JULIE M.T. CHAN
MARILYN KOROSTOFF

TEACHERS' GUIDE TO DESIGNING CLASSROOM SOFTWARE

TEACHERS' GUIDE TO DESIGNING CLASSROOM SOFTWARE

JULIE M.T. CHAN
MARILYN KOROSTOFF

SAGE PUBLICATIONS
Beverly Hills London New Delhi

Copyright © 1984 by Sage Publications, Inc.

All rights reserved. No part of this book may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without permission in writing from the publisher.

For information address:

SAGE Publications, Inc.
275 South Beverly Drive
Beverly Hills, California 90212

SAGE Publications India Pvt. Ltd.
C-236 Defence Colony
New Delhi 110 024, India

SAGE Publications Ltd
28 Banner Street
London EC1Y 8QE, England

Printed in the United States of America

Library of Congress Cataloging in Publication Data

Chan, Julie M. T.
Teacher's guide to designing classroom software.

Bibliography: p.
1. Computer-assisted instruction—Programming.
I. Korostoff, Marilyn. II. Title.
LB1028.65.C48 1984 371.3'9445 84-6962
ISBN 0-8039-2313-9

FIRST PRINTING

TEACHERS' GUIDE TO DESIGNING CLASSROOM SOFTWARE

此为试读,需要完整PDF请访问: www.ertongbook.com

CONTENTS

Introduction

- Why You Should Read This Book
- What This Book Is About
- How You Can Use This Book
- What You Will Learn
- How This Book Is Unique
- Our Philosophy Regarding Computers in Education

This book is written for teachers who want to design their own educational software programs, but who are not sure how to begin. The easiest type of educational software for beginners to design is the Computer Assisted Instruction type of program. CAI programs are widely used and are most helpful to students. These programs provide for greater time on task in order for students to practice skills that require initial exposure; consequently, it might be a good idea for you to begin with a CAI type of program. Then, as you become more proficient at designing educational software or *courseware,* you can "graduate" to designing more creative and complex programs such as simulation, problem solving, modeling, or demonstration type programs.

WHY YOU SHOULD READ THIS BOOK

There's something for everyone in this book . . .

If you are a teacher who is just beginning to use computers, this book will serve as a guide to selecting quality educational software programs. You will learn what features to look for, why they are important, and how they work.

If you are a teacher who is using computers with purchased "off-the-shelf software," but are not always able to find the exact courseware that you need for your instructional objectives, this book will tell you how to design your own.

If you are an educator who has recently acquired programming skills and are now eager to "conquer the world" with programs that are waiting to be written, this book will serve as a guide to designing the courseware, refining your writing, and helping you produce a professional product.

If you are an educator or a computer programmer who has been writing educational software for some time, this book will provide you with techniques to refine your craft. If you are a programmer who is not also an educator, this book contains suggestions that are based on sound principles of teaching and learning that will make your courseware pedagogically more effective.

If you are a student majoring in education, this book will not only help you design educational courseware, it will also show you how to apply the theories of teaching and learning that you have been studying in your methods courses.

WHAT THIS BOOK IS ABOUT

This book is about designing the best possible educational software given the technology that is available at this time, so that you will be able to use the computer to enhance your teaching and so that your students will be able to learn most subjects more quickly and easily in an enjoyable manner.

Chapter 1 involves preliminary planning that requires you to make important choices before you roll up your sleeves and get down to work. Careful planning is the key to creating educational software programs that will be both effective and useful.

Chapter 2 helps you structure the content of your program by showing you how to construct a task analysis, determine benchmark learnings, establish criterion frames, and write the objectives for the program.

After that brief introduction, Chapter 3 is designed to be used as a workbook of step-by-step instructions. It shows you how to develop and use each of the seventeen program modules in the Courseware Design Framework. The modules provide the framework or skeleton upon which to build a courseware program for initial teaching (tutorial), review and reinforcement, or drill-and-practice. The extent to which each module is used will determine whether the final product is a tutorial, review and reinforcement, or drill-and-practice program.

Seventeen modules came to be included in the framework as a result of much time spent in previewing, reviewing, and evaluating educational courseware. We felt that many educational software programs could be strengthened or enhanced if certain elements were included. The seventeen modules constitute an "ideal" program — there is yet to be any one program the authors have looked at that contains all seventeen elements! However, a program can still be of good quality without containing every one of the seventeen elements. But if all of the elements were included in one form or another, the program would be a well thought-out program design.

Chapter 4 contains a sample lesson that illustrates the modules and concepts discussed in Chapter 3.

Chapter 5 discusses techniques for programming your educational software. It contains alternatives on what to do if you are not proficient at computer programming.

Chapter 6 discusses how to evaluate your courseware; several sample evaluation forms are included.

Chapter 7 contains ideas for marketing and selling your completed educational software product. A directory of companies that are developing and marketing educational software will help you decide how to look for a company to help you sell your new product.

The bibliography contains further readings on the topic of software development. Since this is a practical "how to do it" book, much of the educational theory behind program design is not dealt with in this volume. Instead, some of the references provide excellent discussions on program design theory.

HOW YOU CAN USE THIS BOOK

Of course, you can start at the beginning and read through to the end, becoming familiar with how to design educational software along the way. You may also skim through the book until you find a specific area that is of particular interest to you. Following are some guidelines that may help you decide how you can use this book most effectively.

WHAT TO DO BEFORE YOU BEGIN

Before you read this book, decide what *type* of educational software you want to write, *how* you will use it, and *why* you are taking the time (and effort) to write it. If you have spent some time reviewing some of the educational programs that are on the market, you will have a better idea of what most educational software is like. Once you have that background, the ideas discussed in this book will make more sense and the suggested techniques for improving what is currently on the market will be apparent.

WHICH PARTS OF THIS BOOK SHOULD YOU READ?

This book is designed so you can use it whichever way makes the most sense for you, depending on your level of expertise in designing and programming educational software. Each of the sections begins with an overview, followed by a detailed explanation of each concept. Read the chapter overview to determine whether you are interested in or need to read that section.

WHAT YOU WILL LEARN

When you are finished reading this book, you will know how to design a program for educational purposes and be able to construct a task analysis to determine the scope and sequence of the content to be contained in your program. You will be able to recognize the features in quality educational programs and you will have a fundamental idea of the parts of a sound educational program. You will be able to talk to computer programmers and let them know what you want in a program that will help you effectively teach or enable students to practice a given skill or concept. This book will *not* make you an expert programmer, but it will help you become a better educational software purchaser and/or program designer so you can fully exploit the features and potential of the computer.

As an experienced teacher, you are a curriculum specialist in the subject area(s) you teach. Thus, you are in the best position to also design educational courseware since you know *how* students learn — as well as *what* they need to learn . For these reasons, you should become involved in designing educational courseware even if you don't know how to write computer programs. You can always team up with a professional programmer once the design is in place. The design of a program is critical to its effectiveness for students and its usefulness to teachers. Only *you* can do that.

HOW THIS BOOK IS UNIQUE

Although there are other books on how to design educational software, this book is unique in that the Courseware Design Framework is based on the mastery learning model created by Dr. Madeline Hunter of UCLA. All of the effective teaching/learning principles we have used and practiced as successful teachers in the classroom are incorporated in the Courseware Design Framework—from constructing a task analysis for the content items at the beginning to the closure at the end of the courseware.

Sample criterion-referenced pre- and posttests are included to provide examples of how to design easy and accurate placement tests at the proper level of learning and for determining the content, concepts, or skills to be practiced. Specific practice and/or test items are earmarked as *benchmark* or *criterion* items to indicate mastery or understanding of the concept or skill being studied.

此为试读，需要完整PDF请访问：www.ertongbook.com

Another unique feature is that the scope and sequence is based on Bloom's Taxonomy of the Cognitive Domain, which orders items from simple to complex in terms of levels of thinking. Items are coded through an item analysis so that a prescription with follow-up activities can be built in.

The courseware should be coordinated with other teaching materials so that teachers can make use of instructional materials they already have in their classroom. In addition, this courseware coordination makes it easier for teachers to integrate the use of computers into all areas of the curriculum effectively.

OUR PHILOSOPHY REGARDING COMPUTERS IN EDUCATION

We believe that the human element is crucial in the teaching/learning process. The teacher is the curriculum specialist and is the person most knowledgeable about how students learn. Their education, experience, and expertise can never be replicated by a computer. Therefore, we feel that *the computer can never replace the teacher.* However, the computer can be harnessed to make the teacher's job more effective. It can free teachers to concentrate on the one thing they do best—teaching—and providing the guidance and warm encouragement that only a human being can give!

Computer courseware should be used in the same manner that educational films are used in the classroom. For example, before a film is shown to students, teachers normally (1) preview the film, (2) plan for its use in a purposeful manner as a tool to supplement, augment, or enrich their teaching, (3) make sure it promotes the instructional objective(s), (4) provide students with a purpose for viewing the film by giving them points to look for, and (5) follow up the viewing with a discussion or other meaningful activity to get the most out of the time invested viewing the film. These same steps should be applied to using educational software programs.

1 Preliminary Planning

- Identification Data
- Content
- The Student Population
- Program Technique
- Content Structure
- Physical Design
- Presentation Formats
- Evaluation
- Collateral Instructional Materials
- Courseware Coordination

It is important to think through carefully what you will write for use on the computer before you begin its actual design. That way, you will know exactly *what* you are producing and *how* you will use it. This section of the book forces you to make some important choices before you actually roll up your sleeves and get down to work designing a piece of educational courseware. *Careful planning* is the key to creating a program that will be both effective and useful. The following sections should be given careful consideration before you begin the actual courseware design process.

IDENTIFICATION DATA

In this section, include the following information: (1) your name, (2) the starting date, (3) the completion date, and (4) the approximate hours it took you to work on the project. Although it is not necessary to do item four, keeping track of how long it takes you to write a program will enable you to estimate more accurately the time it will take you to work on future projects.

CONTENT

SUBJECT AND CONTENT AREA

Decide on the subject or content teaching area. Subjects such as reading, language arts, math, science, and social studies are popular. Now that computer literacy is becoming a subject for study, this could be yet another content area (and one for which there are relatively few teaching materials).

TOPIC OR SKILL

Within each subject, there are specific topics and/or skills to be mastered. Courseware that teaches a transferable skill and fits in with most academic subjects will be very popular.

INSTRUCTIONAL OBJECTIVE

The objective of your program should be clearly stated at the *beginning* of any collateral material, such as a study guide or teacher's manual, for the convenience of potential users such as teachers and parents. In addition, the objective should be stated in the courseware program itself so students will be made aware of what they are learning, why they are learning it, and how it will be useful to them. If an objective for a program cannot be defined or expressed in one or two sentences, then it may not have been clearly thought through by the courseware designer and needs to be rewritten.

THE STUDENT POPULATION

GRADE LEVEL(S)

The grade level(s) should be clearly identified in the written literature *and* on the disk label. Programs that claim to be appropriate for grades kindergarten through 12 are generally useless because no single teaching material could possibly be "all things to all learners."

ABILITY LEVEL(S)

The ability level should also be specified because one group of third-grade students who are identified as gifted would be very different from another group of third graders who are remedial. Thus, ability level must be kept in mind when designing (and selecting) appropriate courseware.

LEARNING STYLE(S)

Learners can interact with the computer through several channels: visual, auditory, and kinesthetic. Because of this multisensory aspect of computer learning, children with different learning styles can find computers equally beneficial.

PROGRAM TECHNIQUE

There are three main ways that educational courseware is being used in today's classroom: (1) Computer Assisted Instruction, (2) Computer Enriched Instruction, and (3) Computer Managed Instruction. These three ways are also referred to as *program techniques*. Within each of the three techniques, there are further subcategories, called *program functions*. In the paragraphs below, the program functions for each of the three main techniques will be explained.

COMPUTER ASSISTED INSTRUCTION

Programs that are used to teach, reteach, review, or provide for drill and practice after a skill or concept has been previously taught are referred to as computer assisted instruction or CAI software. Tutorial type programs are used to teach a skill "from scratch" and are useful for capable students who are able to learn advanced concepts on their own. Programs that are designed to *reteach, review,* or *reinforce* are useful for students who need a second "go around" on a concept or learning, for students who were absent when the concept was initially presented by the teacher, or for students who need to be retaught a particular skill or concept. Students who need more practice or additional "time on a task" in order to master it benefit from courseware that is of the *drill-and-practice* function. Of all educational courseware sold in 1983, 75% fall into the CAI category. Of this, 90% of all CAI programs were drill-and-practice in nature.

COMPUTER ENRICHED INSTRUCTION

Courseware in this category enables teachers to enrich and enhance their classroom instruction. The three primary functions under CEI include simulations, modeling, and problem solving. In *simulation* programs, the computer is used to "imitate real life situations" and the user becomes the main character or actor in control of the situation. *Modeling* programs enable users to try out or test ideas. For example, in redesigning a school cafeteria, an architect can use a computer to determine the best floor plan for handling a maximum traffic flow during peak use hours. Students can experience the process of *problem solving* because situations can be easily posed on the computer and, in order to be successful, students must (1) define the problem, (2) develop a plan, (3) carry it out, and (4) evaluate it.

COMPUTER MANAGED INSTRUCTION

Computers can also be used to assist teachers in routine classroom management tasks such as test preparation, administration, and scoring; using test results on which to base additional instruction; and other routine tasks such as record keeping, attendance, scheduling, and so on. Although this category of software does not directly affect students per se, computers can reduce the time spent on time-consuming routine administrative tasks and free the teachers for more time to work directly with students.

CONTENT STRUCTURE

Content can be approached in various ways on a computer. For example, an idea can be presented in its simplest form and built upon until it becomes fully developed. Or, it can start with simple examples that can later be generalized. Finally, it can start with a general concept that is then applied to many examples. Any of these approaches may be used in designing your courseware, but the final decision will depend on the nature of the content, the skill being presented, and the ability level of the learners.

PHYSICAL DESIGN

There are five physical designs for educational courseware: (1) linear designs, (2) spiral designs, (3) branching designs, (4) multitrack designs, and (5) regenerative designs.

LINEAR DESIGN

Linear design courseware takes all learners through a program from start to finish using the same set of items in the same sequence regardless of ability. Although it may

此为试读，需要完整PDF请访问：www.ertongbook.com