GROLIER BUSINESS LIBRARY

INTRODUCTION TO BUSINESS DATA PROCESSING

GROLIER BUSINESS LIBRARY

GROLIER BUSINESS LIBRARY

LAWRENCE S. ORILIA
Nassau Community College

Introduction to Business Data Processing

Grolier Incorporated

Reprinted by arrangement with McGraw-Hill, Inc.

INTRODUCTION TO BUSINESS DATA PROCESSING
Second Edition

ISBN: 0-7172-8511-1

To my son Adrian,
my daughter Vanessa,
and my wife Tracy,
who have all helped to fulfill my life
and represent the measure of my happiness.

Preface

The successes associated with the first edition of this introductory data processing text reinforced many of the personal beliefs expressed in its writing. These beliefs included the presentation of data processing principles with carefully integrated illustrations and detailed explanations. Sufficient illustrative examples, both pictorial and narrative, were provided to reinforce concepts and their application. The proper placement of illustrative problems offered students the step-by-step guidance they needed to master the concepts presented.

Because of educational philosophies which I developed when attending college, the first edition offered many practice problems and discussion questions with which to apply the principles learned. I believe quite strongly in the idea that students learn best by doing. Of considerable impact was the text's reading level and style. The conversational manner of presentation was deliberate. It encouraged student learning, avoided the more cryptic or tedious styles of other authors, and did not talk down to the students. The conversational mode avoided the presentation of extremely technical material or material that detracted from student learning.

Many adopters of the first edition commented favorably on these points and remarked on the reference quality of the text. Students could freely turn to the text and learn on their own, at their own pace. These instructors felt that because the text covered the fundamental concepts so well, they were free to discuss other topics of interest or provide more individualized instruction.

These ideas, successfully incorporated into the first edition, have become the foundation of the second edition. Naturally, these concepts were refined and fine-tuned to focus their effectiveness. Every attempt was made to maintain the currentness of the second edition, in spite of the ever-changing nature of this field.

While writing this edition, I continued to have one thought in mind: I wanted the beginning computer student to be capable of reading the text and fully understanding the material presented. This desire evolved from a problem I have frequently encountered in the classroom.

Too often, I have assigned readings to students only to have them state that the material was impossible to follow. This was especially true for discussions of flowcharting, structured programming, and introductions to programming, in which sufficient detail is required to master the subject matter. Many of the other texts currently available merely skim the surface of this material or offer cartoons instead of well-thought-out explanations, leaving the student with nothing concrete on which to base learning. I know that I have provided the detail necessary to permit the independent student development of this material.

This approach has a positive benefit to the instructor as well. Freed of the necessity to cover virtually all aspects of a topic, the instructor can introduce new material for class discussion. This new material might enhance a discussion, motivate increased student participation, provide special projects, or introduce topics which are of particular importance to the individual instructor.

Organization

The overall organization of this text enables the reader to develop a fundamental knowledge of the computer prior to the discussions of programming and systems analysis and design. Each chapter is written as an independent unit, providing complete coverage of a topic within its content. Thus, if an instructor desires to cover a chapter out of sequence, the continuity of the presentation will not be adversely affected.

Classroom testing has proved the chapter organization used in this text to be effective. Chapters 1 to 5 present material consistent with most introductory data processing courses. The material covered provides students with principles fundamental to data processing, enabling them to begin programming. Chapter 6, "Flowcharting," provides a strong foundation for the programming chapters that follow. Chapter 7 introduces programming and programming languages. The computer languages BASIC and COBOL are discussed in Chapters 8 and 9, respectively. Program solutions developed in these chapters are closely tied to the flowcharting problems discussed in Chapter 6. Instructors are also free to develop solutions they have specifically employed in the past.

Two entirely new chapters were written on structured programming and management information systems (MIS). Chapter 10 discusses structured concepts and applies them to BASIC, COBOL, and PASCAL in detailed illustrative examples. Chapter 13's discussion of MIS reviews the organization of these complex systems, their relationship to databases, and the use of IMS systems. Specialized systems such as the CICS and DC/DB configurations are presented, with detailed case studies offering a practical orientation to both MIS and database systems.

The latter part of the text offers special discussions of minicomputers, microcomputers, and other types of computer systems; data communication systems, systems analysis, and design concepts; and a detailed example of a systems documentation package. The review of many of these topics can add much to the content of an introductory computer course and provide the student with a broader overview of the business data processing field.

The organization of the chapters affords the instructor flexibility. The instructor can use the first seven chapters to develop the concepts of data processing for half of the semester. The remainder of the semester can be devoted to programming applications. Another approach might provide a brief discussion of programming and the development of systems-related concepts in the last half of the semester. The instructor is free to choose the topics of coverage and can diversify the material presented.

Learning Objectives

Every chapter begins with a section entitled "Purpose of This Chapter," which presents the student with an overview of the material and topics to be covered. This section provides a general feel for the chapter's content. The student can grasp the organization of the chapter and place topics of discussion in their proper perspective.

The purpose section also presents the learning objectives for the chapter. These briefly stated objectives offer the student a guide to the key areas of the chapter and the skills and concepts to be gained from reading it. The learning objectives are also of value when a review of the chapter is anticipated, prior to a test.

Key terms used throughout the chapter are also listed in the learning objectives. The terms are commonly used in data processing and represent an operational vocabulary vital to the current or future user of computer services. All the terms are defined in the text of the chapter and appear in a glossary at the chapter's end.

Readability

A concerted effort has been made to keep the reading level of this text from becoming overly technical, monotonous, or unduly complicated. Standardized reading tests applied to the text indicate that the average high school graduate should not have difficulty in comprehending the material presented. I have blended this reading level into a conversational mode of presentation. It is my belief that the conversational approach greatly assists the learning process and is uniquely suited to today's student. It does not belittle the student, but rather guides the reader through the required material on a step-by-step basis in an easily comprehensible manner.

Summary

A point-by-point summary of all material covered appears at the end of the chapter. The summary details the major topics discussed in the chapter, capsulizing each point in a few sentences. The summary is organized to follow the presentation of material in the chapter, reinforcing the order of topic coverage. Students will find this type of summary particularly advantageous when reviewing for a test.

Glossary

An introductory text requires clear definitions of all terms used in its discussion. The chapter glossaries list, in alphabetical order, all key terms introduced in the chapter.

End-of-Chapter Tests

The discussion questions and summary tests at the end of each chapter enable the reader to test his or her mastery of the material covered in the chapter. The student is advised to complete the summary test before proceeding on to the next topic of discussion. The summary test can also be used in preparation for an exam. The topics related to questions that have been answered incorrectly can be reviewed before the test. Summary test answers appear at the end of each chapter.

Special-Interest Items

Students like to study material that is current and related to real-life situations. In an effort to meet this requirement, items of special interest have been included in each chapter. These items are drawn from a variety of sources and relate directly to the materials covered in the chapter. In some cases, these special items note the widespread applicability of the computer and some of its more appealing uses. In the chapters related to programming, the special items highlight programming considerations affecting the student. These items point out commonly made student errors, ways to avoid specific mistakes, and tips to help simplify programming assignments. Each special item is intended to enhance the presentation of the material and complement the chapter's coverage of a topic.

Case studies at the end of each chapter also reinforce the practical and diversified uses of computers in our daily environment. Each case study has been rewritten to focus on its high points and avoid highly technical discussions. Case study applications range from the use of computers to prepare mailing labels for small advertising companies to the use of computers by vast administrative organizations supporting millions of people. Each case attempts to bring into focus how the computer is used and why it serves the purpose that it does.

Of particular interest are the new topics included in the second edition. New material includes characteristics and examples of computerized crime, service bureaus, and operating systems; new types of hardware and word processing; newly introduced programming languages; managerial considerations in systems evaluations; canned programs; business simulations and models; and details on the newer versions of both minicomputers and microcomputers.

Many instructors will be especially interested in Chapter 15, "Documentation of a System." Because systems-related subjects represent one of the more difficult topics for many students, Chapter 15 serves as an example of the work a systems analyst may perform. Many instructors assign this chapter for reading before their DP students enter the systems course. Many students do not know what a system is; this exposure offers them an initial glimpse of this topic. Many students retain this text and use Chapter 15 as a guideline in the preparation of their systems project.

In general, I have tried to write a text that is easy to read, is informative, and assists in the development of selected data processing skills. I have attempted to include material which is relevant to the study of computers, without becoming overly technical. I believe that this text provides students with a working knowledge of computer-related data processing skills that can be used in subsequent computer courses or in the performance of their jobs. I would like students to think of this text as a reference that they can turn to when faced with a data-processing-oriented question or task. I have tried to make this text, as well as the learning of data processing skills, an enjoyable experience.

Acknowledgments

Many exceedingly fine and talented individuals have contributed to the success of this text. I was very fortunate in having a group of reviewers whose critical evaluations were of great value during the preparation of the manuscript. I wish to thank Paul Fuhs and James Wynne of Virginia Commonwealth University; Beverly Madron of Western Kentucky University; Robert M. Stewart of Iowa State University; William L. Fox of the College of DuPage; James Baldwin of Nassau Community College; Jerry A. Elam of Saint Petersburg Junior College; Michael Nolan of the University of Nebraska, Lincoln; Victor Sherrer of Trenton State College; Edward Irving of North Carolina Central University; Jacque Vail of Burlington County College; Sheila M. Fay of Solano Community College; Edward Eckhard of North Shore Community College; Richard J. Westfall of Cabrillo College; Thomas W. Kelly of Northeastern University; B. J. Mangold of Montclair State College; Iva Helen Lee of McLennan Community College; and Barry A. Stein of Queensborough Community College.

No text would evolve without the assistance of editorial support. In this regard, I would like to thank the editorial contributions of Jim Vastyan, Fran Neal, and Jon Palace, all of the College Division. My express thanks go to Mel Haber for his assistance in developing the text's art program and his moral support during the initial production phases. I would also like to thank Joe Campanella for keeping both the first and second editions on their production schedules. A word of praise should go to Inge King for her fine photo research efforts.

In general, I would like to thank everyone on the McGraw-Hill College Division staff who has contributed to making this edition and the first edition a success. I am certain that we have all learned a great deal from this project.

My last acknowledgment is reserved for my wife, Tracy, and my children, Adrian and Vanessa. They put up with the frustrations and absence caused by this work. Without their collective encouragement and patience, the successful completion of this project would not have been possible. Tracy's specific support carried me through the dog days of writing, when emotional and verbal encouragement were needed.

Lawrence S. Orilia

Portfolio: The History of Data Processing

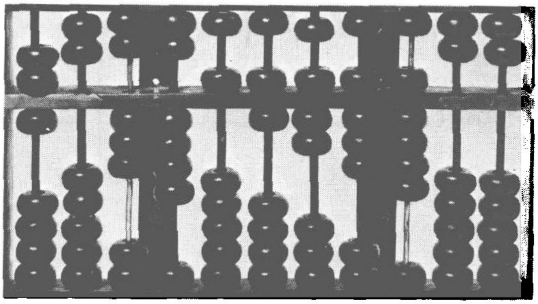

PLATE 1
The suan-pan is the Chinese version of the abacus. *(IBM.)*

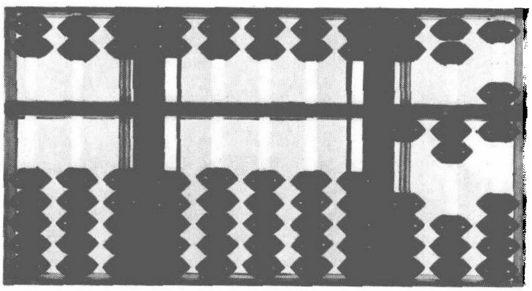

PLATE 2
A Japanese abacus, or soroban. *(The Granger Collection.)*

The abacus is one of the earliest known computational devices and can be traced to ancient Babylonia. The abacus remains useful today in certain small businesses and in elementary schools where students are learning arithmetic.

A system of numerical notation was essential to the processing of data. People needed a shorthand to represent quantities in computations. Many societies have developed methods of representing quantities, some based on the numbers 5, 8, 20, and 64. Most societies today use the Arabic numbering system, a

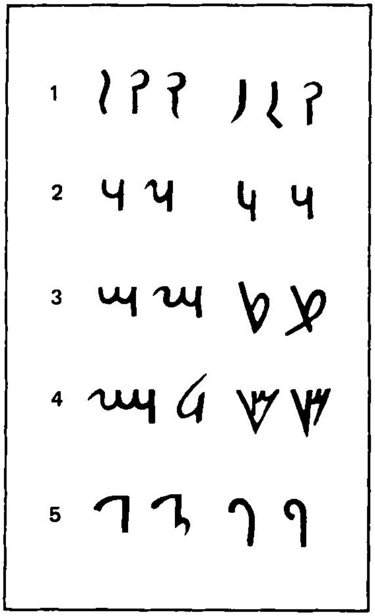

PLATE 3
Numerical symbols used in early Egypt.

PLATE 4
Bookkeeping in a medieval monastery. ***(The Granger Collection.)***

PLATE 5
Fifteenth-century English tally stick. *(The Bettmann Archive.)*

decimal system based on the number 10. Computers use a binary system, based on the number 2, to represent their data.

Fra Luca Pacciola, a fourteenth-century monk, developed and applied the concepts of double-entry bookkeeping, which laid the foundation of modern accounting principles. In recent years, these principles have been adapted to facilitate the computerized accounting of financial data. The computer can analyze large amounts of financial and accounting data and present the information in an immediately usable format. The computer format is a long way from the quills and ink once used to record transactions.

Each notch on the English tally stick represents 1 pound sterling. The tally stick served as a tax receipt and was a permanent record of tax payments. The accurate recording of data was a vital aspect of society, even in the Middle Ages.

In the 1640s, the Frenchman Blaise Pascal invented a mechanical device that functioned as an adding machine. Known as the Machine Arithmetique, the device was constructed of interlocking gears that represented the numbers 0 through 9. It operated like an odometer, which records an automobile's mileage. Pascal's was another historical attempt to develop a mechanical device that would perform arithmetic operations.

Gottfried von Leibniz, a German mathematician, further refined Pascal's concepts and produced a calculating device capable of multiplication, division, addition, and subtraction.

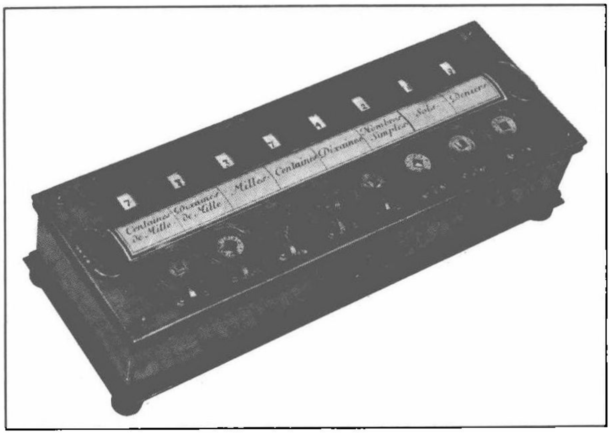

PLATE 6
Pascal's Machine Arithmetique. *(IBM.)*

PLATE 7
Gottfried von Leibniz. *(Culver Pictures.)*

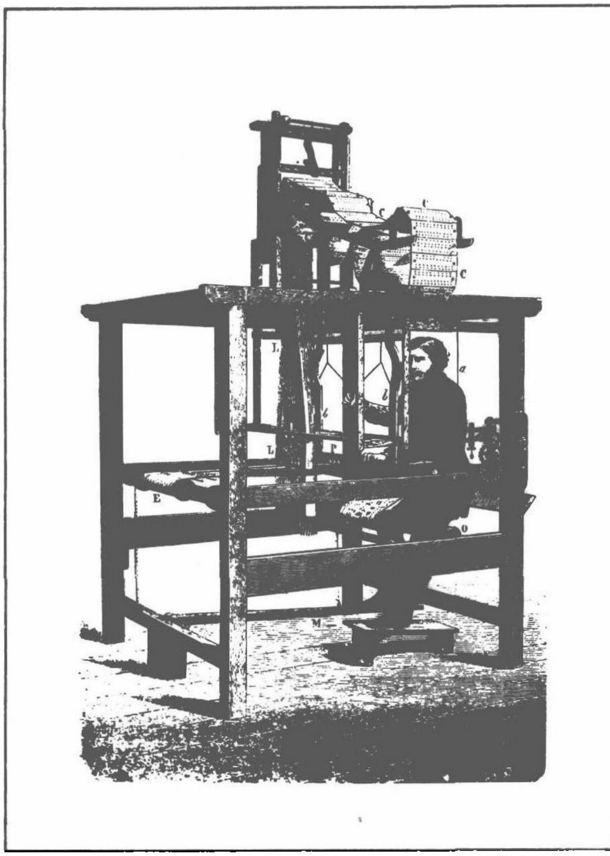

PLATE 8
Jacquard's automated loom. *(IBM.)*

In 1804, Joseph Marie Jacquard perfected the idea of the automated loom. Using holes punched into a series of connected cards, Jacquard was able to control the weaving of fabrics. The loom used in this process sensed the pattern coded into the cards and wove the fabric accordingly. These cards were the forerunners of Hollerith's punched cards.

In the early 1800s, the English inventor Charles Babbage theorized that it was possible to construct an automatic, mechanical calculator. With the support of the British government, Babbage began the construction of the Difference Engine and, years later, the Analytic Engine. The concepts Babbage put forward were eventually used by engineers in the development of the first computer prototypes.

Despite ten years' work, Babbage failed to build a fully operational model of either the Difference or Analytic Engine and lost his government subsidy. Not until 1854 did George Pehr Schuetz build a working model of the Difference Engine.

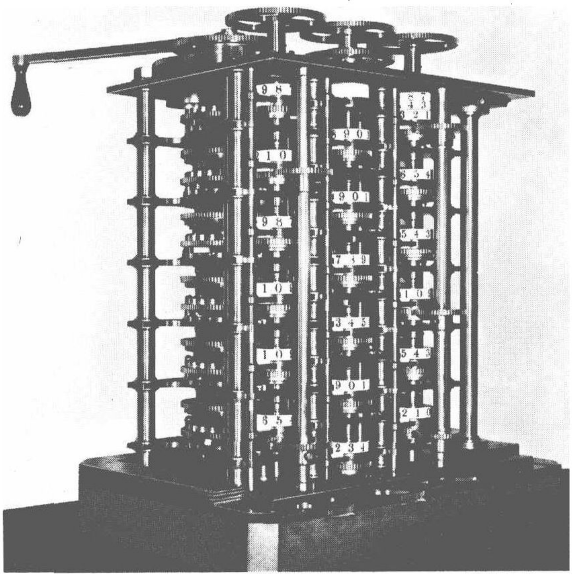

PLATE 9
The Difference Engine was constructed using the theories of Charles Babbage. *(IBM.)*

In 1842, a paper by L. F. Menabrea on the Analytic Engine was translated from Italian into English by Augusta Ada Byron, Countess of Lovelace, and presented to her colleagues. Babbage encouraged Lady Lovelace to conduct her own research and refine many of the concepts in the paper. Lady Lovelace's contributions to binary arithmetic would later be used by John Von Neumann in developing the modern computer.

In the 1880s, the U.S. Census Bureau asked Herman Hollerith to find a way to speed up the processing of census data. Hollerith created punch cards that resemble today's computer cards, their code, and tabulating equipment. The 1890 Census was completed in approximately 3 years rather than the 11 years the Census Bureau had originally estimated.

In 1937, a computer was developed at Harvard University by H. H. Aiken. This device, the Mark I, was a prototype of the computers used today. It is less well-known that a closely related predecessor to the Mark I was built at Iowa State College in the 1930s. This electronic machine was developed under John V. Atanasoff's supervision and laid the groundwork for the ENIAC (electronic numerical integrator and calculator) computer, which appeared in 1946. World War II sparked intense research and computer development, and the ENIAC was the first all-electronic computer.

During the same period, a brilliant mathematician, John Von Neumann, presented technical papers on the stored program concept. According to this concept, the operating instructions and data used in processing should be stored inside the computer. Whenever necessary, the computer would have the capability to modify these program instructions during their execution. The stored program concept was the basis for all future computer advances. In 1949, this concept was incorporated into the computer EDSAC

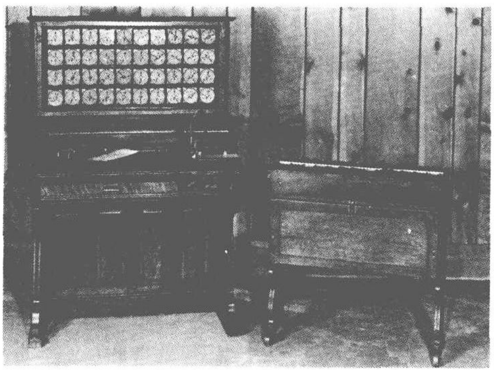

PLATE 10
The electrical tubulator and sorter developed by Hollerith. *(IBM.)*

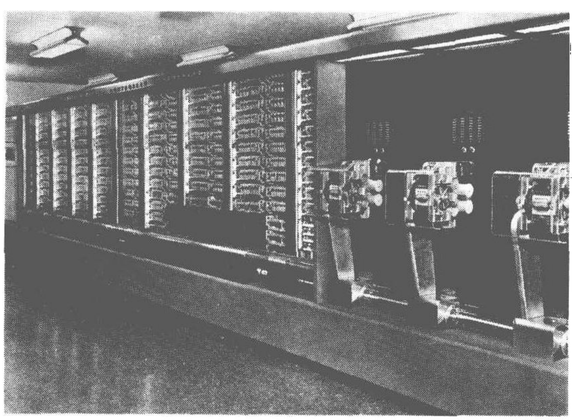

PLATE 11
The Mark I was developed by H. H. Aiken in 1937 at Harvard University. *(IBM.)*

PLATE 12
John Von Neumann. *(UPI.)*