

# Preface

The C++ programming language was introduced by Bjarne Stroustrup of the AT&T laboratories in 1985 as an extension of C, with additional features borrowed from the esoteric language *Simula*. Since then, C++ has grown rapidly in response to the practical need for a programming language that is able to efficiently handle composite and diverse data types. The language implementation is pivoted on the ingenious concept of *object oriented programming* (OOP). Today, C++ dominates the commercial market and is favored among system programmers and application developers.

## Object oriented programming

To illustrate the advantages of an object oriented programming language compared to a structured language such as MATLAB, FORTRAN 77, or C, we assume that an international sports competition has been entered by runners from many countries around the globe. The record of each runner consists of several fields including name, country of origin, city of birth, date of birth, and best performance time.

In a structured language, each one of these fields is normally registered in a separate data vector. In an OOP language, each runner becomes an object defined as a member of the class of runners, and each member is described by the collection of these fields. This formalism allows us to record, recall, and manipulate in any desired way the personal data of each runner using simple symbolic operators. Sub-classes consisting, for example, of runners of a particular nationality can be readily defined to facilitate more detailed manipulations.

An OOP language allows us to introduce a data type of our choice viewed as an object in a defined class, and then use the class as a building block for further development. This flexibility essentially allows us to build a language without building a compiler. In this sense, an OOP language is an ultimate language.

## C and C++

C++ is a generalization of C, but accomplishes much more than C, to the extent that it should be regarded, studied, and taught as a separate language. It is neither necessary nor recommended to study C as a prerequisite of C++, though knowledge of C can be helpful.

## This book

This book is a brief and basic introduction to C++ for everyone and especially for scientists and engineers. The text offers a venue for effectively teaching and rapidly learning the language at the level of an undergraduate course in any discipline of the physical sciences and computer science and engineering. The discussion illustrates step-by-step the grammar, syntax, and main features of the language, and explains the basic premise of OOP with an emphasis on scientific computing.

Chapter 1 reviews basic concepts of computer hardware software and programming.

Chapters 2 and 3 outline the general features of C++ and the basic implementation of the language.

Chapter 4 discusses user-defined functions with an emphasis on scientific computing.

In Chapter 5 introduces pointers to memory addresses and demonstrates their applications.

Chapter 6 explains the basic principles of object oriented programming (OOP) and the implementation of classes.

Chapters 7 and 8 discuss graphics and graphical user interface (GUI) programming based on the fabulous VOGLE library for the X11 server, and on the GLUT, GLUI, and GTK+ utility toolboxes.

Chapter 9 demonstrates the use of MATLAB functions from C++ code for numerics and graphics.

## Transition to C++

Many students, scientists, engineers, and other professionals are familiar with the general concepts of computer programming, are proficient in an easy programming language, such as MATLAB or FORTRAN 77, and would like to learn C++. This book is ideally suited for this audience. Translation tables demonstrating the conversion of MATLAB or FORTRAN 77 code into C++ code are given in an appendix. A side-by-side comparison illustrates the syntactic and functional differences between the three languages.

## Keeping it simple

The C++ language is pluralistic in two ways. First, it allows different commands (tasks) to be stated (implemented) in alternative ways. Second,

it supports several dialects dependent on the chosen compiler. All compilers support the ANSI/ISO standard C++ functions discussed in this text.

In our discussion, structure and forms that make for a transparent and efficient, but not necessarily compact, programming style are adopted. Code obfuscation is avoided at all cost.

## Learning from the Internet

This text was written with a new learning model in mind: study a basic text or take a short course to get acquainted with a subject, and then use the Internet to master the subject. A wealth of up-to-date resources and tutorials are available on the Internet on every imaginable subject.

Study this text to get acquainted with C++, and then use the Internet to master the language.

## Book Internet site

This book is accompanied by a library of programs that can be freely downloaded from the Internet site:

<http://dehesa.freeshell.org/ICPPPG>

Further information on C++ and links of interest are also provided.

## Unix

A C++ programmer without Unix experience is handicapped in many ways. A number of Unix operating systems are freely available and can be readily installed either by themselves or in a dual boot mode along with Windows on desktops and laptops. Examples include *Fedora Core*, *CentOs*, and *BSD*. Appendix A summarizes the basic Unix commands.

## cygwin for Windows users

The software package *cygwin* allows Windows users to work in a Unix environment and utilize Unix libraries and applications on top of the windows operating system. Effectively, *cygwin* creates a computer running Unix inside another computer running Windows. To distinguish between the two, we refer to the former as an “environment.” MATLAB users are familiar with the concept of a computing environment. *cygwin* derives its name from three components:

1. gnu: standing for “GNU’s Not Unix”. This is a free, open-source operating system consisting of a kernel, libraries, system utilities, compilers,

and end-user applications. Its development was announced by Richard Stallman in 1983.

2. Cygnus: a genus of beautiful birds.
3. Windows: an operating system produced by the Microsoft corporation.

The *cygwin* package can be freely downloaded and easily installed from the Internet site <http://www.cygwin.com>. The package contains a wealth of applications and tools, including the X11 graphics library and a C++ compiler.

Windows users are strongly advised to download and install the package as a prelude to studying this book.

## Acknowledgment

I am grateful to Todd Porteous, Conrad Palmer, and Micheal Waltz for providing hardware, software, and moral support.

I am grateful to anonymous reviewers and to my editor Valerie Schofield who immediately recognized the significance of this book and provided useful comments.

C. Pozrikidis

San Diego, March 2007



<http://www.springer.com/978-0-387-68992-0>

Introduction to C++ Programming and Graphics

Pozrikidis, C.

2007, XII, 372 p., Hardcover

ISBN: 978-0-387-68992-0