Table of Contents

1: Introduction 1

About this book » 1
Why use Python's advanced features? » 1
How to use this book » 2
Exercises and solutions » 5
A note on setting up your environment » 6
Joined-up programming » 7
Getting in touch » 8

2: Recursion and trees 9

Recursively generating kmers » 9
Processing tree-like data » 16
Recap » 28
Exercise » 30
Solution » 31

3: Complex data structures 36

Tuples » 36
Sets » 38
Lists of lists » 40
Lists of dicts and lists of tuples » 42
Other complex structures » 45
Recap » 55
Exercises » 57
Solutions » 58

4: Object-oriented Python 64

Introduction » 64
A simple DNA sequence class » 66
Constructors » 70
Inheritance » 76
Overriding » 85
Calling methods in the superclass » 87

Polymorphism » 89
Recap » 90
Exercise » 92
Solution » 93

5: Functional Python 109

Introduction » 109
State and mutability » 109
Side effects » 110
Functions as objects » 113
What is to be calculated » 116
Built-in higher order functions » 117
map » 117
filter » 122
sorted » 124
reduce » 129
Writing higher-order functions » 131
Recap » 137
Exercises » 138
Solutions » 141

6: Iterators, comprehensions & generators 157

Defining lists » 157
Lists and iterables » 158
List comprehensions » 159
Dictionary comprehensions » 163
Set comprehensions » 165
Iterators and generators » 165
Recap » 171
Exercises » 172
Solutions » 173

7: Exception handling 180

Catching exceptions » 181
Catching specific errors » 183
else blocks in exception handling » 188
finally blocks in exception handling » 189

Nested try/except blocks » 193
Exceptions bubble up » 194
Raising exceptions » 197
Custom exception types » 200
Recap » 202
Exercises » 204
Solutions » 208