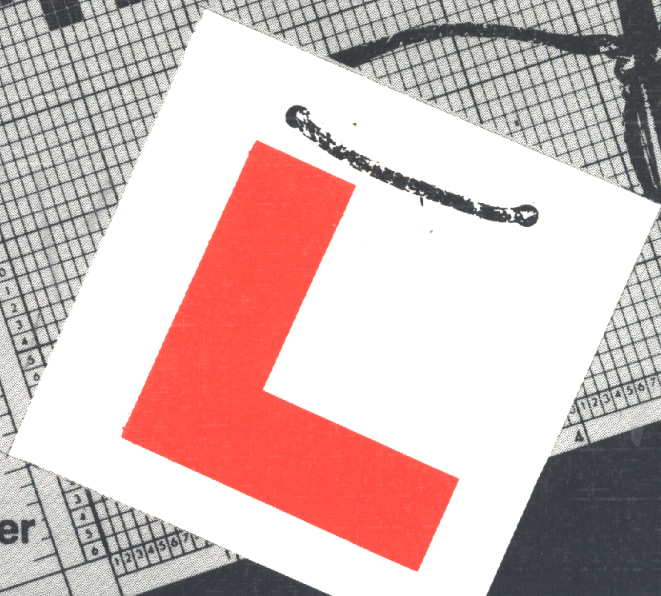




BIBLIOTHEQUE DU CENTRE

Programming Techniques and Practice



Alan Chantler

Programming Techniques and Practice

Alan Chantler

PUBLISHED BY NCC PUBLICATIONS

4581

British Library Cataloguing in Publication Data

Chantler, Alan

Programming techniques and practice

1. Electronic digital computers – Programming

I. Title

001.64'2

QA76.6

ISBN 0-85012-338-0

© THE NATIONAL COMPUTING CENTRE LIMITED, 1981

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system; or transmitted, in any form or by any means, without the prior permission of The National Computing Centre Limited.

First published in 1981 by:

NCC Publications, The National Computing Centre Limited,
Oxford Road, Manchester M1 7ED, England.

Typeset in 10pt Times Roman and printed by
UPS Blackburn Limited, 76-80 Northgate, Blackburn, Lancs.

ISBN 0-85012-338-0

Acknowledgements

The production of this book would not have been possible without the encouragement and support of many people. It would not be appropriate for me to attempt to mention them all by name. I would, however, offer a special thank-you to Mike Eaton. I am also grateful to all those colleagues and students whose bad programming has encouraged me to find a better way.

My greatest indebtedness is to Margie for her continuous support and interest, mostly unconnected with programming.

BIBLIOTHEQUE DU CERIST

Preface

The increasing use of computers of all sizes in a proliferation of application areas brings with it a need for more and more skilled and professional programmers. In order that the available personnel can be made more productive, it is necessary that they receive appropriate training and practice, involving the study and use of the most up-to-date, well proven techniques. No amount of training, however good it may be, will compensate for a lack of professionalism and of a dedication to perfection. For many years, there have been two identifiable and overlapping groups of persons associated with programming – those who write programs (the larger group) and programmers. This book is intended to help both groups to become synonymous. It may also help to encourage non-programmers to become programmers, by illustrating how a dedicated, systematic approach not only increases the chances of success, but also makes the job easier.

To gain most benefit from this text, the reader should have undergone some initial instruction in programming, including the production of some simple programs in BASIC or a similar language. Ideally, the text should be supported by the encouragement of a course leader, although much benefit can be gained by the 'self study' reader. All readers should attempt the suggested practical examples before comparing their answers with those given.

The techniques and practices described in this book are essentially practical in nature. They have been used by the author and others for a number of years, both in teaching students and in the production of real-life commercial software. By following them, it is possible to produce programs which work first time, continue to work and are easy to main-

tain. Thus, we can at last dispense with the myth, still widely taught, and even more widely believed, that 'you can't expect a program to work first time', an opinion which, I suggest, no-one would be prepared to accept about a surgical operation.

Alan Chantler
Yelvertoft
June 1981

Contents

	Page
Acknowledgements	
Preface	
1 Introduction	1
What is Good Programming?	1
Program Descriptions	3
Conclusions	9
2 The Systems Development Life Cycle	13
The Stages of a System	14
The Elements of the Life Cycle	15
3 The Program Specification	21
Scope of the Specification	22
Checking the Specification	24
Three Example Program Specifications	24
Project	27
4 Descriptions of Data	31
Concept of Data Type	31
Numeric Data	32
Range of Numeric Data	32
Accuracy of Data	33
Non-numeric Data	33

Character Strings	34
Logical Values	34
The Structure of Data	34
Describing a Data Structure	36
Practical	41
5 Data Manipulation Techniques	43
Table Searching	43
Sorting of Data	49
A Simple Internal Sort	50
6 Systematic Program Design	55
The Universal Program	55
An Example	58
7 Checking a Program Design	65
Dry-running	67
Performing a Dry-run	67
The Walk-through: A Catalytic Approach	68
Catalytic Checking	69
Independent Inspection	69
'Egoless' Programming	69
8 Coding Practice	71
Program Readability	71
Program Maintainability	73
Coding Techniques	73
Some General Points	81
Practice	84
9 Testing and Debugging	85
Levels of Correctness	85
Devising a Test Strategy	86
Use of 'Dummy Stubs'	87
Description of Tests	88
Choice of Test Data	89
Identification of Errors	91

Location of Errors	91
Correction of Errors	92
Debugging Aids	92
Help from the Operating System	94
Interactive Debugging	94
10 Program Implementation	97
Source File Creation	97
Role of the Compiler	99
Compile-time Errors	99
Source File Maintenance	100
Interactive Discipline	101
11 Program Release and Maintenance	105
Field Trials and Parallel Running	106
Changeover Problems	107
Maintenance	107
Program Reviews	109
12 Software Aids and Utilities	111
Program File Creation and Maintenance	111
Macro Processors and Preprocessors	112
Data Generators	113
SORT Utilities	115
Report Generators	116
Trace and Snapshot Software	118
Flowcharters	118
Applications Packages	118
13 Communicating with the Computer	121
Early Systems	121
Multiprogramming	123
Operating System Facilities	124
Multi-access Working	127
Conclusions	127

14 Program Logic Description	129
Flowcharts	129
The Sequence	132
The Selection	133
The Iteration, or Loop	134
Decision Tables	136
Processing a Decision Table	142
Practical Exercise No. 1	143
Alternative Methods	144
Practical Exercise No. 2	147
15 Programming Languages	149
Essential Features	151
Data Description	151
Data Manipulation	152
Data Communication	153
Sequence Control	154
Optional Extras	155
Data Description and Manipulation	155
Multiple-choice Selection	156
The Myth of the Universal Language	156
Appendix	
1 Practical Solutions	159
Index	173