
Preface

Purpose

The purpose of this text is to provide the reader a concise introduction to data structures and associated algorithms. This text is intended for a second-semester course in programming using the Java programming language prior to a course on advanced data structures and algorithms. It is a continuation of the text *Guide to Java: A Concise Introduction to Programming using Java* written by the authors and published by Springer Verlag London Limited [2].

Comparison and Need for This Text

There are a number of introductory texts on data structures using the Java programming language. Many of these texts are very comprehensive, but unfortunately, they sometimes seem to cover so many details that might make it difficult for a beginning programmer to discern which concepts are the most relevant. The old adage that “one can’t see the forest for the trees” might apply in some instances because if one is too busy with the many details when trying to learn data structures one might not have a clear grasp of the fundamental concepts necessary to completely understand the material. There are also some shorter texts, but sometimes they seem to cover important concepts very quickly which can confuse readers too. This text attempts to fulfill the need for an introduction to data structures by helping the reader to concentrate on the fundamentals which in turn allows the text to be more concise and help the reader remain focused on the key concepts. The result is that the reader can learn data structures quickly and also have a good foundation to learn more complex topics later.

Features of This Text

As mentioned above, this text is a concise introduction to data structures which is accomplished by concentrating on the fundamentals. This text is written in the same style as the previously mentioned *Guide to Java* text. It provides many examples and illustrations and uses visual contour diagrams to illustrate object-oriented semantics as needed. Also as before, in some paragraphs of the text, questions are asked of the reader to help them interact with the material and think about the subject matter presented.

The text starts with data structures using arrays to reinforce array concepts learned previously and then introduces linked data structures to compare with array-based structures. A data structure is first introduced with a simple data type to help with understanding basic concepts, and then, it is reinforced using generic data types. In addition, elementary algorithm analysis is introduced and discussed as needed throughout the text.

To help further reinforce concepts, each chapter has one or more complete programs to illustrate many of the concepts presented and to also help readers learn how to write programs on their own. In addition, for review and practice, there are summaries and exercises provided at the end of each chapter. Further, in the appendices at the end of the text, there are answers to selected exercises and a glossary of important terms. A summary of all these features is listed below:

- Stresses the fundamentals.
- Provides many examples and illustrations.
- Begins with array-based data structures to reinforce array concepts learned previously.
- Follows with linked data structures for comparison and to reinforce methods.
- Uses both primitive and generic data types in each chapter.
- Uses contour diagrams to illustrate object-oriented concepts.
- Asks readers questions to help them interact with the material.
- Contains one or more complete programs in every chapter.
- Provides chapter summaries.
- Includes exercises at the end of each chapter, with selected answers in an appendix.
- Has a glossary of important terms.

Overview of the Chapters

After an overview of preliminary concepts, this text introduces stacks and queues using arrays along with a discussion of array-based lists. This is followed by an introduction to linked lists and the implementation of stacks and queues using references. Next, there is an introduction to binary trees, a discussion of various

sorting techniques, heaps, and hashing. The appendices include a glossary and answers to selected exercises. Lastly, there is a reference and useful Web site section and an index. The following provides a brief synopsis of the chapters and appendices:

- Chapter 1 reviews and discusses various preliminary concepts.
- Chapter 2 introduces stacks using arrays.
- Chapter 3 illustrates queues using arrays.
- Chapter 4 discusses lists using arrays.
- Chapter 5 introduces lists using references and objects.
- Chapter 6 examines linked lists.
- Chapter 7 explores stacks and queues using references.
- Chapter 8 introduces binary trees.
- Chapter 9 explores sorting algorithms.
- Chapter 10 discusses heaps.
- Chapter 11 introduces hashing.
- Appendix A contains a glossary of key terms.
- Appendix B provides answers to selected exercises.

Note that the above order is the authors' preferred sequence; however, it is understood that some instructors, professionals, and independent students might want to pursue some topics in a different order. As given, all linked list structures follow the array-based structures, but alternatively, one could have Chaps. 5 and 6, following Chap. 1, and then, Chaps. 2–4 could be examined later. Further, parts of Chap. 9 can be introduced earlier at the instructor's discretion. Of course, other combinations are possible given the preference of the instructor or the reader's background.

Scope

As mentioned previously, this text concentrates on the fundamentals of data structures such as stacks, queues, lists, (using both arrays and links), sorting, and elementary binary trees, heaps, and hashing. Since it concentrates on the fundamentals, it might not cover all the details that are found in some other texts, and if necessary, these topics can be supplemented by the instructor or reader, or covered in a subsequent text and/or course.

Audience

This text is intended primarily for readers who have had a previous course or used a text in programming using Java. Some of the concepts and specific skills needed are discussed and reviewed in Chap. 1. As mentioned previously, this text sequentially

follows the authors' previous text *Guide to Java*, but this does not preclude a reader from having used other texts as well. Note that although one might be able to read this text with knowledge of another language such as C++, this text does not review the Java programming language and previous knowledge of Java is highly recommended. In addition to being a classroom text for a second-semester course in programming in preparation for a subsequent advanced course in data structures and algorithms, it can be used as a self-study guide in either academe or industry.

Acknowledgements

The authors would like to thank the reviewers Mark E. Bollman of Albion College, James W. Chaffee of the University of Iowa, and Naomi E. Hahn of Illinois College. Also, the authors would like to acknowledge the students of Illinois College who have read and used various sections of the text in classroom. On a personal note, James Streib would like to thank his wife Kimberly A. Streib and son Daniel M. Streib for their patience, and Takako Soma would like to thank her family and friends, near and far.

Note that Java is a registered trademark of Oracle and/or its affiliates and that Windows is a registered trademark of Microsoft Corporation in the USA and/or other countries.

Feedback

The possibilities of errors exist in any text, and therefore, any corrections, comments, or suggestions are welcome and can be sent to the authors via the e-mail addresses below. In addition to copies of the complete programs presented in the text, any significant corrections can be found at the Website below.

Website: <http://www.jtstreib.com/GuideDataStructures.html>

September 2017
Jacksonville, IL, USA

James T. Streib
james.streib@jtstreib.com
Takako Soma
tsoma@ic.edu

Guide to Data Structures

A Concise Introduction Using Java

Streib, J.T.; Soma, T.

2017, XIII, 376 p. 43 illus., 9 illus. in color., Softcover

ISBN: 978-3-319-70083-0