Numbers and Computers

Ronald T. Kneusel

Numbers and Computers

Second Edition

Springer

Ronald T. Kneusel
Broomfield, Colorado, USA

ISBN 978-3-319-50507-7 ISBN 978-3-319-50508-4 (eBook)
DOI 10.1007/978-3-319-50508-4

Library of Congress Control Number: 2016960676

Printed on acid-free paper

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

To my parents, Janet and Tom, for fostering my love of science.

Preface to the Second Edition

This book is about numbers and how they are represented and manipulated by computers. In this second edition, we expand on the coverage of the first edition by adding three new chapters and new material to several sections.

Broomfield, CO, USA
November 2016

Ronald T. Kneusel

Preface

This is a book about numbers and how those numbers are represented in and operated on by computers.

Of course, numbers are fundamental to how computers operate because, in the end, everything a computer works with is a number. It is crucial that people who develop a code understand this area because the numerical operations allowed by computers and the limitations of those operations, especially in the area of floating point math, affect virtually everything people try to do with computers. This book aims to help by exploring, in sufficient, but not overwhelming, detail, just what it is that computers do with numbers.

Who Should Read This Book

This book is for anyone who develops software including software engineers, scientists, computer science students, engineering students, and anyone who programs for fun.

If you are a software engineer, you should already be familiar with many of the topics in this book, especially if you have been in the field for any length of time. Still, I urge you to press on, for perhaps you will find a gem or two which are new to you.

If you are old enough, you will remember the days of Fortran and mainframes. If so, like the software engineers above, you are probably also familiar with the basics of how computers represent and operate on numbers, but, also like the software engineers above, you will likely find a gem or two of your own. Scientists in particular should be aware of the limitations and pitfalls associated with using floating point numbers since few things in science are restricted to integers.

Students need this book because it is essential to know what the computer is doing under the hood. After all, if you are going to make a career of computers, why would you not want to know how the machine works?

How to Use This Book

This book consists of two main parts. The first deals with standard representations of integers and floating point numbers, while the second details several other number representations which are nice to know about and handy from time to time. Either part is a good place to start, though it is probably best if the parts themselves are read from start to end. Later, after the book has been read, you can use it as a reference.

There are exercises at the end of each chapter. Most of these are of the straightforward pencil and paper kind, just to test your understanding, while others are small programming projects meant to increase your familiarity with the material. Exercises that are (subjectively) more difficult will be marked with either one or two stars (* or **) to indicate the level of difficulty.

Example code is in C and/or Python version 2.7 though earlier 2.x versions should work just as well. 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 low-level language, close to the numbers we will be working with, and because C is the grandfather of most common programming languages in current use including Python. In general, code will be offset from text and in a `monospace` font. 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 e-book format.

At the end of each chapter are references for material presented in the chapter. Much can be learned by looking at these references. Almost by instinct we tend to ignore sections like this 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

This book was not written in a vacuum. Here, I want to acknowledge those who helped make it a reality. First the reviewers, who gave of their time and talent to give me extremely valuable comments and friendly criticism: Robert Kneusel, M.S.; Jim Pendleton; Ed Scott, Ph.D.; and Michael Galloy, Ph.D. Gentlemen, thank you. Second, thank you to Springer, especially my editor, Courtney Clark, for moving ahead with this book. Lastly, and most importantly, thank you to my wife, Maria, and our children: David, Peter, Paul, Monica, Joseph, and Francis. Without your patience and encouragement, none of this would have been written.

Broomfield, CO, USA Ronald T. Kneusel
December 2016 AM+DG

Contents