Your working environment is very important. If you are to do a lot of programming, optimise your program development and enjoy your programming as much as possible, then you naturally need the best environment. This chapter attempts to highlight a few key requirements and tools that you should look for in your environment by providing an overview of the Borland C++ for Windows compiler and development suite.

2.1 Hardware, Software and Setup Used

All of the program code presented throughout this book was written and prepared with the help of the following hardware and software.

2.1.1 Hardware

An IBM-compatible desktop PC with an Intel Pentium Pro 200 MHz processor, 256 Mbyte RAM, 8 Mbyte video RAM, 10 Gbyte hard disk and an SVGA monitor was used exclusively for all development. This specification was found to be an excellent platform for all of the programs presented in this book, with quick compile and link times.

2.1.2 Software

The compiler used was Borland's Borland C++ for Windows, version 5.x (2000). Why Borland? Basically, because I have used the Borland C++ compiler since 1990, when it was (and still is, in my opinion), the leader in commercial C++ compilers for the PC. Incidentally, Borland first released their C++ compiler in May 1990. Although it is not for me to state which is the best C++ compiler on the market for your particular applications, you should, if you can, simply choose the best compiler and environment to suit your needs and style.

No keywords, libraries, container classes, ObjectWindows Library (OWL) classes or features specific to the Borland compiler have been used in this book and all programs provided on the
supplied disk should (in theory) be portable to any preferred draft ANSI/ISO C++-compatible compiler.

2.2 The Borland Integrated Development Environment

Commercial compilers such as Borland C++, Microsoft Visual C++ and Symantec C++ for Windows all support excellent development environments. This section presents a brief overview of the Borland C++ compiler and development tools.

The Borland Integrated Development Suite is an excellent environment to work in (Fig. 2.1). Apart from the Compiler, Linker and Librarian, it includes such Windows tools as a customisable Editor, SpeedBar, Browser, Debugger, WinSight and WinSpector, AppExpert, ClassExpert, Project Manager, and Resource Workshop. Command-line tools such as MAKE, TLINK and TLIB are also available. Borland C++ also fully implements the ANSI C standard with numerous additions. The list of tools and features is endless. Let us take a brief look at a few of the key features offered by the Borland C++ Integrated Development Environment (IDE).

2.2.1 Editor

Figure 2.1 illustrates a typical editing session. Since the client area of the IDE is a Multiple Document Interface (MDI), we can have multiple files open simultaneously. The editor is completely customisable.

![Fig. 2.1 The Borland IDE.](image-url)