Random Numbers and Computers

Ronald T. Kneusel

Random Numbers and Computers

Springer

Ronald T. Kneusel
Thornton, CO, USA

ISBN 978-3-319-77696-5 ISBN 978-3-319-77697-2 (eBook)
https://doi.org/10.1007/978-3-319-77697-2

Library of Congress Control Number: 2018935925

Printed on acid-free paper

This Springer imprint is published by the registered company Springer International Publishing AG part of Springer Nature.
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

To my wife, Maria, for her love and her patience. It is the randomness of life that makes it most exciting, especially when shared.

Preface

This is a book about random numbers and computers. That is an ambiguous sentence, so let's be more precise: this is a book about pseudorandom numbers and computers. While closer to the mark, we are not quite there, so let's try to be still more precise: this is a book about algorithms capable of generating sequences of numbers that, according to a series of statistical tests, are suitably indistinguishable from number sequences generated by true random processes.

Let's break it down some more:

- Computers often need sequences of numbers where it is not possible, given n_i, to predict n_{i+1}. Ideally, n_{i+1} should not be predictable from any combination of the previous k numbers in the sequence (for any k).
- While true random processes exist, an assumption on our part we will not prove, it is not typically practical for computers to make use of random processes so we are left with algorithmic processes. Computers are good at following algorithms.
- This is a book about algorithms that can generate sequences of numbers approximating the requirements of a true random sequence. We call these algorithms *pseudorandom number generators* (PRNGs).
- So, then, in the end, this is a book about pseudorandom number generators: how they work, how to test them, and how to use them with computers.

The above implies that this is not a book about randomness *per se*. Concepts related to random sequences are discussed briefly in Chap. 1 but only as far as is necessary to motivate the introduction and discussion of pseudorandomness and eventually pseudorandom number generators.

The goal of the book is to equip the reader with enough background in pseudorandom number generators to understand how they work and how to demonstrate they are useful. We will do this by developing actual code, in C or Python, to make the algorithms concrete. Along the way we will run various experiments designed to make the code practical and to demonstrate key concepts from the discussion.

A Note About Terminology

The phrases "pseudorandom number" and "pseudorandom number generator" are somewhat tedious. Therefore, we will frequently succumb to the temptation to write "random number" or "random number generator" when we in fact mean "pseudorandom number" or "pseudorandom number generator." We will rely on context to clarify when these phrases are used loosely or precisely. We trust that the reader will understand and follow along without difficulty.

Who Should Read This Book

This is a book for anyone who develops software, including software engineers, scientists, engineers, and students of those disciplines. It is particularly suitable for scientists and engineers because of their frequent use of random numbers, either directly or indirectly, through the software they use on a daily basis. A poor choice of random number generator can prove catastrophic or at the least frustrating. Knowing something about how random number generators work will help avoid unfortunate consequences.

How To Use This Book

A basic reading of the book includes Chaps. 1, 2, and 4. These chapters introduce the concepts of pseudorandomness, the core set of uniform pseudorandom number generators, and how to test such generators.

Chapter 3 will be useful to those engaged in simulations of some kind. Read it after Chap. 2 as nonuniform pseudorandom number generation is based on uniform generators.

Chapter 5 covers methods for generating parallel streams of pseudorandom numbers. This chapter is best read after Chap. 2 and Chap. 4 on testing. While no examples are given specifically for graphics processors (GPUs), the methods are generic and apply to GPUs as well as distributed processes on CPUs.

Chapter 6 covering cryptographically secure pseudorandom number generators should also be read after Chaps. 2 and 4.

Chapter 7 on other random sequences can be read for fun, most profitably after Chaps. 2 and 4.

There are exercises at the end of each chapter. Most of these are small programming projects meant to increase familiarity with the material. Exercises that are (subjectively) more difficult will be marked with stars to indicate the level of difficulty.

Example code is in C and/or Python. For Python, we use version 2.7 though earlier 2.x versions should work just as well. Python 3.x will work with minor adjustments.

Intimate knowledge of these programming languages is not necessary in order to understand the concepts being discussed. If something is not easy to see in the code, it will be described in the text. Why C? Because C is a high-performance language that will make the example code all the more useful and because C is the grandfather of most of the common programming languages in current use, including Python.

Each code listing includes the name of the file containing the code. These files are available on the book website:

http://www.numbersandcomputers.com/random/

Note that the code listing in the book may be compressed to save space. The files on the website are as written and include comments and appropriate spacing.

For readers not familiar with C and/or Python, there are a plethora of tutorials on the web and reference books by the bookcase. Two examples, geared toward people less familiar with programming, are *Beginning C* by Ivor Horton and *Python Programming Fundamentals* by Kent Lee. Both of these texts are available from Springer in print or ebook format.

At the end of each chapter are references for the material presented in the chapter. Much can be learned by looking at these references. Almost by instinct we tend to ignore sections like these as we are now programmed to ignore advertisements on web pages. In this former case, resist temptation; in the latter case, keep calm and carry on.

Acknowledgments

In addition to my wife, Maria, I want to thank all of our children: David, Peter, Paul, Monica, Joseph, and Francis. Without your patience and encouragement none of this would have been written. Thank you for providing a living example of a random process. Life is truly more exciting the more it is shared.

Thornton, CO, USA
January 2018

Ronald T. Kneusel
AM+DG

Contents