

Contents

Let's Go

xiii

1

Getting Started with Python

1



- 1.1 Algorithms and algorithmic thinking 1
 - 1.1.1 Algorithmic thinking 1
 - 1.1.2 States 2
 - 1.1.3 What does $a = a + 1$ mean? 3
 - 1.1.4 Symbolic versus numerical 4
- 1.2 Launch Python 4
 - 1.2.1 IPython console 5
 - 1.2.2 Error messages 9
 - 1.2.3 Sources of help 9
 - 1.2.4 Good practice: Keep a log 11
- 1.3 Python modules 11
 - 1.3.1 import 11
 - 1.3.2 from ... import 12
 - 1.3.3 NumPy and PyPlot 12
- 1.4 Python expressions 13
 - 1.4.1 Numbers 13
 - 1.4.2 Arithmetic operations and predefined functions 13
 - 1.4.3 Good practice: Variable names 15
 - 1.4.4 More about functions 15

2

Organizing Data


17

- 2.1 Objects and their methods 17
- 2.2 Lists, tuples, and arrays 19
 - 2.2.1 Creating a list or tuple 19
 - 2.2.2 NumPy arrays 19
 - 2.2.3 Filling an array with values 21
 - 2.2.4 Concatenation of arrays 22
 - 2.2.5 Accessing array elements 23
 - 2.2.6 Arrays and assignments 24
 - 2.2.7 Slicing 24
 - 2.2.8 Flattening an array 26

- 2.2.9 Reshaping an array 26
- 2.2.10  Lists and arrays as indices 26
- 2.3 Strings 27
 - 2.3.1 Formatting strings with the `format()` method 29
 - 2.3.2  Formatting strings with `%` 30



3 Structure and Control


31

- 3.1 Loops 31
 - 3.1.1 `for` loops 31
 - 3.1.2 `while` loops 33
 - 3.1.3 Very long loops 33
 - 3.1.4 Infinite loops 33
- 3.2 Array operations 34
 - 3.2.1 Vectorizing math 34
 - 3.2.2 Matrix math 36
 - 3.2.3 Reducing an array 36
- 3.3 Scripts 37
 - 3.3.1 The Editor 37
 - 3.3.2  Other editors 38
 - 3.3.3 First steps to debugging 38
 - 3.3.4 Good practice: Commenting 40
 - 3.3.5 Good practice: Using named parameters 43
 - 3.3.6 Good practice: Units 44
- 3.4 Contingent behavior: Branching 44
 - 3.4.1 The `if` statement 45
 - 3.4.2 Testing equality of floats 46
- 3.5 Nesting 47

4 Data In, Results Out

48

- 4.1 Importing data 48
 - 4.1.1 Obtaining data 49
 - 4.1.2 Bringing data into Python 49
- 4.2 Exporting data 52
 - 4.2.1 Scripts 52
 - 4.2.2 Data files 52
- 4.3 Visualizing data 54
 - 4.3.1 The `plot` command and its relatives 55
 - 4.3.2 Manipulate and embellish 57
 - 4.3.3  More about figures and their axes 59
 - 4.3.4  Error bars 60
 - 4.3.5 3D graphs 60
 - 4.3.6 Multiple plots 61
 - 4.3.7 Subplots 62

- 4.3.8 Saving figures 62
- 4.3.9  Using figures in other applications 63

5

First Computer Lab



64


- 5.1 HIV example 64
 - 5.1.1 Explore the model 64
 - 5.1.2 Fit experimental data 65
- 5.2 Bacterial example 66
 - 5.2.1 Explore the model 66
 - 5.2.2 Fit experimental data 66

6

More Python Constructions

68

- 6.1 Writing your own functions 68
 - 6.1.1 Defining functions in Python 69
 - 6.1.2 Updating functions 71
 - 6.1.3 Arguments, keywords, and defaults 71
 - 6.1.4 Return values 72
 - 6.1.5 Functional programming 73
- 6.2 Random numbers and simulation 74
 - 6.2.1 Simulating coin flips 74
 - 6.2.2 Generating trajectories 75
- 6.3 Histograms and bar graphs 76
 - 6.3.1 Creating histograms 76
 - 6.3.2 Finer control 77
- 6.4 Contour plots and surfaces 77
 - 6.4.1 Generating a grid of points 78
 - 6.4.2 Contour plots 78
 - 6.4.3 Surface plots 79
- 6.5 Numerical solution of nonlinear equations 79
 - 6.5.1 General real functions 80
 - 6.5.2 Complex roots of polynomials 81
- 6.6 Solving systems of linear equations 81
- 6.7 Numerical integration 82
 - 6.7.1 Integrating a predefined function 82
 - 6.7.2 Integrating your own function 83
 - 6.7.3 Oscillatory integrands 84
 - 6.7.4  Parameter dependence 84
- 6.8 Numerical solution of differential equations 84
 - 6.8.1 Reformulating the problem 85
 - 6.8.2 Solving an ODE 86
 - 6.8.3  Parameter dependence 87
- 6.9 Vector fields and streamlines 88
 - 6.9.1 Vector fields 88
 - 6.9.2 Streamlines 89

7	Second Computer Lab	91
	7.1 Generating and plotting trajectories	91
	7.2 Plotting the displacement distribution	91
	7.3 Rare events	93
	7.3.1 The Poisson distribution	93
	7.3.2 Waiting times	94
8	Still More Techniques	96
	8.1 Image processing	96
	8.1.1 Images as NumPy arrays	96
	8.1.2 Saving and displaying images	97
	8.1.3 Manipulating images	97
	8.2 Displaying Data as an Image	98
	8.3 Animation	99
	8.3.1 Creating animations	99
	8.3.2 Saving animations	100
	HTML movies	100
	 Using an encoder	102
	8.4 Analytic calculations	103
	8.4.1 The SymPy library	103
	8.4.2 Wolfram Alpha	104
9	Third Computer Lab	106
	9.1 Convolution	106
	9.1.1 Python tools for image processing	107
	9.1.2 Averaging	108
	9.1.3 Smoothing with a Gaussian	108
	9.2 Denoising an image	109
	9.3 Emphasizing features	109
	Get Going	111
A	Installing Python	113
	A.1 Install Python and Spyder	113
	A.1.1 Graphical installation	114
	A.1.2 Command line installation	115
	A.2 Setting up Spyder	116
	A.2.1 Working directory	116
	A.2.2 Interactive graphics	117
	A.2.3 Script template	117
	A.2.4 Restart	118
	A.3 Keeping up to date	118
	A.4 Installing FFmpeg	118

A.5 Installing ImageMagick 119

B Jupyter Notebooks 120


B.1 Getting Started 120

B.1.1 Launch Jupyter Notebooks 120

B.1.2 Open a Notebook 121

B.1.3 Multiple Notebooks 121

B.1.4 Quitting Jupyter 122

B.1.5  Setting the Default Directory 123

B.2 Cells 123

B.2.1 Code cells 123

B.2.2 Graphics 124

B.2.3 Markdown cells 124

B.2.4 Edit mode and command mode 125

B.3 Sharing 125

B.4 More details 125

B.5 Pros and Cons 125

C Errors and Error Messages 127

C.1 Python errors in general 127

C.2 Some common errors 128

D Python 2 versus Python 3 131

D.1 Division 131

D.2 Print command 131

D.3 User input 132

D.4 More assistance 133

E Under the Hood 134

E.1 Assignment statements 134

E.2 Memory management 135

E.3 Functions 135

E.4 Scope 137

E.4.1 Name collisions 138

E.4.2 Variables passed as arguments 139

E.5 Summary 140

F Answers to “Your Turn” Questions 141

Acknowledgments 145

References 147

Index 149