

Oberon-2 Programming with Windows

Springer

Berlin

Heidelberg

New York

Barcelona

Budapest

Hong Kong

London

Milan

Paris

Santa Clara

Singapore

Tokyo

Jörg R. Mühlbacher Bernhard Leisch
Brian Kirk Ulrich Kreuzeder

Oberon-2 Programming with Windows

With 66 Figures



Springer

Jörg R. Mühlbacher
Bernhard Leisch
Ulrich Kreuzeder

Johannes-Kepler-Universität Linz
Forschungsinstitut für Mikroprozessortechnik (FIM)
A-4040 Linz, Austria

Brian Kirk
Robinson Associates
Red Lion House
St. Mary's St.
GL6 6QG Painswick, Glos., UK

Library of Congress Cataloging-in-Publication Data

Oberon-2 programming with Windows / Jörg R. Mühlbacher ... (et al.).

p. cm.

Includes bibliographical references and index.

ISBN-13: 978-3-540-62522-3

1. Oberon. 2. Object-oriented programming. 3. Operating systems
(computers) I. Mühlbacher, Jörg, 1946-

QA76.76.063024 1997

005.13'3--dc21

97-8043

CIP

ISBN-13: 978-3-540-62522-3

e-ISBN-13: 978-3-642-45762-3

DOI: 10.1007/978-3-642-45762-3

Additional material to this book can be downloaded from <http://extras.springer.com>

© Springer-Verlag Berlin Heidelberg 1997

The use of general descriptive names, trademarks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

Cover Design: d&p, Heidelberg,

Typesetting: Camera ready by authors

SPIN 10561309 45/3142 – 5 4 3 2 1 0 – Printed on acid-free paper

Preface

This book is aimed at students who need to learn the basics of programming or who are studying computing. It is a “hands on” book containing many examples which start by illustrating basic Oberon-2 language features and gradually increase in scope to cover object-oriented programming concepts and constructs. Oberon-2 is a successor to the language Pascal, which was also designed by Prof. N. Wirth [Wir71]. It has quickly become a major language used for teaching purposes. The only thing you need for successfully working through the book is to have access to a computer running Windows 3.11 or Windows 95. The material in the book is useful to students of schools, colleges, and universities for teaching Oberon-2 and programming at an introductory level.

The scope of the book is not focused on software engineering or object-oriented technology; other books mentioned in the reference section already cover these topics in much greater depth. However, the examples in the book have been designed with these topics firmly in mind. Currently the term “object-oriented” is very much in fashion, having taken over from structured programming of the 1970s and ’80s. In this book we have taken the view that a structured programming approach can be used to teach the fundamentals of programming algorithms. The object-oriented approach is then brought in as a complementary way to think, analyze, design and program. It also provides richer techniques for achieving encapsulation and reuse of program implementations. Oberon-2 is often referred to as a hybrid language as it provides elegant support for programming using either structured or object-oriented programming, or indeed any practically useful combination of these two approaches.

A CD-ROM containing *POW!* software for Windows is provided with the book. It consists of a programming support environment including an Oberon-2 compiler, libraries, and the source code for the examples. The software is royalty free for educational use and so may be freely used within educational establishments. The first two chapters of the book describe the installation of the *POW!* Borland-like environment and how to work with the integrated text editor, which includes many features to support the development of programs written in Oberon-2. The facility for creating and linking Windows DLLs (Dynamic Link Libraries) makes it possible to use existing DLL based libraries and to interoperate with other Windows programs written in languages other than Oberon-2, for example C and C++.

In the text each Oberon-2 language construct is described using text, syntax diagrams, and also EBNF. This semi-formal definition of the language syntax can be skipped on first reading; full definitions and the syntaxes of the Oberon-2 and EBNF languages are given in the appendices.

We would like to encourage you not just to read the book but to use the *POW!* system and the examples on the CD-ROM to work through the book.

This way you will gradually gain confidence and ultimately be encouraged to start designing and writing your own object-oriented programs.

POW! and Oberon-2 for Windows originate from a joint project between FIM and the Johannes Kepler University, Linz, Austria, and Robinson Associates in the UK. This book evolved from an earlier work of three of the now four authors [MLK95].

We would like to thank the following people for their advice and contributions to *POW!* and the production of this book:

Rudolf Hörmanseder and Peter Dietmüller for their comments and suggestions on the manuscript.

Professor Hanspeter Mössenböck for permission to include the Oberon-2 Language Report as an Appendix and for his valuable contributions.

Günther Schwarz and Ernst Unger from the Federal Institute for Teacher Training for encouraging the use of *POW!* Oberon-2 in schools.

Students of computer science at the University of Linz who have used *POW!* widely and provided feedback which has contributed to many improvements.

Linz
Painswick
1997

J. R. Mühlbacher
B. Leisch
B. R. Kirk
U. Kreuzeder

Table of Contents

1. Introduction	1
2. First Steps	7
2.1 Installation of <i>POW!</i>	7
2.2 Operating the system.....	7
2.3 A first program.....	11
3. The Working Environment of <i>POW!</i>	15
3.1 Basic ideas	15
3.2 Editor.....	15
3.3 Project management.....	19
3.4 Templates.....	24
3.5 Button bar.....	27
3.6 Configuration	28
3.7 Working with windows.....	31
3.8 Tools	33
3.9 On-line help.....	36
3.10 Menu functions	37
3.11 Typical errors	40
3.12 Advanced topics	42
4. Introduction to Oberon-2	49
4.1 Introduction.....	49
4.2 Basics of a language.....	49
4.3 Basic language elements	52
4.4 Oberon-2 programs under <i>POW!</i>	57
4.5 Declarations	60
4.6 Statements	63
4.7 Expressions and assignments	64
4.8 Selection.....	70
4.9 Repeated execution (<i>Iteration</i>).....	76
4.10 Structured data types.....	82
4.11 Procedures.....	90
4.12 Modules.....	108
4.13 Pointer types.....	125
4.14 Recursion	149
4.15 Exercises	173

5. Object-Oriented Programming	175
5.1 Introduction.....	175
5.2 From abstract data types to classes	175
5.3 Concepts.....	182
5.4 Design	194
5.5 Application Examples	195
5.6 An object-oriented project.....	208
5.7 Retrospective and summary	238
5.8 Exercises	240
6. Oberon Portable Applications Library	243
6.1 What is a library?	243
6.2 Overview of the modules	243
6.3 Notes on the Windows implementation	244
6.4 The module ColorPlane.....	245
6.5 The module Display	251
6.6 The module File	259
6.7 The module Float	269
6.8 The module In	274
6.9 The module Out	277
6.10 The module Param	278
6.11 The module Print.....	280
6.12 The module Process	282
6.13 The module Strings	283
6.14 The module Utils.....	287
6.15 The module Volume.....	291
6.16 The module XYplane	296
7. Programming with the Windows API	299
7.1 Overview	299
7.2 Language extensions	299
7.3 Interfaces to “foreign” DLLs	300
7.4 Basic Windows program	302
7.5 The Mini-resource compiler.....	306
Appendix A: Oberon-2 for Pascal programmers	313
A.1 Constants.....	313
A.2 Operators.....	314
A.3 Data types.....	314
A.4 Arrays.....	315
A.5 Strings	315
A.6 Open Arrays	316
A.7 Pointer types.....	316
A.8 Memory management.....	317

A.9 Compound data types	318
A.10 Extensions of record types	318
A.11 Procedure types	319
A.12 Control structures	320
A.13 Procedures	322
A.14 Type-bound procedures	322
A.15 Modules	323
A.16 Export of identifiers	324
A.17 Programs in Oberon-2	325
A.18 Object-oriented programming	326
Appendix B: The Programming Language Oberon-2	329
B.1 Introduction	329
B.2 Syntax	329
B.3 Vocabulary and Representation	330
B.4 Declarations and scope rules	331
B.5 Constant declarations	332
B.6 Type declarations	333
B.7 Variable declarations	336
B.8 Expressions	336
B.9 Statements	340
B.10 Procedure declarations	345
B.11 Modules	350
B.12 Definition of terms	352
B.13 Syntax of Oberon-2	354
B.14 The module SYSTEM	355
B.15 Windows related language extensions	357
Appendix C: Table of ASCII Codes	359
Literature	361
Index	363