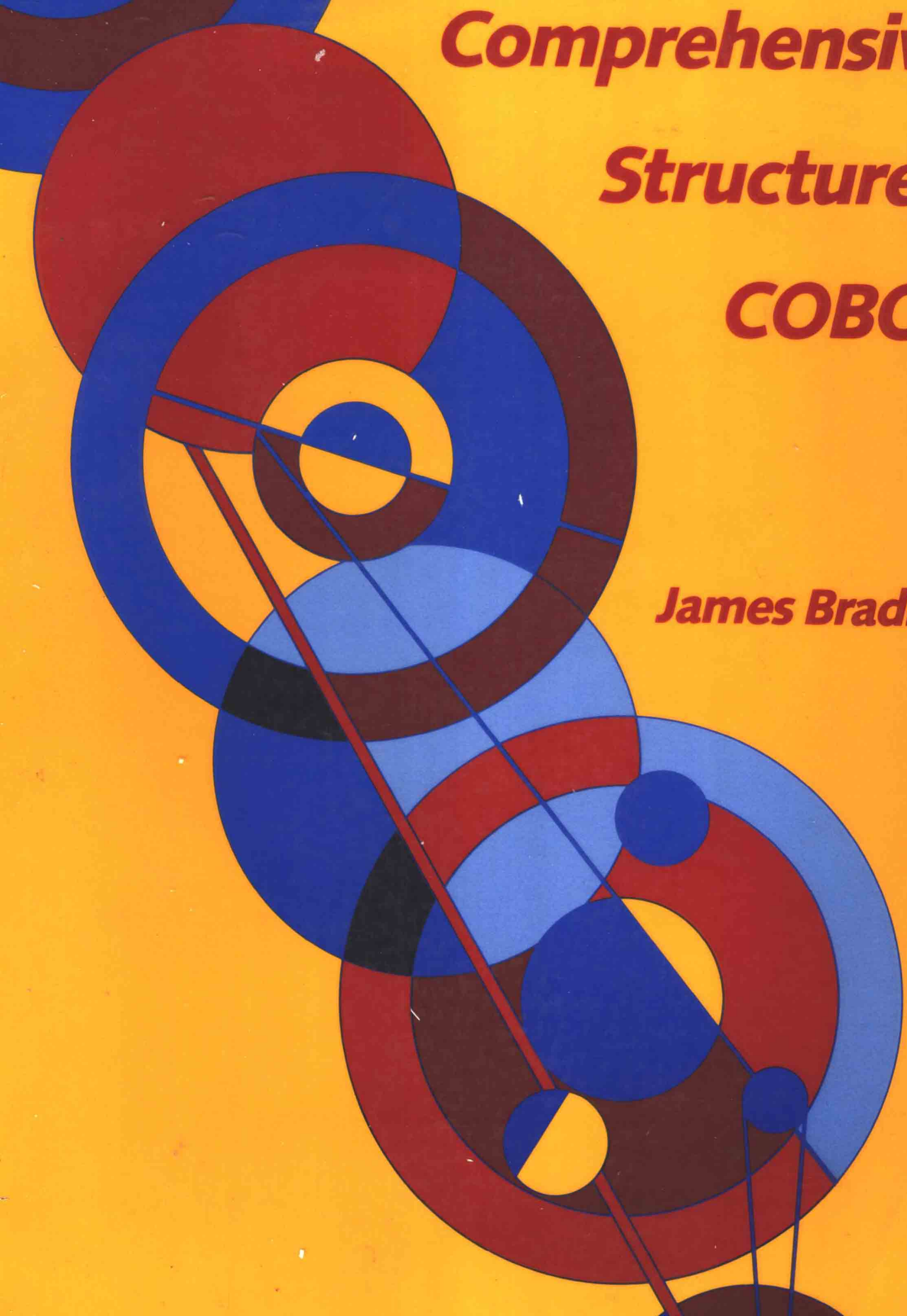
Comprehensive Structured COBOL



James Bradley

Comprehensive Structured COBOL

James Bradley

University of Calgary

Mitchell McGRAW-HILL

New York St. Louis San Francisco Auckland Bogotá Caracas
Hamburg Lisbon London Madrid Mexico Milan Montreal
New Delhi Oklahoma City Paris San Juan São Paulo
Singapore Sydney Tokyo Toronto Watsonville

Comprehensive Structured COBOL

1 2 3 4 5 6 7 8 9 0 WEB WEB 9 3 2 1 0

ISBN 0-07-007078-4

This book was set in Syntax and Berkeley Old Style by TBH/Typecast. The editors were Karen M. Jackson and Raleigh Wilson; the designer was Randall Goodall, Seventeenth Street Studios; drawings were done by Joseph Maas, Advanced Presentations. Development and production management were provided by Cole and Associates. Webcrafters was the printer and binder.

Library of Congress Cataloging-in-Publication Data

Bradley, James, 1942–
Comprehensive structured COBOL / James Bradley.
p. cm.
ISBN 0-07-007078-4
1. COBOL (Computer program language) 2. Structured programming.
I. Title.
QA76.73.C25B73 1990 89-13743
005.13'3--dc20 CIP

PREFACE

COBOL REMAINS A MOST important programming language, with more than 75 percent of business application programs still being written in COBOL. COBOL does not, however, stand still, but continues to develop in light of experience and advances in programming theory. COBOL 85 compilers are now commonplace on both mainframes and microcomputers. While this book teaches COBOL 85 (COBOL 85 facilities are clearly marked throughout the book), it also provides a complete presentation of COBOL 74.

The major theme of *Comprehensive Structured COBOL* is that learning COBOL and the ideas associated with it should be both challenging and interesting. *Comprehensive Structured COBOL* emphasizes understanding COBOL and programming concepts as the key to becoming a successful COBOL programmer. Thus, the amount of explanatory material in the text is greater than would normally be expected, and many examples are given to strengthen the concepts and techniques presented. The student is taught to understand, in a commonsense manner, the "why" of COBOL concepts and programming techniques.

Explanations and analogies used in the book have been verified for complete accuracy. Having been an instructor in computer programming for more than 15 years and having been involved in computer-related research and commercial programming for some 20 years, I have experienced the ideas inherent in COBOL 85 from many different viewpoints.

This book was written with the goal of making it easy to read and understand and with the hope that the ideas will leap off the page at you. I have been encouraged by reviewers who are of the opinion that this writing goal has been successfully met. However, *Comprehensive Structured COBOL* does not neglect detail. The book includes a level of COBOL detail that should make learning COBOL an achievable goal of every student.

In addition to the movement from COBOL 74 to COBOL 85, exciting as that is, another transition is taking place. Computer file processing, the original domain of COBOL applications programming, is moving to relational data base processing using COBOL with embedded SQL.

Every student of COBOL must learn the difference between file processing and relational data base processing and must develop the ability to write COBOL/SQL programs for processing relational data bases. Relational data base systems that can interface with COBOL are now commonly available, and systems for use with supermini computers and workstations can be purchased at a nominal cost. For example, at the University of Calgary, where the relational system ORACLE runs on supermini computers, workstations, and personal computers, the mainstream COBOL 85 course includes assignments on the use of COBOL with SQL for data base processing. That relational data base processing using COBOL with embedded SQL will become a strong component of COBOL courses in the 1990s seems clear, and *Comprehensive Structured COBOL* contains the necessary COBOL/SQL material.

Organization and presentation

The book is organized into three sections:

I. COBOL Foundations

The first section deals with the foundations of COBOL and presents the divisions of COBOL programs, as well as the use of IF-statements and looping with the PERFORM verb. The first chapter assumes that the student has no prior programming experience and explains computer processing and elementary data processing concepts, together with elementary COBOL programming concepts. The student with programming experience can begin either in Chapter 1 with COBOL programming concepts or with Chapter 2.

Chapters 2 through 4 explain the common considerations in writing the four divisions of a COBOL program. In addition, Chapter 4, which deals with the PROCEDURE DIVISION, details the use of the common COBOL verbs. Many program examples are used throughout these three chapters to illustrate the concepts and COBOL constructs presented.

The IF-statement and the new EVALUATE statement are given an entire chapter (Chapter 5), with particular attention paid to understanding conditions, both simple and compound, and to nested IF-statements. Chapter 6 is devoted to looping, including nested loops, using the PERFORM . . . UNTIL and PERFORM . . . TIMES verbs. Structured programming is used throughout the first section, and principles are explained gradually at appropriate points in the text.

II. COBOL Processing Methods

The second section is devoted to COBOL programming methods. I believe that the order chosen is the best one, although it could easily vary

according to the wishes of the instructor or experiences of the students. The section begins with a chapter on printing simple reports, followed by a chapter on validation of input data. Then comes a chapter on the COBOL SORT verb as a preliminary to the next chapter, which presents the all-important topic of control-break processing for generating more complex reports. The final two chapters of the section cover programming techniques for single- and multiple-level arrays.

With the possible exception of the use of the Report Writer (in Chapter 20 in the third section), material presented in the first two sections is generally covered in many extensive one-semester introductory COBOL courses.

III. COBOL for File and Data Base Processing

The final section teaches file and data base processing using COBOL. Chapter 13 covers sequential file processing and file updating techniques in detail; Chapter 14 details indexed sequential files. VSAM is used in programming examples, although provision is made for the use of ISAM. Coverage of secondary key processing is also included. The elements of relative files are covered in Chapter 15. Students are taught the three different types of relative files, including hash files, that can be used. Coverage of the concept of hash files extends as far as chained progressive overflow for handling collisions. Full coverage of hash files with chained progressive overflow will be included in a forthcoming advanced COBOL book.

Relational data bases and SQL are covered in Chapters 16 and 17. Chapter 16 covers essential relational data base and SQL concepts, and Chapter 17 shows how COBOL and SQL together can be used to process a relation (or "file") in a relational data base.

Chapter 18 covers on-line processing using COBOL and CICS. Many instructors include CICS in a COBOL course because of its importance in business programming. The chapter material is sufficient for an explanation of the basic concepts involved and for some introductory COBOL/CICS programming. However, if you intend doing extensive CICS programming, you will need a supplementary CICS text.

Chapter 19 covers some remaining peripheral topics, including CALL and COPY statements, the USAGE clause, and STRING and UNSTRING statements. Chapter 20 covers the use of the COBOL Report Writer feature. The chapter includes basic Report Writer concepts and instructions for generating reports using programs that utilize this feature.

Supplementary material

An effective course is dependent on a team effort involving the instructor and students, as well as the teaching and learning materials they both share. The complexities of teaching COBOL to a varied student audience gives rise to the need for a truly useful instructional support package to assist the process. For this reason, I have spent considerable time talking to COBOL instructors about their course needs to determine what combination of

teaching materials would be most useful. With these conversations as a backdrop, I have developed a comprehensive instructor's manual to accompany *Comprehensive Structured COBOL* that satisfies a broad cross-section of course needs.

Included for each chapter are

- a chapter guide
- topics for classroom discussion
- solutions to chapter questions
- solutions to chapter programming assignments
- complete examination materials containing true/false, multiple-choice, fill-in, and debugging questions, and questions requiring students to write COBOL program excerpts

Also in the instructor's manual is a complete set of overhead transparency masters with art from the text and both full programs and program excerpts.

To bring COBOL into the lab, there is an accompanying data disk in ASCII for convenient up- or downloading of files in either mainframe or micro environments. Utilizing this data disk enables students to learn while doing, without the drudgery of having to rekey the assignments in the text.

Acknowledgments

The author is grateful for the hard work of the following people who reviewed several versions of the manuscript and contributed many constructive comments: Eileen Bobman, Spring Garden College, Philadelphia; Lawrence E. Jeralds, Southern Illinois University, Carbondale; Dr. Mo Adam Mahmood, University of Texas, El Paso; Jeretta Horn Nord, Oklahoma State University, Stillwater; Karen Schnepf, Metro Community College, Omaha; Dr. S. Srinivasan, University of Louisville, Louisville; and Michael M. Werner, Wentworth Institute, Boston.

Dennis L. Varin, of Southern Oregon State University, Ashland, must be thanked for independently running and checking all the programs used in the text.

I would also like to warmly acknowledge my editors at Cole and Associates, Annette Gooch and Brete Harrison, and at McGraw-Hill, Karen Jackson and Raleigh Wilson, for their patience and commitment; the book designers, Seventeenth Street Studios, for a design that contributes significantly to the book as a learning tool; and my production editor at Cole, Lorna Cunkle, for her untiring efforts in bringing it to fruition. Developing a good book is a team effort; to those at Cole and McGraw-Hill who brought a successful team together, thank you.

Another essential aspect of this team effort is the continuing dialogue with adopters of a book, whose comments can be valuable in developing subsequent editions. For this reason, I'm always open to ideas and suggestions for improving the book. Please write to the author (Computer Science Department, University of Calgary, Calgary, Alberta, Canada T2N 1N4). Your contributions, if incorporated, will be acknowledged.

James Bradley
January, 1990

The following acknowledgment has been reproduced from COBOL Edition, U.S. Department of Defense, at the request of the Conference on Data Systems Languages.

Any organization interested in reproducing the COBOL report and specifications in whole or in part, using ideas taken from this report as the basis for an instruction manual or for any other purpose is free to do so. However, all such organizations are requested to reproduce this section as part of the introduction to the document. Those using a short passage, as in a book review, are requested to mention "COBOL" in acknowledgment of the source, but need not quote this entire section.

COBOL is an industry language and is not the property of any company or group of companies, or of any organization or group of organizations.

No warranty, expressed or implied, is made by any contributor or by the COBOL Committee as to the accuracy and functioning of the programming system and language. Moreover, no responsibility is assumed by any contributor or by the committee, in connection therewith.

Procedures have been establised for the maintenance of COBOL. Inquiries concerning the procedures for proposing changes should be directed to the Executive Committee of the Conference on Data Systems Languages.

The authors and copyright holders of the copyrighted material used herein—FLOW-MATIC (Trademark of Sperry Rand Corporation), Programming for the Univac (R) I and II, Data Automation Systems copyrighted 1958, 1959, by Sperry Rand Corporation; IBM Commercial Translator Form No. F28-8013, copyrighted 1959 by IBM; FACT, DSI 27A5260-2760, copyrighted 1960 by Minneapolis-Honeywell—have specifically authorized the use of this material in whole or in part, in the COBOL specifications. Such authorization extends to the reproduction and use of COBOL specifications in programming manuals or similar publications.

CONTENTS

1 COBOL FOUNDATIONS

3 The COBOL data division, 68

COBOL PROCESSING METHODS