METADATA

• • •

Abstract

The book is mainly aimed at undergraduate students who study object oriented programming for the first time. It assumes that the reader is familiar with procedural programming, though. First, the concepts of object oriented programming, the required mechanisms and the paradigm properties are presented, independently of a specific programming language. The aim is to highlight the concepts' universality. The book, then, focuses on the C++ language, presenting the above using C++. It begins by summarizing procedural programming constructs, with an emphasis on specialized ones. Then, the concept of names' scope is analyzed, stepping towards data abstraction. Talking about data abstraction, the one of the two main areas in object oriented programming, class and object definitions are discussed, as well as related topics such as visibility declarations to hide implementation details, constructors and destructors of objects as well as object assignments.

Class composition is separately discussed. Next, a thorough presentation of the second important area in object-oriented programming, which is inheritance, follows. The C++ related topics are completed by some other class related issues of the language that are considered worth mentioning. Emphasis is put on static class members, which allow a smooth transition from the C++ to the Java language. C++'s input/output system is outlined, also as an interesting example of using object oriented programming. Generic programming is also briefly presented. The book then switches to the Java language and presents its object oriented features. Next, there is a summary of object oriented characteristics of other object oriented programming languages, not necessarily procedural. The book concludes with complex problems, both for personal engagement and practice, but also as appropriate examples where the use of object oriented programming is needed.