



## 1. ELEMENTS OF PROGRAMMING

Our goal in this chapter is to convince you that composing a computer program is easier than writing a piece of text such as a paragraph or an essay. Writing prose is difficult: we spend many years in school to learn how to do it. By contrast, just a few building blocks suffice to take us into a world where we can harness the computer to help us solve all sorts of fascinating problems that would be otherwise unapproachable. In this chapter, we take you through these building blocks, get you started on programming in Python, and study a variety of interesting programs.

- [1.1 Your First Program](#) instructs you on how to compose and execute a Python program on your system.
- [1.2 Built-in Types of Data](#) describes Python's built-in data types for manipulating strings, integers, real numbers, and booleans.
- [1.3 Conditionals and Loops](#) introduces Python structures for control flow, including `if`, `while`, and `for` statements.
- [1.4 Arrays](#) considers a data structure known as the array for organizing large quantities of data.
- [1.5 Input and Output](#) extends the set of input and output abstractions (from command-line input and standard output) to include standard input, standard drawing, and standard audio.
- [1.6 Case Study: Random Web Surfer](#) presents a case study that models the behavior of a web surfer using a Markov chain.

### Python Programs in this Chapter

Below is a list of Python programs and data files used in this chapter.

REF	PROGRAM	DESCRIPTION	DATA
<a href="#">1.1.1</a>	<a href="#">helloworld.py</a>	Hello, World	—
<a href="#">1.1.2</a>	<a href="#">useargument.py</a>	using a command-line argument	—
<a href="#">1.2.1</a>	<a href="#">ruler.py</a>	string concatenation example	—
<a href="#">1.2.2</a>	<a href="#">intops.py</a>	integer operators	—
<a href="#">1.2.3</a>	<a href="#">floatops.py</a>	float operators	—
<a href="#">1.2.4</a>	<a href="#">quadratic.py</a>	quadratic formula	—
<a href="#">1.2.5</a>	<a href="#">leapyear.py</a>	leap year	—
<a href="#">1.3.1</a>	<a href="#">flip.py</a>	flipping a fair coin	—
<a href="#">1.3.2</a>	<a href="#">tenhellos.py</a>	your first loop	—
<a href="#">1.3.3</a>	<a href="#">powersoftwo.py</a>	computing powers of two	—
<a href="#">1.3.4</a>	<a href="#">divisorpattern.py</a>	your first nested loops	—
<a href="#">1.3.5</a>	<a href="#">harmonic.py</a>	harmonic numbers	—
<a href="#">1.3.6</a>	<a href="#">sqrt.py</a>	Newton's method	—
<a href="#">1.3.7</a>	<a href="#">binary.py</a>	converting to binary	—
<a href="#">1.3.8</a>	<a href="#">gambler.py</a>	gambler's ruin simulation	—
<a href="#">1.3.9</a>	<a href="#">factors.py</a>	factoring integers	—
<a href="#">1.4.1</a>	<a href="#">sample.py</a>	sampling without replacement	—
<a href="#">1.4.2</a>	<a href="#">couponcollector.py</a>	coupon collector simulation	—
<a href="#">1.4.3</a>	<a href="#">primesieve.py</a>	sieve of Eratosthenes	—
<a href="#">1.4.4</a>	<a href="#">selfavoid.py</a>	self-avoiding random walks	—
<a href="#">1.5.1</a>	<a href="#">randomseq.py</a>	generating a random sequence	—
<a href="#">1.5.2</a>	<a href="#">twentyquestions.py</a>	interactive user input	—
<a href="#">1.5.3</a>	<a href="#">average.py</a>	averaging a stream of numbers	—
<a href="#">1.5.4</a>	<a href="#">rangefilter.py</a>	a simple filter	—
<a href="#">1.5.5</a>	<a href="#">plotfilter.py</a>	standard input to draw filter	<a href="#">usa.txt</a>
<a href="#">1.5.6</a>	<a href="#">bouncingball.py</a>	bouncing ball	—
<a href="#">1.5.7</a>	<a href="#">playthattune.py</a>	digital signal processing	<a href="#">elise.txt</a> <a href="#">ascale.txt</a> <a href="#">stairwaytoheaven.txt</a> <a href="#">entertainer.txt</a> <a href="#">firstcut.txt</a> <a href="#">freebird.txt</a> <a href="#">looney.txt</a>
<a href="#">1.6.1</a>	<a href="#">transition.py</a>	computing the transition matrix	<a href="#">small.txt</a> <a href="#">medium.txt</a>
<a href="#">1.6.2</a>	<a href="#">randomsurfer.py</a>	simulating a random surfer	—
<a href="#">1.6.3</a>	<a href="#">markov.py</a>	mixing a Markov chain	—