

Introduction to Computer Science WITH ANDROID™

—{ THE LEARNING-BY-DOING SERIES }—

Second Edition

<http://book.roumani.ca>

Hamzeh Roumani

Department of Electrical Engineering and Computer Science
York University, Toronto, Ontario, Canada

DOWNLOAD

Copyright © 2020 CompuScope Consulting. All rights reserved.

The i2c Library and its API © 2020 Hamzeh Roumani. All rights reserved.

Android is a trademark of Google LLC.

ISBN-10: 1775125416

ISBN-13: 978-1-7751254-1-9

Contents at a Glance

CHAPTER	PART	TOPICS
0. PRELIMINARIES	Doing	The Zero App
	Learning	Computer Science, Programming, and Java
1. THE LANDSCAPE	Doing	The BMI App
	Learning	OOP
2. DECLARE, SET, GO!	Doing	Mortgage Calculator
	Learning	Types, Assignment and Memory, Expressions
3. A SYMPHONY OF APIS	Doing	The MCalcPro App
	Learning	Selection, Iteration, Exceptions
4. STRING LAND	Doing	The KryptoNote App
	Learning	The String API and Regular Expressions
5. COLLECTIONS	Doing	The Country Game
	Learning	Lists, Sets, Maps, and the Array Bridge

The CS Trail

CHAPTER	THEME	TOPICS
L0	About	Roadmap
	I/O	Console I/O
L1	Abstraction	Indirection
	I/O	GUI and Parameter/Return
L2	Computability	Decidability
	Computability	Complexity
L3	Abstraction	Delegation
	Computability	Tractability
L4	Abstraction	Recursion
	I/O	Network I/O
L5	Computability	P vs NP
	Epilogue	The Case of the Missing Number!

Table of Contents

Preface	xiii	
Acknowledgment	xv	
CHAPTER 0	PRELIMINARIES	DOING
D0.1 – The Toolbox		2
D0.1.a – The IDE		2
D0.1.b – The Device		2
D0.2 – A Quick Tour of the IDE		3
D0.2.a – Creating a Project		3
D0.2.b – The Main Screen		3
D0.2.c – User Interface Tools		4
D0.2.d – Saving Your Work		10
D0.2.e – Running a Project		10
D0.3 – Exercises		11
CHAPTER 0	PRELIMINARIES	LEARNING
L0.1 – Computer Science (CS)		18
L0.1.a – A Broad Definition		18
L0.1.b – Programming		19
L0.2 – A Brief Encounter with Java		20
L0.2.a – Creating a Java Class		20
L0.2.b – Testing a Java Class		23
L0.2.c – Installing and Testing a Java Library		26
L0.2.d – Using APIs		27
L0.3 – The CS Trail		32
L0.3.a – About: Roadmap		32
L0.3.b – I/O: Console		32

L0.4 – Exercises	34
-------------------------	-----------

CHAPTER 1	THE LANDSCAPE	DOING
------------------	----------------------	--------------

D1.1 – Requirement	39
---------------------------	-----------

D1.1.a – The BMI Calculator App	39
---------------------------------	----

D1.1.b – UI and Behaviour	39
---------------------------	----

D1.2 – Design	40
----------------------	-----------

D1.2.a – The MVC Architecture	40
-------------------------------	----

D1.2.b – The Model's API	40
--------------------------	----

D1.3 – Implementation	41
------------------------------	-----------

D1.3.a – Building the View	41
----------------------------	----

D1.3.b – Building the Model	42
-----------------------------	----

D1.3.c – Testing the Model	43
----------------------------	----

D1.3.d – The Controller-View Interaction	44
--	----

D1.3.e – The Controller-Model Interaction	45
---	----

D1.4 – Deployment	46
--------------------------	-----------

D1.4.a – Integration Testing	46
------------------------------	----

D1.4.b – Deployment Options	47
-----------------------------	----

D1.5 – Exercises	48
-------------------------	-----------

CHAPTER 1	THE LANDSCAPE	LEARNING
------------------	----------------------	-----------------

L1.1 – Java Basics	56
---------------------------	-----------

L1.1.a – About	56
----------------	----

L1.1.b – Language Elements	57
----------------------------	----

L1.1.c – Naming Convention and Style	59
--------------------------------------	----

L1.1.d – Syntax, Runtime, and Logic Errors	60
--	----

L1.2 – Object Oriented Programming	62
---	-----------

L1.2.a – The Two Paradigms	62
----------------------------	----

L1.2.b – OOP in Action	65
------------------------	----

L1.2.c – Anatomy of an API	66
L1.2.d – Separation of Concerns	68
L1.3 – The CS Trail	69
L1.3.a – Abstraction: Indirection	69
L1.3.b – I/O: GUI and Parameter/Return	70
L1.4 – Exercises	71
CHAPTER 2	DECLARE, SET, GO!
CHAPTER 2	DOING
D2.1 – Requirement	77
D2.1.a – The Mortgage App	77
D2.1.b – UI and Behaviour	77
D2.2 – Design	78
D2.2.a – The Model	78
D2.2.b – The View	79
D2.2.c – The Controller	79
D2.3 – Implementation	79
D2.3.a – JUnit Testing	79
D2.3.b – Testing with Assertions	80
D2.4 – Deployment	81
D2.5 – Exercises	82
CHAPTER 2	DECLARE, SET, GO!
CHAPTER 2	LEARNING
L2.1 – Declare	86
L2.1.a – The int Type	86
L2.1.b – The double Type	87
L2.1.c – The boolean Type	88
L2.1.d – Other Primitive Types	89
L2.1.e – Non-Primitive Types	90
L2.2 – Set	92

L2.2.a – The Assignment Statement	92
L2.2.b – Memory Allocation	93
L2.2.c – Setting Class-Type Variables	95
L2.2.d – The Birth and Death of Objects	96
L2.3 – Go	99
L2.3.a – Expression Evaluation	99
L2.3.b – Relational Operators	100
L2.3.c – Mixing Primitive Types	102
L2.3.d – Mixing Non-Primitive Types	104
L2.4 – The CS Trail	105
L2.4.a – Computability: Decidability	105
L2.4.b – Computability: Complexity	105
L2.5 – Exercises	107
CHAPTER 3	A SYMPHONY OF APIs
	DOING
D3.1 – Requirement	113
D3.1.a – The MCalcPro App	113
D3.1.b – UI and Behaviour	113
D3.2 – Design	116
D3.2.a – TheMPro API	116
D3.2.b – The Toast API	116
D3.2.c – The IME API	116
D3.2.d – The TTS API	116
D3.2.e – The Accelerometer API	116
D3.3 – Implementation	117
D3.3.a – Building the View	117
D3.3.b – Mortgage Data	118
D3.3.c – Exception Handling	122

D3.3.d – Voice Output	124
D3.3.e – Clear on Shake	125
D3.4 – Exercises	127
CHAPTER 3	A SYMPHONY OF APIs
	LEARNING
L3.1 – Control Structures	138
L3.1.a – The if Statement	138
L3.1.b – The for Statement	141
L3.1.c – The try-catch Blocks and throw	145
L3.2 – Other Flow Constructs	148
L3.2.a – The switch Statement	148
L3.2.b – The while Statement	150
L3.2.c – The do-while Statement	151
L3.3 – The CS Trail	152
L3.3.a – Abstraction: Delegation	152
L3.3.b – Computability: Tractability	153
L3.4 – Exercises	155
CHAPTER 4	STRING LAND
	DOING
D4.1 – Requirement	161
D4.1.a – The KryptoNote App	161
D4.1.b – UI and Behaviour	161
D4.2 – Design	163
D4.2.a – The Vigenère Cipher	163
D4.2.b – File Storage	164
D4.3 – Implementation	165
D4.3.a – Building the View	165
D4.3.b – The Cipher Model	166
D4.3.c – The Activity-View Connection	168

D4.3.d – File I/O	168
D4.4 – Exercises	172
CHAPTER 4	STRING LAND
	LEARNING
L4.1 – The String API	184
L4.1.a – The Basics	184
L4.1.b – Transformers	186
L4.1.c – Comparators	186
L4.1.d – String-to-Number	187
L4.1.e – String-to-Character	188
L4.2 – Regular Expressions	189
L4.2.a – Pattern Description Language	189
L4.2.b – Pattern Recognition	192
L4.2.c – Pattern-Driven Transformers	195
L4.3 – The CS Trail	197
L4.3.a – Abstraction: Recursion	197
L4.3.b – I/O: Network	200
L4.4 – Exercises	202
CHAPTER 5	COLLECTIONS
	DOING
D5.1 – Requirement	209
D5.1.a – The Caps App	209
D5.1.b – UI and Behaviour	209
D5.2 – Design	212
D5.2.a – The Country API	212
D5.2.b – The CountryDB API	213
D5.2.c – The Game API	215
D5.3 – Implementation	216
D5.3.a – Building the View	216

D5.3.b – The Game Model	217
D5.3.c – Building the Controller	218
D5.4 – Exercises	219
CHAPTER 5	COLLECTIONS
CHAPTER 5	LEARNING
L5.1 – The Collection Framework	230
L5.1.a – Overview	230
L5.1.b – Working with Lists	231
L5.1.c – Working with Sets	233
L5.1.d – Working with Maps	235
L5.1.e – The Array Bridge	237
L5.1.f – The Iterator	238
L5.2 – Selected Algorithms	239
L5.2.a – Sorting a List	239
L5.2.b – Stats with Maps	240
L5.2.c – Inverting a Map	241
L5.2.d – Compressing a Document	244
L5.3 – The CS Trail	249
L5.3.a – Computability: P vs NP	249
L5.3.b – Epilogue: The Case of the Missing Number	250
L5.4 – Exercises	253
APPENDIX A	OPERATORS
Appendix-A: Operators	258
APPENDIX B	REGULAR EXPRESSIONS
Appendix-B: Regular Expressions	260
APPENDIX C	CHARACTER CODES
Appendix-C: Character Codes	262

APPENDIX D THE I2C API

Appendix-D: The i2c API _____ 267

APPENDIX E ASSESSMENT

Appendix-E: Assessment _____ 270

INDEX _____ 281

DOWNLOAD

PREFACE

This book is ideally suited for a first-year course in computer science, engineering, or related field. It adopts a two-pillared approach that has proven successful:

Pillar 1

It covers the standard topics for such a course *immersed* in mobile app development. By using such an attractive medium to present ideas, students see relevance and engage. And thanks to the richness of this medium, advanced students can be challenged. The extent of coverage of the mobile aspects was carefully tuned so as not to overwhelm or dominate.

Pillar 2

Chapters come in pairs: *Doing* and *Learning*. The first builds an app and the second exposes its underlying concepts. This choreography—a concrete experience followed by just-in-time conceptualization with reflection in between—lies at the heart of experiential pedagogy. Best practices and methodologies are actually practiced, not just preached.

The book has six chapters numbered 0 to 5. The *Doing/Learning* parts of each are prefixed by D/L, so D3 is the Doing part of Chapter 3.

The *Second Edition* contains numerous revisions and additions to fix errors; address changes in the UI of Android Studio; provide additional examples; and beef up the coverage of core Computer Science concepts. The most prominent addition is the **CS Trail**, which appears at the end of every Learning Chapter and comes with its own exercises. It has lookout points that invite the reader to take a glimpse at separation of concerns, complexity, and P vs NP at a level appropriate for this book. This edition includes a sample *assessment* in Appendix-E that tests reasoning and problem solving skills in addition to subject-matter comprehension.

The *website* of this book:

<http://book.roumani.ca/>

has a host of resources, including an errata, tutorial videos, answers to selected assessment and exercise questions, and the i2c library. Moreover, Chapter 0 (whose content is revised as needed when new versions of Android are released) can be downloaded from the website.

*Prof. Hamzeh Roumani,
3M National Teaching Fellow
Toronto, Ontario, Canada*

DOWNLOAD

ACKNOWLEDGMENT

*I would like to thank my wife **Souad** for reviewing the manuscript and building the apps. She has provided insightful critique and refreshing suggestions, including the juxtaposition of the doing/ learning parts.*

DOWNLOADED

DOWNLOAD