

Algorithmic Problem Solving with Python

John B. Schneider

Shira Lynn Broschat

Jess Dahmen

February 22, 2019

Contents

| | | |
|----------|--|-----------|
| 1 | Introduction | 1 |
| 1.1 | Modern Computers | 1 |
| 1.2 | Computer Languages | 2 |
| 1.3 | Python | 3 |
| 1.4 | Algorithmic Problem Solving | 6 |
| 1.5 | Obtaining Python | 7 |
| 1.6 | Running Python | 8 |
| 1.6.1 | Interactive Sessions and Comments | 9 |
| 1.6.2 | Running Commands from a File | 11 |
| 1.7 | Bugs | 12 |
| 1.8 | The <code>help()</code> Function | 13 |
| 1.9 | Comments on Learning New Languages | 14 |
| 1.10 | Chapter Summary | 15 |
| 1.11 | Review Questions | 15 |
| 2 | Core Basics | 19 |
| 2.1 | Literals and Types | 19 |
| 2.2 | Expressions, Arithmetic Operators, and Precedence | 22 |
| 2.3 | Statements and the Assignment Operator | 24 |
| 2.4 | Cascaded and Simultaneous Assignment | 27 |
| 2.5 | Multi-Line Statements and Multi-Line Strings | 29 |
| 2.6 | Identifiers and Keywords | 30 |
| 2.7 | Names and Namespaces | 32 |
| 2.8 | Additional Arithmetic Operators | 37 |
| 2.8.1 | Exponentiation | 37 |
| 2.8.2 | Floor Division | 37 |
| 2.8.3 | Modulo and <code>divmod()</code> | 38 |
| 2.8.4 | Augmented Assignment | 40 |
| 2.9 | Chapter Summary | 42 |
| 2.10 | Review Questions | 43 |
| 2.11 | Exercises | 49 |
| 3 | Input and Type Conversion | 51 |
| 3.1 | Obtaining Input: <code>input()</code> | 51 |
| 3.2 | Explicit Type Conversion: <code>int()</code> , <code>float()</code> , and <code>str()</code> | 53 |

| | | |
|----------|--|------------|
| 3.3 | Evaluating Strings: eval () | 57 |
| 3.4 | Chapter Summary | 59 |
| 3.5 | Review Questions | 59 |
| 3.6 | Exercises | 65 |
| 4 | Functions | 67 |
| 4.1 | Void Functions and None | 68 |
| 4.2 | Creating Void Functions | 69 |
| 4.3 | Non-Void Functions | 72 |
| 4.4 | Scope of Variables | 76 |
| 4.5 | Scope of Functions | 78 |
| 4.6 | print () vs. return | 79 |
| 4.7 | Using a main () Function | 81 |
| 4.8 | Optional Parameters | 82 |
| 4.9 | Chapter Summary | 86 |
| 4.10 | Review Questions | 87 |
| 4.11 | Exercises | 92 |
| 5 | Introduction to Objects | 95 |
| 5.1 | Overview of Objects and Classes | 95 |
| 5.2 | Creating a Class: Attributes | 97 |
| 5.3 | Creating a Class: Methods | 99 |
| 5.4 | The dir () Function | 101 |
| 5.5 | The __init__ () Method | 103 |
| 5.6 | Operator Overloading | 105 |
| 5.7 | Take-Away Message | 106 |
| 5.8 | Chapter Summary | 107 |
| 6 | Lists and for-Loops | 109 |
| 6.1 | lists | 110 |
| 6.2 | list Methods | 111 |
| 6.3 | for -Loops | 113 |
| 6.4 | Indexing | 115 |
| 6.5 | range () | 118 |
| 6.6 | Mutability, Immutability, and Tuples | 121 |
| 6.7 | Nesting Loops in Functions | 123 |
| 6.8 | Simultaneous Assignment with Lists | 126 |
| 6.9 | Examples | 127 |
| 6.9.1 | Storing Entries in a list | 127 |
| 6.9.2 | Accumulators | 129 |
| 6.9.3 | Fibonacci Sequence | 130 |
| 6.10 | Chapter Summary | 132 |
| 6.11 | Review Questions | 133 |

| | | |
|----------|--|------------|
| 7 | More on for-Loops, Lists, and Iterables | 143 |
| 7.1 | for-Loops within for-Loops | 143 |
| 7.2 | lists of lists | 148 |
| 7.2.1 | Indexing Embedded lists | 151 |
| 7.2.2 | Simultaneous Assignment and lists of lists | 154 |
| 7.3 | References and list Mutability | 157 |
| 7.4 | Strings as Iterables or Sequences | 162 |
| 7.5 | Negative Indices | 164 |
| 7.6 | Slicing | 165 |
| 7.7 | list Comprehension (optional) | 168 |
| 7.8 | Chapter Summary | 171 |
| 7.9 | Review Questions | 172 |
| 8 | Modules and import Statements | 177 |
| 8.1 | Importing Entire Modules | 179 |
| 8.2 | Introduction to Complex Numbers | 181 |
| 8.3 | Complex Numbers and the cmath Module | 185 |
| 8.4 | Importing Selected Parts of a Module | 187 |
| 8.5 | Importing an Entire Module Using * | 189 |
| 8.6 | Importing Your Own Module | 190 |
| 8.7 | Chapter Summary | 193 |
| 8.8 | Review Questions | 194 |
| 9 | Strings | 195 |
| 9.1 | String Basics | 196 |
| 9.2 | The ASCII Characters | 199 |
| 9.3 | Escape Sequences | 200 |
| 9.4 | chr () and ord () | 203 |
| 9.5 | Assorted String Methods | 209 |
| 9.5.1 | lower (), upper (), capitalize (), title (), and swapcase () | 209 |
| 9.5.2 | count () | 210 |
| 9.5.3 | strip (), lstrip (), and rstrip () | 210 |
| 9.5.4 | __repr__ () | 211 |
| 9.5.5 | find () and index () | 212 |
| 9.5.6 | replace () | 215 |
| 9.6 | split () and join () | 215 |
| 9.7 | Format Strings and the format () Method | 218 |
| 9.7.1 | Replacement Fields as Placeholders | 219 |
| 9.7.2 | Format Specifier: Width | 222 |
| 9.7.3 | Format Specifier: Alignment | 223 |
| 9.7.4 | Format Specifier: Fill and Zero Padding | 224 |
| 9.7.5 | Format Specifier: Precision (Maximum Width) | 225 |
| 9.7.6 | Format Specifier: Type | 226 |
| 9.7.7 | Format Specifier: Summary | 227 |
| 9.7.8 | A Formatting Example | 228 |

[Click here to download full PDF material](#)