
10.8	Inheritance	250
10.9	Overriding Methods	251
10.10	Accessing Methods and Variables	252
10.11	Constructors	253
10.12	Method Overloading	254
10.13	Passing Objects as Method Arguments	255
10.14	Chapter Problem Solution	255
10.15	Summary	269
10.16	Exercises	270
10.17	Epilogue	270
A	Clara's World User Guide	271
A.1	Quick Start	272
A.2	Creating an Account	273
A.3	Logging in	275
A.4	Recovering your Password and Verification	275
A.5	User Home Screen	276
A.6	Schedules, Practicals and Exercises	277
A.6.1	Schedules	277
A.6.2	Practicals	280
A.6.3	Exercises	281
A.7	Rankings	284
A.8	Profile	285
A.9	How to Create Your Own Exercises	287
	References	293

About the Authors



Dr Anton Bogdanovych is a senior lecturer of entertainment computing and applied artificial intelligence at Western Sydney University, School of Computer, Data and Mathematical Sciences.

He has worked at Western Sydney University since 2007 and currently teaches “Video Game Development” and “Programming Fundamentals”.

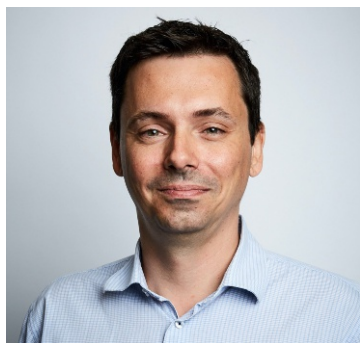
Anton’s innovative approach to teaching programming fundamentals (that laid the foundations of this book) has been recognised on a national and international level. Dr Bogdanovych has been awarded the **ICT Higher Education Educator of the Year 2015** by the Australian Computer Society (ACS) and received the **International ICT Educator of the Year 2016 Award** by The South East Asia Regional Computer Confederation (SEARCC).

Since 2017 Anton Bogdanovych has been the director of academic program for the Bachelor of Entrepreneurship degree at Western Sydney University.

Anton completed his PhD in Computer Science at the University of Technology Sydney in May 2008. He is currently involved in a number of research projects related to virtual reality, artificial intelligence, robotics and motion capture.

Before starting his PhD, Dr Bogdanovych worked as a researcher and software developer in the Multiagent Systems Department in the German Research Centre for Artificial Intelligence (DFKI). The DFKI is one of the largest and most prestigious artificial intelligence research centres in the world.

Anton's industry experience with Software MacKiev (one the largest companies that produces Macintosh versions of famous Windows titles) has helped him to get familiar with programming methodologies, tools and people management techniques employed for producing large-scale commercial software. One of his products (WorldBook Encyclopedia) was preinstalled on every iMac computer sold in 2001.



Dr. Tomas Trescak holds a PhD in computer science with a specialisation in artificial intelligence from the Artificial Intelligence Research Institute, Barcelona, Spain (IIIA) of the Spanish Research Council (CSIC). For the past seven years he has worked as a senior lecturer of intelligent systems at Western Sydney University. Since 2020, Tomas has acted as a director of academic program for undergraduate studies in information and communication technology (ICT).

Tomas is an avid open-source enthusiast, contributing to existing software initiatives. He has published over 50 open-source packages for back-end and front-end development, as well as many software tools for virtual and augmented realities.

Tomas is the creator of the Clara's World framework that we utilise in this book. For his academic and software development work, Tomas has received multiple awards, among which are the **2016 Best Innovator Award** by Unearthed Association and **2015 Gold Disruptor Award** in the Best Australian ICT Educator category by the Australian Computer Society.

In his academic work Tomas seeks new ways to facilitate complex cognitive tasks in simulation, education, healthcare, cyber security and social sciences through the application of artificial intelligence. The primary focus of his research is on facilitation of creation and execution of self-adaptable, interactive normative 3D environments and their subsequent application to the fields of agent-based simulation, as well as virtual and augmented realities.