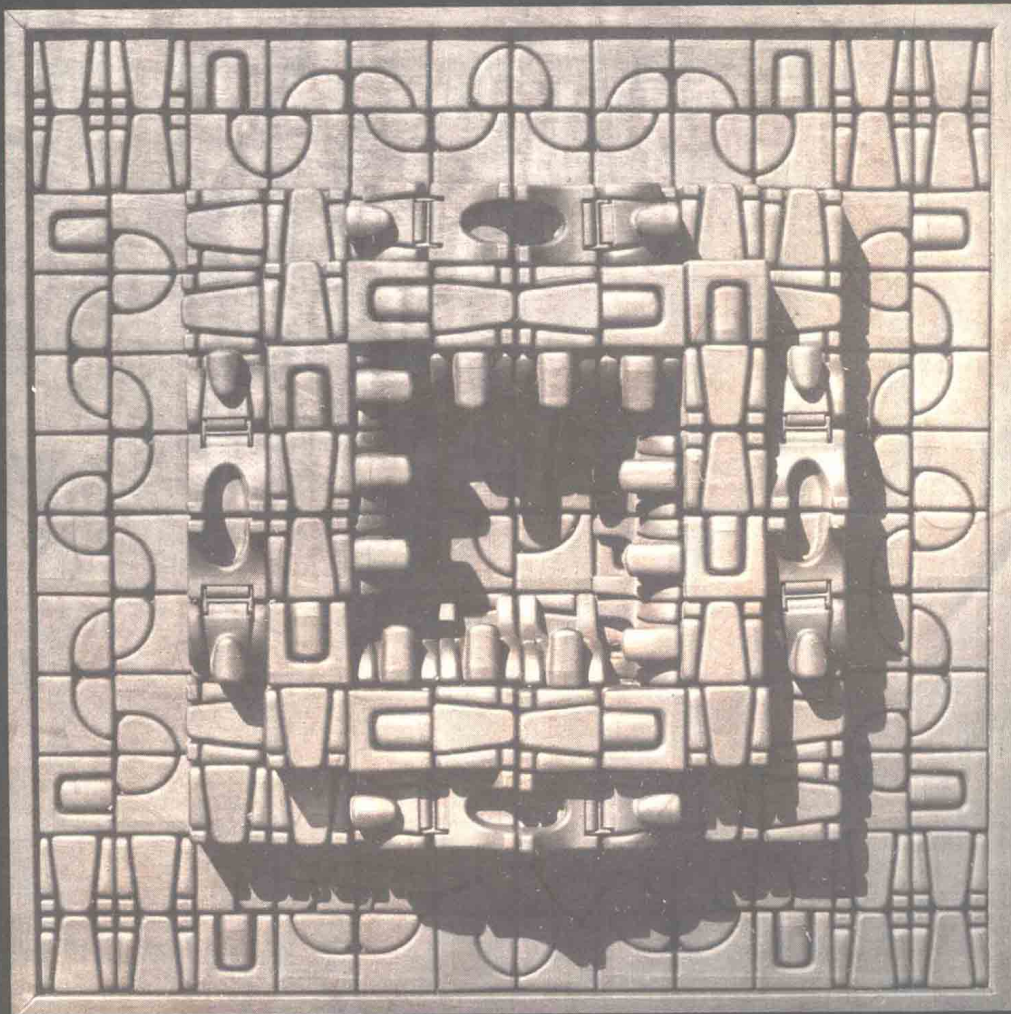FOURTH EDITION

STRUCTURED COBOL

FUNDAMENTALS AND STYLE

Tyler Welburn & Wilson Price

STRUCTURED COBOL

FUNDAMENTALS AND STYLE

TYLER WELBURN
WILSON PRICE

McGraw-Hill
San Francisco, California

Structured COBOL, 4th Edition
Fundamentals and Style

DIRECTORY TO EXAMPLE PROGRAMS

Example Programs and Required Data Files

Note: The extension C74 means the program is coded using COBOL-74 standards. All others (except SORTPP74.CBL) use COBOL-85.

Program	Input File	Comments
Chapter 1		
PATLIST.CBL	PATRON.DAT	Simple file listing
PATLIST.C74	PATRON.DAT	Simple file listing
Chapter 2		
PATDFCT.CBL	PATRON.DAT	Simple file listing with condition
Chapter 3		
No new programs		
Chapter 4		
PATCRE8.CBL		Creates PATRON.DT
PATADD.CBL	PATRON.DT	Add records to PATRON.DT
PATDISP.CBL	PATRON.DAT	Display records on screen
Chapter 5		
PATDFCT2.CBL	PATRON.DAT	Basic report headings
PATADD2.CBL	PATRON.DAT	Screen positioning for DISPLAY
Chapter 6		
PATDFCT3.CBL	PATRON.DAT	Percentage calculation
LOAN-PAY.CBL		Loan payment calculation
Chapter 7		
PATLIST2.CBL	PATRON.DAT	Basic page control & report totals
PATLIST2.C74	PATRON.DAT	Basic page control & report totals
PATLIST3.CBL	PATRON.DAT	Page totals
Chapter 8		
No complete programs		

Program	Input File	Comments
Chapter 9		
SCTLBRK.CBL	SALES_S.DAT	Single-level control break
SCTLBRK.C74	SALES_S.DAT	Single-level control break
MCTLBRK.CBL	SALES_M.DAT	Multiple-level control break
MCTLBRK.C74	SALES_M.DAT	Multiple-level control break
Chapter 10		
DATAVAL.CBL	SALES_TR.ERR	Data validation
DATAENT.CBL	SALES_T.DAT	Data entry/validation
Chapter 11		
AVERAGE.CBL	SALES_S2.DAT	Using an array
HIGHREP1.CBL	SALES_S2.DAT	Loading/processing data in an array
HIGHREP2.CBL	SALES_S2.DAT	Processing with parallel arrays
HIGHREP3.CBL	SALES_S3.DAT	Using an index
BEVERAGE.CBL	none	Searching a table
Chapter 12		
PAYTBLE1.CBL	none	Two-level table—direct lookup
PAYTBLE2.CBL	none	Two-level table lookup—positional/indexed
PAYTBLE3.CBL	none	Two-level table lookup—indexed
PAYTBLE4.CBL	PAY-RATE.TBL	Three-level table processing—input loaded
PAYTBLE5.CBL	none	Printing a three-level table
PAYTBLE6.CBL	none	Three-level table processing
PAYTBLE7.CBL	none	Printing a three-level table
ROOMS.CBL	ROOMS.DAT	Table processing—room chart
Chapter 13		
SORTONLY.CBL	INVEN-U.DAT	Basic sort
SORTPP85.CBL	INVEN-U.DAT	Sort with pre- and post-processing
SORTPP74.CBL	INVEN-U.DAT	Sort with pre- and post-processing—COBOL-74
SORTALPH.CBL	PROS-A.DAT, PROS-B.DAT	Case-insensitive sort
Chapter 14		
SEQ-MNT.CBL	BUDGET1.DAT, BUD-TRA.DAT	Sequential file maintenance
SEQ-MNT.C74	BUDGET1.DAT, BUD-TRA.DAT	Sequential file maintenance
Chapter 15		
EMPCRE8I.CBL	none	Create EMPLOYEE.DAI
EMPCRE8R.CBL	none	Create EMPLOYEE.DAR
EMPADDI.CBL	EMPLOYEE.DAT	Add records to EMPLOYEE.DAI
EMPADDR.CBL	EMPLOYEE.DAT	Add records to EMPLOYEE.DAR
EMPLIST.CBL	EMPLOYEE.DAI	Sequential list of an indexed file
EMPCRE8A.CBL	none	Create EMPLOYEE.DAA
EMPADDA.CBL	EMPLOYEE.DAT	Add records to EMPLOYEE.DAA
EMPLISTA.CBL	EMPLOYEE.DAA	List an indexed file using alternate key
EMPMNT.CBL	EMPLOYEE.DAI	Indexed file maintenance
EMPMNT.C74	EMPLOYEE.DAI	Indexed file maintenance

Program	Input File	Comments
SLSCRE8.CBL	none	Create SALES_MA.DAI
SLSADD.CBL	SALES_MA.DAT	Add records to SALES_MA.DAI
SLSUPD.CBL	SALES_S.DAT	Master-transaction update (indexed files)
SLSUPD.C74	SALES_S.DAT	Master-transaction update (indexed files)
EMPGETA.CBL	EMPLOYEE.DAA	Access indexed records using alternate key
EMPGETA.C74	EMPLOYEE.DAA	Access indexed records using alternate key

Chapter 16

PAYTBLE8.CBL	PAY-RATE.TBL	Using the COPY statement
PAYTBLE8.DTA		Source code for the COPY statement
PAYTBLE8.PRO		Source code for the COPY statement
PAY-T-SE.CBL	PAY-RATE.TBL	Subprogram to search a table
PAY-T-LD.CBL	PAY-RATE.TBL	Subprogram to load a table
PAY-T-GL.CBL	PAY-RATE.TBL	Program with nested programs

For Chapter 16 assignment

INV-CRE8.CBL LINEITEM.DAI	none	Create INVOICE.DAI and
ORD-CRE8.CBL	CUSTOMER.DAT, SOFTWARE.DAT	Generate customer and software files(16-1)

Appendix A

RPATLIST.CBL	PATRON.DAT	Report with headings and report total using Report Writer.
RSLCTBRK.CBL	SALES_S.DAT	Single-level control break using Report Writer.
RMLCTBRK.CBL	SALES_M.DAT	Multiple-level control break using Report Writer.

Appendix C

PRNCTL-I.CBL	PATRON.DAT	Switch to compressed font—IBM/Epson
PRNCTL-L.CBL	PATRON.DAT	Switch to compressed font—HP Laserjet

Alphabetic List of Example Programs

Program	Chapter	Description
AVERAGE.CBL	11	Using an array
BEVERAGE.CBL	11	Searching a table
DATAENT.CBL	10	Data entry/validation
DATAVAL.CBL	10	Data validation
EMPADDA.CBL	15	Add records to employee file for EMPMNT
EMPADDI.CBL	15	Add records to an indexed file
EMPADDR.CBL	15	Add records to a relative file

Program	Chapter	Description
EMPCRE8A.CBL	15	Create indexed employee file for EMPMNT
EMPCRE8I.CBL	15	Create an indexed file
EMPCRE8R.CBL	15	Create a relative file
EMPGETA.CBL	15	Access indexed records using alternate key
EMPGETA.C74	15	Access indexed records using alternate key—COBOL-74
EMPLIST.CBL	15	Sequential list of an indexed file
EMPLISTA.CBL	15	Listing an indexed file using alternate key
EMPMNT.C74	15	Indexed file maintenance—COBOL-74
EMPMNT.CBL	15	Indexed file maintenance
HIGHREP1.CBL	11	Loading and processing data in an array
HIGHREP2.CBL	11	Processing with parallel arrays
HIGHREP3.CBL	11	Using an index
LOAN-PAY.CBL	6	Loan payment calculation
MCTLBRK.C74	9	Multiple-level control break—COBOL-74
MCTLBRK.CBL	9	Multiple-level control break
PATADD.CBL	4	Add records to a file
PATADD2.CBL	5	Screen positioning for DISPLAY
PATCRE8.CBL	4	Create an empty file
PATDFCT.CBL	2	Simple file listing with condition
PATDFCT2.CBL	5	Basic report headings
PATDFCT3.CBL	6	Percentage calculation
PATDISP.CBL	4	Display records on screen
PATLIST.C74	1	Simple file listing—COBOL-74
PATLIST.CBL	1	Simple file listing
PATLIST2.C74	7	Basic page control & report totals—COBOL-74
PATLIST2.CBL	7	Basic page control & report totals
PATLIST3.CBL	7	Page totals
PAY-T-GL.CBL	16	Program with nested programs
PAY-T-LD.CBL	16	Subprogram to load a table
PAY-T-SE.CBL	16	Subprogram to search a table
PAYTBLE1.CBL	12	Two-level table—direct lookup
PAYTBLE2.CBL	12	Two-level table lookup—positional/indexed
PAYTBLE3.CBL	12	Two-level table lookup—indexed
PAYTBLE4.CBL	12	Three-level table processing
PAYTBLE5.CBL	12	Printing a three-level table
PAYTBLE6.CBL	12	Three-level table processing
PAYTBLE7.CBL	12	Printing a three-level table
PAYTBLE8.CBL	16	Using the COPY statement
PAYTBLE8.DTA	16	Source code for the COPY statement
PAYTBLE8.PRO	16	Source code for the COPY statement
PRNCTL-I.CBL	App. C	Switch to compressed font—IBM/Epson
PRNCTL-L.CBL	App. C	Switch to compressed font—HP Laserjet

Program	Chapter	Description
RMCTLBRK.CBL	App. A	Multiple-level control break—Report Writer
ROOMS.CBL	9	Table processing—room chart
RPATLIST.CBL	App. A	Report headings and totals—Report Writer
RSCTLBRK.CBL	App. A	Single-level control break—Report Writer
SCTLBRK.C74	9	Single-level control break—COBOL-74
SCTLBRK.CBL	9	Single-level control break
SEQ-MNT.C74	14	Sequential file maintenance—COBOL-74
SEQ-MNT.CBL	14	Sequential file maintenance
SLSADD.CBL	15	Add records to indexed files for SLSUPD
SLSCRE8.CBL	15	Create indexed files for SLSUPD
SLSUPD.CBL	15	Master-transaction update (indexed files)
SLSUPD.C74	15	Master-transaction update (indexed files)—COBOL-74
SORTALPH.CBL	13	Case-insensitive sort
SORTONLY.CBL	13	Basic sort
SORTPP74.CBL	13	Sort with pre- and post-processing—COBOL-74
SORTPP85.CBL	13	Sort with pre- and post-processing

PREFACE

The prefaces to previous editions of this book included the following: "This text utilizes the most current COBOL program design and coding techniques together with contemporary business computer system concepts. It offers a solid foundation for beginning COBOL students to build upon as their knowledge and skills develop and grow along with the exciting field of computer technology."

Since the third edition, two significant changes in the COBOL field have occurred. First, the focus has finally shifted from COBOL-74 to COBOL-85 in our instructional programs (although COBOL-74 and COBOL 9-X are mentioned where appropriate). Second, powerful microcomputer hardware and COBOL software are readily available. These two changes were primary considerations for adjustments to the fourth edition in order to maintain the quality of the first three editions.

Goals for the Fourth Edition

First, this fourth edition has benefited immensely from the observations and criticisms of many users of the third edition. Their input provided us with a solid foundation on which to incorporate needed improvements and updating. In addition to maintaining the quality of the third edition, our broad goals were as follows:

- Incorporate file handling and other advanced topics (covered in the companion volume, *Advanced Structured COBOL: Batch, On-Line, and Data-Base Concepts*) to make this a comprehensive, full year textbook.

- Shift the emphasis from COBOL-74 to COBOL-85, while including suitable references to both COBOL-74 and COBOL-9X.

- Introduce interactive programs early and expand upon interaction in subsequent chapters as appropriate and meaningful.

- Further diversify and expand the complete program examples for illustrating new programming principles.

- Intensify the focus on solid structured methodology, emphasizing programming practices that produce intuitively apparent and easily maintained programs.

Each of these is described in more detail later in this preface.

Key Strengths Retained

In preparing the fourth edition, we made a special effort to maintain—and *improve*—the qualities and features that instructors and students found useful in the third edition.

Text Organization

The COBOL syntax is presented within the framework of commonly encountered business-system program models. Concepts are developed step by step—proceeding from the simple to the more complex. Each program category builds upon and adds to the knowledge, techniques, and skills developed in the previous one.

The programs presented here also serve as a ready reference for each application type, and they introduce students to the fact that several traditional application-program categories exist. This permits students to analyze programming tasks in relation to program type and to use common approaches rather than to "reinvent the wheel" for each program. Further, with this method of presentation, coding is never divorced from practical application.

Concepts Modules

The third edition included nine *concepts modules* describing general principles related to the programming chapters that followed. Student response to separating general theory from COBOL principles was very positive. Those already familiar with the general theory simply skipped the corresponding module; those requiring the background appreciated the delineation between theory and COBOL. The major criticism was their number. To that end, we took two tacks. First, we cut down the material; second, we incorporated appropriate elements directly into corresponding COBOL chapters. Consequently, this edition includes only five concepts modules.

Programming Style Conventions

Over the years, through trial and error, COBOL installations and programmers have developed programming style conventions. The establishment and use of such conventions significantly aid program readability and maintainability. In this text, therefore, applicable style considerations are presented—along with the syntactical coding rules—to provide an introduction to proper coding form. Through presentation of these style conventions, students can quickly learn sound coding practices that might otherwise take years to acquire. Basic programming conventions are introduced with the first program and are expanded upon as appropriate in later examples.

Comprehensive Program Design Documentation

New concepts are introduced through program examples. Most new sample programming applications include programming specifications, a structure chart, a pseudocode solution, and the COBOL program. Programming specifications include complete input/output descriptions and a detailed list of program operations. Structure charts are provided to show overall program organization and module definition. English-like pseudocode is used to make comprehension of the detailed logic easier for the beginning programming student. The resulting COBOL solution, complete with in-program comments, is evolved logically from the preceding program design documentation.

Combination Tutorial/Reference Approach

Students often react to COBOL textbooks in one of two ways. Typically, either they feel that the text is a good reference manual but doesn't really explain how to write certain types of programs, or they claim that the book explains concepts well but is difficult to use as a reference. This text blends tutorial and reference features. Whenever a subject is presented, it is fully covered in one place. However, to guard against information overload, topics are covered on a step-by-step basis and are integrated with programming examples. Furthermore, subjects are organized to give a quick description, and then go into detail. This enables students to skip much of the detail if desired, progress to the next topic (or chapter), and then return at a later date for the fine points.

Extensive End-of-Chapter Material

At the end of each chapter, a chapter summary is presented. Appropriate chapters also contain COBOL language element summaries and style summaries.

Exercises appear after the summaries. Each chapter includes terms for definition and review questions. Most chapters also contain syntax/debug exercises.

Range of Programming Assignments

Each chapter includes several programming assignments that relate directly to the material covered within the chapter. Most of the assignments are independent of other assignments. Some are cumulative; for instance, Programming Assignment 5-2 requires that the program of 3-2 be expanded.

A complete set of test data is included on the data disk accompanying the book. The program specifications for each programming assignment identify the appropriate data set.

Numerous Illustrations and Examples

Over 350 figures are in the text. The figures include illustrations, diagrams, programming specifications, design documentation and coding examples.

A Complete Teaching Package

This fourth edition is not merely a textbook; it is a *teaching package,* designed to provide you—the instructor—with the broadest possible support. The package consists of the following components:

- A comprehensive **Instructor's Manual,** containing chapter objectives, teaching tips, answers to exercises, test questions, transparency masters, and data files for programming assignments.

- A **data disk** containing a complete set of sample programs from the textbook and data files for each programming assignment.

- A **solutions disk** containing solutions for each programming assignment.

Chapter/Topic Organization

Each chapter contains a preliminary overview discussion. Chapters that cover two or more distinct topics identify these topics as individual entities. Such topic organization clearly indicates the subject of each text segment. This permits easy identification of the material that the instructor might choose to skip or cover in a different sequence.

COBOL-85 Software Available

Because all programs conform to COBOL-85, they can be compiled and run on a mainframe, mini, or personal computer. Students desiring their own personal computer software have two options: Micro Focus or Ryan-McFarland.

Micro Focus COBOL-85

The most comprehensive approach is the purchase of Micro Focus Personal COBOL. It is important to avoid considering this as "microcomputer COBOL," which implies that it is something less than full COBOL-85, or a limited-use educational version. *This is COBOL-85;* it even includes the 1989 Intrinsic Functions package. Although not documented in the *Micro Focus Personal COBOL 2.0 Programmer's Guide*, Personal COBOL also includes the COBOL Report Writer feature. The Report Writer examples programs in Appendix A were tested with Personal COBOL.

The cost of the system, available through Micro Focus to users of this book, was less than $50 at the time of this book's publication.

**Ryan-McFarland
COBOL-85**

For the more economically inclined, this book can be ordered complete with the special education version of the Ryan-McFarland COBOL-85 compiler. It is important to recognize that this version is *not* a watered-down system, useless for all except the simplest problems. In fact, the opposite is true; the compiler includes the following powerful features:

1. COBOL programs of up to 1,000 lines of source code can be compiled.
2. Supported file types include sequential, line sequential, relative, and indexed.
3. File size limitation for indexed files is 100 records; for all others, it is 1,000 records.
4. Records in the File Section can be up to 132 bytes in length.
5. Up to four file definitions may be used in any program (that is, four files may be open at one time).
6. Indexed files may include one or two keys.
7. The CALL statement can be implemented to one level. That is, a program can call another, but the called program cannot call a third program.

This compiler is adequate for all but this book's most advanced applications. For major projects, the restriction on the program size and the number of simultaneous files open can create problems.

**Separate "Getting
Started" Manuals**

For users of Micro Focus or Ryan-McFarland, separate "getting started" student manuals are available through McGraw-Hill.

Start-Up! with Micro Focus Personal COBOL 2.0 provides step-by-step instructions for using the Editor, Checker, and Animator. The wide variety of Micro Focus features are illustrated through a series of example dialogues with the corresponding software. This volume also includes a description of the Micro Focus implementations of: (1) extensions of the ACCEPT and DISPLAY statements, (2) a SCREEN SECTION for interactive input/output, and (3) a screen painter facility to produce the detail code necessary for keyboard/screen activities.

Start-up! with RM 5.2 provides step-by-step instructions for compiling and running programs. A complete chapter is devoted to using RM/COSTAR for managing programs, entering and editing, syntax checking and compiling, and using the debugger. This volume also includes a description of the Ryan-McFarland implementations of: (1) extensions of the ACCEPT and DISPLAY statements, and (2) a SCREEN SECTION for interactive input/output.

**Expanded and
Updated Material**

When preparing this revision, we considered every topic and figure of the third edition in order to present the material as clearly and understandably as possible. Similarly, each topic was reassessed for relevance, and additional topic area candidates were considered. Areas in which significant new material appears are as follows.

**Expansion to Include
File Processing**

The third edition was written to be used in conjunction with *Advanced Structured COBOL: Batch, On-Line, and Data-Base Concepts* to provide a full year's coverage of COBOL. This edition was trimmed where appropriate, then expanded to include advanced file-processing concepts and program management. The following topics are included: sequential file maintenance and updating, random file processing (with a strong focus on interactive file updating), and subprograms (separately called programs and nested programs).

Interactive Programs

Next to expanding the third edition to cover file processing, the most commonly encountered request was to include interactive processing capabilities. When doing this, we gave special consideration to avoid token interaction and incorporate it in a useful, meaningful context. Chapter 4, a short chapter, shows the reader how to (1) create a new field, (2) add records to an existing field, and (3) display records, one at a time, from a file. COBOL-85 versions of the ACCEPT and DISPLAY are used in all examples. These basic concepts are expanded in subsequent chapters with assignments as appropriate.

Chapter 5 deals with improving the appearance of computer output, both printed and displayed on a screen. At this point, nonstandard forms of the ACCEPT and DISPLAY are included, allowing for line/column screen addressing, clearing the screen, and displaying in reverse video. Forms for several compilers are described.

For instructors who don't want to include interactive programs, these sections can be omitted with no loss of continuity.

Tools Chapters

In each chapter, we have sought to maintain a balance between COBOL language features and COBOL programming principles—new features are generously illustrated with complete program applications. However, there are two language-feature exceptions: editing (the first topic of Chapter 5), and conditional operations (Chapter 8). These provide extensive coverage of the corresponding language features. To a degree, they can be considered almost as reference materials. The student does not need to exhibit mastery of all these features after completing each chapters. But he or she should have a basic insight to them and know where to look when they are used in example programs or required for programming assignments. In this sense, these can be considered "tools" chapters.

Complete Programs

Students learn best by working from solid examples of new concepts—we learned this from previous editions. To that end, we illustrate each new principle with one or more examples. In most cases, these examples are not merely abbreviated program segments, but complete programs that both demonstrate the new principles and show them in a typical context. The number of complete programs included in this edition has almost doubled (to approximately 50) over the third edition.

The complete set of example programs is included on the data disk accompanying this book. For each program, the filename is identical to the PROGRAM-ID designated in the program. The name extension is CBL for COBOL-85 programs and C74 for the limited number of COBOL-74 programs. For example, PATLIST is stored as PATLIST.CBL and PATLIST.C74. Test data for the sample programs is also included on the disk; those filenames are as designated in the programming specifications for the program.

Controlling Printer Pitch for Personal Computers

Limited print-line width on personal computers was an expressed concern by several institutions—the recommendation was that all report-generating assignments be limited to a report line maximum of 80 characters. Instead, we included, as Appendix C, COBOL code that can be inserted in any program to change from 10 characters per inch to compressed printing (approximately 17 characters per inch) and back again. Example code is included for the IBM Graphics/Epson printers and the Hewlett-Packard LaserJet. The technique employed is easily converted to any other printer type.

COBOL-85

In the third edition, equal treatment was given to COBOL-74 and COBOL-85. Most current books that place their primary emphasis on COBOL-85 (but include COBOL-74) are written with the constraints of COBOL-74. It is approximately one decade since the 1985 Standard was introduced. We believe the time has arrived to teach COBOL by taking advantage of COBOL-85's many powerful features—without living within the COBOL-74 limitations. To achieve this, the fourth edition maximizes the capabilities of COBOL-85 to produce good, easy-to-follow COBOL programs. However, COBOL-74 is not ignored. Equivalent COBOL-74 forms are described; in a few cases, a complete COBOL-74 program is shown when the differences are substantial.

Significant additions and changes to COBOL scheduled with the next standard (COBOL-9X) are noted where appropriate.

Acknowledgments

Anyone who has written a book knows of the many efforts by people behind the scenes. This book is no exception. At the top of the list, Frank Ruggirello of McGraw-Hill provided the resources to make everything work—and work well. Deborah Colon, also at McGraw-Hill, provided hands-on coordination to ensure that the many facets of the project moved smoothly at all levels.

Overall project coordination from Jane Granoff was the norm for her—outstanding, as usual. Karen Richardson did far more than her normal copyediting task. Not only did she improve significantly on our English, but she did a terrific job of proofreading. We did become weary of her harassing comments and notes, but it was difficult to be critical when she was usually right and added to the book's quality. Rad Proctor, of Proctor-Willenbacher, and her staff did an excellent job of turning a complex manuscript into a well-crafted book.

Chuck Nelson (Programming Assignment solutions) and Jorene Kirkland (Instructor's Manual) deserve special thanks for their efforts in preparing critical and ancillary materials. Numerous others at the nuts-and-bolts level put forth special efforts to ensure that this project was done properly and completed on time. To them and to the following people who provide special assistance in preparation of this book's four editions, we offer our sincere thanks and appreciation.

William Barth
Cayuga Community College

Robert H. Doursen
California Polytechnic State University

Paul Duchow
Pasadena Community College

Janie Epstein
Johnson County Community College

Mary Ann Grams
San Antonio College

Carol C. Grimm
Palm Beach Junior College

R. Wayne Hedrick
Texas A&M University

Susan Hinrichs
Missouri Western State College

Jorene Kirkland
Amarillo College

Dr. Ron Kizior
Loyola University, Chicago

Sue Krimm
Los Angeles Pierce College

David M. Kroenke
Wall Data, Inc.

Marjorie Leeson
Delta College

Anthony Mann
Central State University

Anne M. McBride
California State University, Chico

Joe Murray
North Adams State College

Charles Nelson
Rock Valley College

Jack Olson
Merritt College

Edward Rategan
College of San Mateo

Cyndi Reese
Santa Rosa Junior College

Dan Rota
Robert Morris College

Joseph Southern
LaGuardia Community College

George Vlahakis
Evergreen Valley College

BRIEF CONTENTS

CONTENTS

CONCEPTS MODULE A   **THE PROGRAM DEVELOPMENT PROCESS**   1

CHAPTER 1   **A First Look at COBOL**   12