

This Web site contains the book's contents and preface, answers to selected exercises, and Java code for all ADTs and case studies. New features will be added from time to time.

## Case studies

Case studies are an important feature of this book. They can be found in Chapters 6–9, 11, 13–15, and 17.

Each case study leads the reader through the object-oriented design and implementation of a fair-sized application program. The presentation focuses on the selection of an ADT appropriate for the application, and later on the choice of the most suitable implementation of that ADT (from among those presented earlier in the chapter). The object-oriented design of the application program itself, and the development of algorithms to be employed by the application program, are also discussed. User interface issues are generally omitted here, however.

Complete Java source code for all case studies may be found at the companion Web site. For some case studies, there is an application program that can be downloaded and run. For others, there is an applet that can be run immediately using a Java-enabled Web browser. Where appropriate, each case study comes in two versions, one using the classes developed in this book and the other using the Java collection classes.

## Exercises

Each chapter is followed by a set of exercises. Some of these exercises are drill, or simple modifications to designs, algorithms, or implementations presented in the chapter. The more challenging exercises are clearly marked as \* (hard) or \*\* (harder).

Sample answers to selected exercises can be found at the companion Web site.

## About the authors

David Watt is a Professor of Computing Science at the University of Glasgow. He has 26 years' experience of teaching undergraduates and graduates, in programming, algorithms, data structures, software engineering, and other topics. He developed the material of Chapters 1–12 for a course in algorithms and data structures, first taught in early 1999. His research interests are design, specification, and implementation of programming languages.

Deryck Brown is a Senior Lecturer in Computing Science at the Robert Gordon University, Aberdeen. He has 6 years' experience of teaching undergraduates and graduates, in object-oriented programming, software engineering, and several other topics. His research interests are specification and implementation of programming languages, and genetic algorithms.