DOCUMENT RESUME

ED 158 724 IR 006 178

AUTHOR	Black, Dean; Bunderson, C. Victor
TITLE	Teaching Instructional Design on TICCIT.
INSTITUTION	Brigham Young Univ., Provo, Utah. Inst. for Computer Uses in Education.
REPORT NO	ICUE-TR-18
PUB DATE	12 Jul 76
NOTE	103p.; For related documents, see IR 006 172-177 and 180-183
EDRS PRICE	MF-$0.83 HC-$6.01 Plus Postage.
DESCRIPTORS	*College Teachers; *Computer Assisted Instruction; *Course Descriptions; Course Objectives; Display Systems; Educational Principles; Higher Education; Individualized Instruction; *Instructional Design; *Teacher Education
IDENTIFIERS	*TICCIT Computer System

ABSTRACT

An individualized instructional development training program was developed that allows teachers to enter at several points and move through the instruction in a way that suits their own particular needs and challenges. The program was designed for implementation on the TICCIT computer controlled television system, although it could be adapted to any medium. Seven principles guided the course design: (1) simplicity, compatibility, and quality; (2) prescriptive instruction; (3) starting where teachers are; (4) production of lesson materials immediately useful to teachers; (5) a "zoom lens" feature; (6) simple worksheets; and (7) case study examples. The course summary included contains a course map, introduction, and four units; each unit is divided into lesson and segment sub-categories. Unit 5 was still in preparation. (CMV)

INSTITUTE FOR COMPUTER USES IN EDUCATION

TEACHING INSTRUCTIONAL DESIGN ON TICCIT

Dean Black
C. Victor Bunderson

ICUE Technical Report No. 18

July 12, 1976

TEACHING INSTRUCTIONAL DESIGN ON TICCIT

By Dean Black and C. Victor Bunderson
Institute for Computer Uses in Education

For some time, BYU's Teacher Development Committee has been carrying on a university-wide program to upgrade instruction. Many teachers have taken the Committee's courses to find new ways to help their students learn.

There are some built-in limitations to an instructional improvement program based primarily on classroom courses: a classroom course can serve only a few teachers at a time, and everyone must enter at the same time and progress at about the same rate. Each teacher's needs are different, and they will not be met by having teachers move together through a classroom experience in which they receive general teachings on all aspects of instruction, including those that they already do well.

During the past four months, people from the Institute for Computer Uses in Education have developed an individualized instructional development training program that lets teachers enter at several points and move through the instruction in a way that suits their own particular challenges. The program has been designed to be implemented on the TICCIT computer-controlled television system, although it could be adapted to any medium.

THE NEED

Evaluations of BYU teachers show that they score highly on overall ratings of teaching effectiveness. Students find them organized, scholastically prepared, and motivating. However, other areas show

up as points for improvement. BYU teachers tend to get below average ratings on defining and achieving objectives, explaining things clearly, and making helpful evalutations of their students' work.

Recent research at BYU has shown that these factors are among the most important in helping students learn. Dr. M. David Merrill has developed a method of evaluating instruction called the Instructional Strategies Diagnostic Profile. It emphasizes the points for improvement that are listed above. Dr. Merrill and his students have shown that lessons that meet the criteria listed in the Instructional Strategies Diagnostic Profile teach better than those that do not. In fact, well designed instruction seems to minimize the effect of differences in the intellectual ability of students and differences in the effectiveness of teachers.

Our TICCIT experiences seems to show us that evaluations are an especially important area of need. TICCIT is test-heavy. Given the usual frame-of-mind of most students, that should be a big black mark for TICCIT. However, students say that they <u>like</u> the tests on TICCIT. They would <u>not</u> like to see fewer tests. Tests are obviously an important part of the learning system -- one that students can enjoy and appreciate. But can anyone doubt that classroom tests often miss the mark and thereby become a burden to everyone?

The BYU General Education Program is also a test-heavy system. Students will eventually be required to pass evaluations in communicative skills like reading, writing, and listening, in mathematics, in health and physical fitness, and in citizenship education. They must pass other evaluations in the arts, in social systems, in natural science, and in perhaps other areas as well. All of those evaluations must be prepared by teachers and approved by the General Education Program

administrators. The program obviously places high demands on the instructional skills of BYU teachers.

In designing the TICCIT-based teacher development program, we imposed upon ourselves the condition that nothing could take from teachers their full stewardship. One might argue that quality instruction can come only from a cooperative effort between teachers and professional instructional scientists. While it is true that the best results might come from such cooperative efforts, it is likewise true that most teachers will never work with an instructional scientist. They must meet their stewardship alone. We have therefore tried to design a system that will let teachers produce quality instruction without the intervention of instructional scientists. This is not intended to discourage teachers from working with instructional scientists; we simply admit that such relationships are the exception rather than the general case. The program that will meet BYU's needs must be capable of becoming the general case.

DESIGN PRINCIPLES

The TICCIT-based instructional development program should give teachers a readily available individualized way to learn how to improve their instruction. If it truly meets BYU's needs, it must be successful with dedicated and concerned teachers who, nonetheless, are reluctant to ask for help (even from people who call themselves instructional scientists), who do not like jargon, who are impatient with explanations that do not quickly lead to products they can use, and who, in many cases, question whether they even need the course.

In an effort to meet that perhaps unmeetable challenge, we set out seven principles that have guided the course design:

1. The course must be simple -- in fact, as simple as possible, while still leading to a quality instructional product. And, despite its simplicity, it should be compatible ("upward compatible" as they say) with more complex instructional development approaches, in case some teachers want to go to greater depth.

2. The course should be prescriptive; that is, it should put forth a set of step-by-step procedures.

3. The prescriptive procedures should start with the kinds of thinking teachers already do, and lead them gradually into new principles and concepts.

4. The procedure should lead as quickly as possible into the production of actual lesson materials which are immediately useful to teachers.

5. The course should have a "zoom lens" feature, where the teacher can keep the integrated process in mind, and yet move to greater detail if he needs to.

6. The course should be supported by worksheets that are simple and, insofar as possible, non-constraining, but that help the teacher keep track of the decisions he must make about the instruction he is preparing.

7. The examples should be case studies that show how real teachers have applied the procedures, how they had to fudge a bit or change the process from time-to-time, and the results they got.

PRINCIPLE 1: SIMPLICITY, COMPATIBILITY, QUALITY

Everyone desires simplicity, or at least everyone seems to claim to desire simplicity. Therefore, when we say that we strove for simplicity in the course design, we in some ways say very little. Nevertheless, "simplicity" deserves its place as design principle number one because we worked very hard at it: it was one of the major criteria against which we checked all of our design decisions.

The approach. Our basic approach was to simplify by eliminating. That is, we reviewed existing systematic approaches to instruction and deleted parts of them to reduce complexity. However, eliminating parts of a systematic process can be complicating and devastating. To say that something is systematic means that each step derives

from what has gone on before and is a basis for what is to come; deleting parts of the system can make it break down later on.

The first step in the simplification was therefore a study of the interdependent relationships between the many steps of several closely related approaches to instructional development, including the procedures used by the authors of the TICCIT course materials developed here at Brigham Young University, Courseware, Incorporated's "systematic approach," and the work of Dr. M. David Merrill. We reviewed the approaches using a matrix procedure which let us trace the impact of each step on later parts of the process. This meant that, if we felt a given step might be eliminated, we could review the matrix and determine how that would impact on the process.

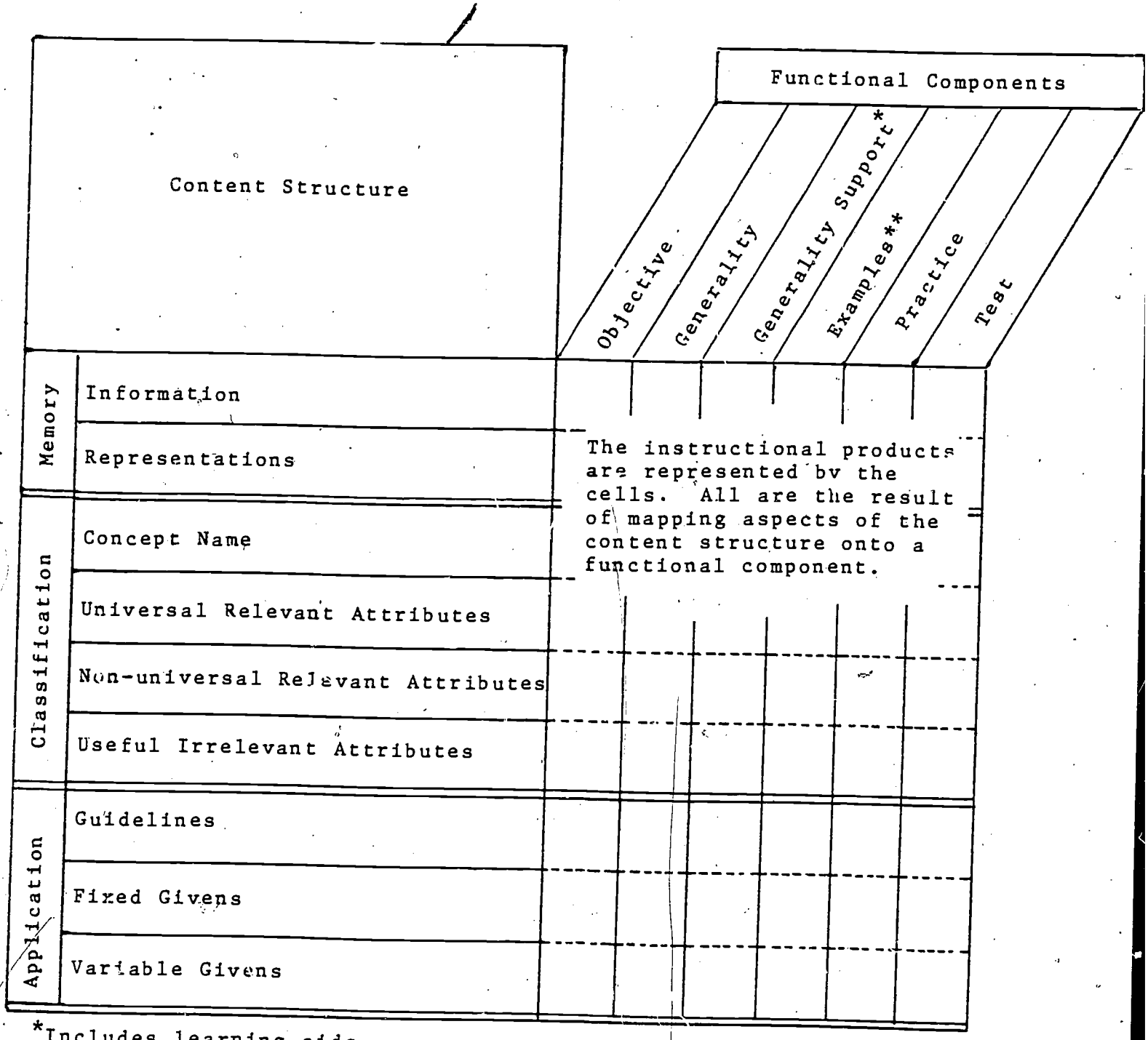
The result of our analysis was a set of procedures that represents only a part of what might be covered in a complete exposition of any of the approaches we reviewed. However, what remained appeared to us to be fundamental, integrated, capable of bringing about substantial increments in the quality of teacher-produced instruction, and completely compatible with those parts of the complex processes that we chose not to include in the course.

Elements of the process. Figure 1 shows the basic elements of the instructional development process that are covered in the "simplified" course. There are elements which represent content structure and there are functional instructional components.

Content structure means nothing more than ways of breaking up content information into smaller, more understandable, pieces. What those pieces are depends upon what the students are to do with what they learn. When memory-level learning is the goal, the elements of content structure are the information to be memorized and the way that

Figure 1

ELEMENTS OF THE INSTRUCTIONAL DEVELOPMENT SYSTEM



*Includes learning aids.
**Includes non-examples for classification-level instruction.

information is represented to the student. When the goal is classification-level learning (where the student shows he knows a concept by correctly classifying examples and non-examples), the elements are the concept name, universal and non-universal relevant attributes, and useful irrelevant attributes. When the goal is application-level learning (where the student applies his learning in the production of some outcome), the elements of content structure are the guidelines (or rules, or hints, or suggestions) that the student is to follow, and the givens he works with, both fixed and variable.

The elements of content structure are the basis of the first part of the development process. The teacher decides what kind of learning outcomes he wants, and then, depending on his choice of learning outcomes, he analyzes his content information into the elements of content structure that are shown on Figure 1.

Mere definition of the content structure is not enough. The content must be prepared as components of instruction with a defined purpose or function. The functional components onto which the teacher "maps" his content structure in order to produce instructional materials are also listed on Figure 1. They are the objective, the generality, the generality support, examples, practice, and the test. In the second part of the development process the teacher transforms the content structure into these functional instructional components that meet the students learning needs.

Instructional quality. The principles covered in this simplified approach to instructional design are but a subset of all that could be taught. Nevertheless, instruction prepared according to the procedures presented in the course has the following characteristics that have been shown to help students learn:

- o The instruction includes a complete set of instructional components.
- o All of the components are consistent with one another.
- o The components are presented separately so the student can move easily from one to another to meet his individual learning needs.
- o The essential information is separated from all elaborations of that information and other related material which is simply nice-to-know.
- o The instruction covers the full range of significant variation or "divergence" within the content area.
- o Non-examples are used where appropriate, and they are matched with examples to accentuate the differences between them.
- o Examples are prepared in a variety of representation forms.
- o The instruction includes learning helps.
- o Efficient item sequences based on divergence, matching, and difficulty are used in instruction, but random item sequences are used in practice and tests.
- o Practice items and tests are accompanied by helpful feedback.
- o Tests avoid most of the common pitfalls of teacher-made tests.

PRINCIPLE 2: PRESCRIPTIVE INSTRUCTION

It seems reasonable that people who are learning in order to meet pressing needs will find themselves most motivated to continue when they can see that what they are learning clearly meets their needs. That is why we made our generalities prescriptive rather than explanatory.

At this point, you might turn to the last section of this paper and review a few of the generalities. You will see that, for the most part, they lead the teacher step-by-step through the creation of instructional products. Teachers who work through the generalities should find that each thing they learn is a "building block" that makes a clear contribution to the production of useful instructional materials. There will be explanations and justifications that show why things

are done the way they are, but they will not be in the generality; rather they will be in the generality support where the teacher can read them or ignore them as he chooses.

As we tried to specify prescriptions for such a complex and comprehensive process, we had to make some arbitrary choices that we may want to change later when we have had some experience with the course. For example, in the section on classification-level instruction, we teach the teachers to identify ways examples of a concept class may vary. Non-universal relevant attributes define subclasses within the concept set, and there are irrelevant attributes which introduce variation. Instances may therefore vary or diverge in two ways: they may belong to different subclasses of the concept class, and they may be characterized by different states of the irrelevant attributes.

The teacher specifies an instance by listing the attributes, or variations of attributes, that will characterize it. A teacher could specify a complete set of instance types in which every possible combination of variations occurs. But that may be neither necessary nor wise. How, then, do you decide when you have specified enough instances? We decided that a set is complete when each variation occurs twice. But we may change that and other arbitrary points when we see what happens to teachers when they try to follow the directions we wrote, and when we see what happens to students when they study the materials teachers prepare by following the directions we wrote.

PRINCIPLE 3: START WHERE TEACHERS ARE

The phrase "start where teachers are" describes a second aspect of our effort to make the instruction immediately useful to teachers. The prescriptive procedures are based on a specific way of thinking

about instruction. Most teacher think in a different way. Learning the procedures means most fundamentally learning a different way of thinking.

Shall we therefore teach them the different way of thinking so they can then learn the procedures? Or can we teach them the procedures and let that be the way they learn the different way of thinking. We chose the latter course -- to jump into the procedures with no prior preparation in the conceptual framework that underlies the process. But to do that, we had to begin with steps that work on things the teachers are already used to thinking about.

Teachers generally think about their instruction in terms of topics. Instructional developers generally insist that teachers no longer think about topics, but about behavioral objectives instead. That shift in thinking is often a stumbling block for teachers who want to learn development procedures, especially if the procedures won't let them go on until they make that shift.

Since teachers think about topics, we chose to begin with topics. The beginning procedures transform those topics into the behavior part of a behavioral objective. I emphasize the "behavior part" because "good" behavioral objectives also include conditions and standards. However, our matrix analysis of the instructional development process showed that the conditions and standards parts of the objective are not needed in the continuation of the process as soon as the behavior part. So our first goal became this: to help the teacher think about his topic in terms of student behaviors; the conditions and the standards come later.

There are several ways to justify thinking about student behaviors as you begin the development process. One is the typical argument

in favor of behavioral objectives. Another goes like this:

> Both teachers and students must work if learning is to take place. Teachers work in the preparation of instruction, but when time for learning comes, the most learning occurs when the student, not the teacher, does the work. Therefore, an important part of the teacher's job as he prepares is to figure out things for the student to do with the material he is to learn -- in other words activities that help him learn.

It is our judgment that this rationale "fits" the way teachers think better than the typical argument in favor of behavioral objectives. We therefore do not refer in the beginning phases of the course to behavioral objectives, but rather to "learning activities," and the development process is presented as a way of preparing learning activities that help students learn.

When the time comes, the teacher adds the conditions and standards and he ends up with a behavioral objective. We believe that, by that time, the reason for the objective will be clearer than it is during the first steps of development.

We hope that instruction based on learning activities will be just as good as instruction based on full-blown behavioral objectives. We do feel confident that if there is any loss in quality, it will not be substantial and it will be more than offset by the greater ease with which teachers may move into what for them is the "meat" of the development process. We will look at this point as part of our evaluation.

PRINCIPLE 4: ON TO LESSON DEVELOPMENT

One of our "simplifications" was to do away with "front-end" content analysis as part of the course. Not that it should not be done as part of instructional development; simply that we chose not to teach it now.

We therefore disregard the idea of hierarchies and topic interrelationships and ask the teacher to choose a single topic that is important to him and seems to give the students trouble. Working with one topic, he can try out the procedures and begin to learn the different way of thinking that they imply. When he's finished one topic, if he is so inclined he may try another.

By working with one topic at a time, the teacher can move directly into lesson preparation, an activity that is probably more immediately rewarding to him than an arduous content analysis. To be sure, the course organization may need upgrading, but the teacher will at least be working an area that he has judged to be important and in need of improvement.

Some content analysis naturally falls out as a consequence of clearly specifying lesson content. Our experience seems to show that when a teacher has analyzed content structure as he is taught in this course, prior learning steps become apparent. In fact, it is hard to imagine anyone doing a successful content analysis without understanding the "top box" content with a rigor like that imposed by the procedures taught in this course. Without that rigor, how can one be sure his content analysis decisions are correct? So we have skipped content analysis, both to get quickly down to the brass tacks of lesson development and also with the thought that the teacher will better understand content analysis when he knows the principles that have been included in the course.

PRINCIPLE 5: THE "ZOOM LENS"

There are many details in something as complex as what we're looking at here, despite our hopes for its simplicity. And each of

those details makes sense only because of what it contributes to the whole. That poses a problem for learners, and for those who are trying to help people learn: learning must cover both the detail and its place in the whole

We have tried to meet that problem with a "zoom lens" sort of organization that will let teachers study the process at both a global and a detailed level. The global representation presents the basic steps with little explanation of how they are accomplished. Each step in the global representation that is not self-explanatory includes a reference to a "how-to" explanation that provides the detail.

This means that the teacher can step through the process at a global level, "zooming" in for detail when things aren't clear. Similarly, whenever the teacher is studying a detailed explanation, he may jump for a moment to the global level to see where it all fits in the context of the whole.

The global presentations are in the introductions, or in what, on TICCIT, are called mini-lessons. There is a general course introduction as well as an introduction to each unit and lesson. The course introduction is a global presentation of the entire process. Each unit introduction is a global presentation of a substep within the process. Lesson introductions, in turn, cover parts of the process described in the unit introduction.

All introductions are on the same level of detail. A unit introduction is a compilation of lesson introductions; the course introduction is a compilation of unit introductions. This means that the global information is presented three times so that the teacher always has easy access to it.

The detail is presented in the segments that make up each lesson. Each detailed segment covers one of the steps presented in the lesson introduction.

The best learning strategy is probably one in which the teacher maintains a sort of "moving home base" in one of the introductions, dipping readily into segment-level detail, and back again to home base. We have tried to encourage that strategy by having very little integrating information in the segments.

PRINCIPLE 6: SIMPLE WORKSHEETS

Worksheets are often more trouble than they are worth. They force those who use them to write in sometimes-too-small spaces, and they impose order that may be counter-productive when people should let their thoughts run free. Nevertheless, we decided to use worksheets because they can help people keep track of things in the early stages of learning when it's hard to keep track of things and not easy to let one's thoughts run free without getting lost. One thing teachers may learn as they use the worksheets is to do just fine without them.

The worksheets are places to record decisions. The basic decisions are: (1) What shall I choose as a learning activity? (2) What should go in the generality? And (3) what will my instances (examples and practice) be like? Since these are the three basic decisions, the three basic worksheets are (1) the Learning Activity Worksheet, (2) the Generality Worksheet, and (3) the Instance Worksheet. Examples of the worksheets follow on the next three pages. If you want to know how teachers will use the worksheets you may review the course introduction.

PRINCIPLE 7: CASE STUDY EXAMPLES

No set of simplified procedures can cover all the problems people

LEARNING ACTIVITY WORKSHEET

Topic	Learning Activity	Activity Type		
		Application	Classification	Memory

GENERALITY WORKSHEET

Topic:

Learning Activity:

Generality:

INSTANCE WORKSHEET

Topic:

Learning Activity:

Attributes or Givens	Instance Types									
	A	B	C	D	E	F	G	H	I	J

Items needed for examples:

Items needed for practice:

ems needed for test:

ems needed for replacement or other tests:

will run into. If, as a course designer, one tries to handle everything that might go wrong, instruction becomes burdened with exceptions and warnings. The main message -- the thing that keeps people on track through the troubles -- can get lost.

We chose to keep the procedures simple, to trust teachers to be clever and adaptive, and to show them just how clever they'll have to be by using real case studies as examples. As course authors, Diane Thompson and I prepared complete lessons following the procedures outlined in the course. As we worked, we recorded our thoughts. Those thoughts are the basis for case studies, and they show that we did have to hedge a bit here-and-there, that everything wasn't as easy as the generalities suggest, but that things worked out in the end because we were guided by the generalities, but not constrained by them.

During the first implementations of the course, we will ask teachers from several subject matters to do the same thing. Their recorded case studies will become part of the example pool, and they will become contributing authors.

Each case study covers the complete development process. That means that a single example can be followed from the beginning of the course to the end. This is important. The development process is systematic; each step contributes to the others. If the examples from step-to-step are unique -- if they bear no relationship to one another -- they cannot show that so-important interdependency.

GENERALITIES

Most of what has been expressed somewhat abstractly in the preceding pages is illustrated in the generalities that follow. The course map shows how the generalities are organized. The course introduction

(as explained earlier) is a global presentation of the entire process. The five units cover aspects of the process.

Unit 1 generalities teach about choosing a learning activity. Units 2, 3, and 4 cover the planning and preparation of instructional materials and tests, each for a different kind of learning activity. Unit 2 generalities teach about memory-level instruction; Unit 3 covers classification-level instruction; and Unit 4 is about application-level instruction. Unit 5 tells how the developed instruction can be implemented in three formats: lecture, workbook, and computer-assisted instruction.

You will notice that there are page numbers by the unit boxes on the course map. As you would suspect, each page number tells where the section for its corresponding unit begins.

The first page of each unit section is a unit map. The unit introductions are global presentations of that part of the process covered in the unit. Lessons are the subdivisions within the unit, and there is a number by each lesson box that tells where the corresponding lesson section begins.

The pattern is repeated at the lesson level. The first page of each lesson section is the lesson map; the introduction is a global presentation, and the segments cover the details of the process. Page numbers on the lesson map direct you to the generalities for each segment.

The following page is an outline-type summary of all of the unit, lesson, and segment titles. Maps and generalities follow.

COURSE SUMMARY

Course Map 23

Course Introduction 24

Unit 1 Map - SPECIFYING LEARNING ACTIVITIES 32

Lesson 1.1 Map specifying learning activities 33

Introduction 34

Segment 1.1.1 - Levels of learning activity 35
Segment 1.1.2 - Checking for application-level learning activities 36
Segment 1.1.3 - Checking for classification-level learning activities 37
Segment 1.1.4 - Checking for memory-level learning activities 38

Unit 2 Map - APPLICATION-LEVEL INSTRUCTION 39

Lesson 2.1 Map - Planning application-level instruction 40

Introduction 41

Segment 2.1.1 - Specifying guidelines 42
Segment 2.1.2 - Specifying givens 43
Segment 2.1.3 - Writing the application-level generality 44
Segment 2.1.4 - Specifying divergent application-level examples 45

Lesson 2.2. Map - Preparing application-level instruction 46

Introduction 47

Segment 2.2.1 - Application-level student objectives 48
Segment 2.2.2 - Application-level generality supports 49
Segment 2.2.3 - Application-level learning aids 50
- (page 2) 51
Segment 2.2.4 - Preparing application-level examples 52
Segment 2.2.5 - Application-level practice 53

Lesson 2.3 Map - Preparing application-level test 54

Introduction 55

Segment 2.3.1 - Application-level test formats 56
- (page 2) 57

Segment 2.3.2 - Writing application-level test items 58
- (page 2) 59
- (page 3) 60

Unit 3 Map - CLASSIFICATION-LEVEL INSTRUCTION 61

Lesson 3.1 Map - Planning classification-level instruction 62

Introduction 63
(page 2) 64

Segment 3.1.1 - Relevant attributes 65
Segment 3.1.2 - Irrelevant attributes 66
Segment 3.1.3 - Writing the classification 67
Segment 3.1.4 - Specifying divergent examples 68
Segment 3.1.5 - Specifying matched non-examples 69

Lesson 3.2 Map - Preparing classification-level instruction 70

Introduction 71

Segment 3.2.1 - Classification-level student objectives 72
Segment 3.2.2 - Classification-level generality supports 73
Segment 3.2.3 - Classification-level learning aids 74
Segment 3.2.4 - Collecting an instance pool 75
Segment 3.2.5 - Classification examples and non-examples 76
Segment 3.2.6 - Attribute isolation 77
Segment 3.2.7 - Classification-level practice 78

Lesson 3.3 Map - Preparing classification-level tests 79

Introduction 80

Segment 3.3.1 - Classification-level test formats 81
Segment 3.3.2 - Writing classification-level test items 82
- (page 2) 83

Unit 4 Map - MEMORY-LEVEL INSTRUCTION 84

Lesson 4.1 Map - Planning memory-level instruction 85

Introduction 86

Segment 4.1.1 - Writing the memory-level generality 87
Segment 4.1.2 - Specifying alternative representations 88

Lesson 4.2 Map - Preparing memory-level instruction 89

Introduction 90

Segment 4.2.1 - Memory-level student objectives 91
Segment 4.2.2 - Memory-level generality supports 92
Segment 4.2.3 - Memory aids 93
- (page 2) 94
Segment 4.2.4 - Memory-level practice 95

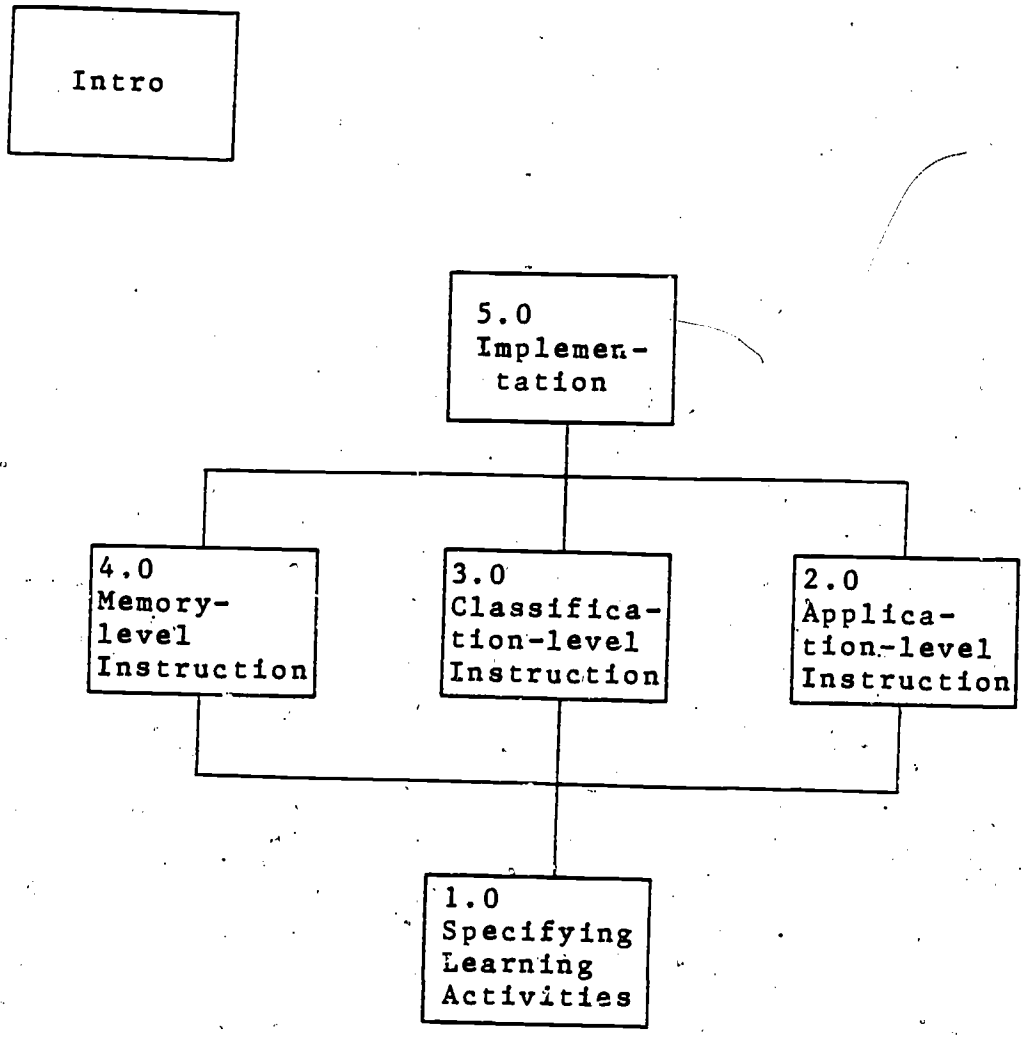
Lesson 4.3 Map - Preparing memory-level tests 96

Introduction 97

Segment 4.3.1 - Memory-level test formats 98
Segment 4.3.2 - Writing memory-level test items 99
- (page 2) 100

Unit 5 Map - IMPLEMENTATION 101

Note: This section is still in preparation.


Intro
5.0 Implementation
4.0 Memory-level Instruction
3.0 Classification-level Instruction
2.0 Application-level Instruction
1.0 Specifying Learning Activities

Course Map

COURSE INTRODUCTION

This introduction is an overview of the instructional development process taught in this course. Parts of it are repeated at the beginning of each unit and lesson. We've added the overviews so that, as you work, you can keep track of where you are. They include references to segments where you can find more detail if you need it.

THE PROCESS

First, decide what you will have your students do as a learning activity. (This is covered in Unit 1.)

1. Review your course and select a topic for development that:

   - o relates to an essential course outcome, and.....
   - o seems to give the students trouble.

2. Students will not learn a topic unless they actively work with it in some way. Specify meaningful student learning activities for your topic at the application, classification, or memory levels. (See segment 1.1.1 for the difference between the three levels.)

   - o CHECK FOR APPLICATION-LEVEL LEARNING ACTIVITIES FIRST. (See segment 1.1.2 for guidelines and examples.)

     Write any application-level learning activities you have identified on the Learning Activity Worksheet and check the "Application" column.

     (Note: A topic can be too broad to let you think easily about the problems you may face when developing instruction. If your topic seems too broad -- if you feel a bit overwhelmed as you try to specify learning activities for your topic -- break it into subtopics and consider each one separately.)

   - o CHECK FOR CLASSIFICATION-LEVEL LEARNING ACTIVITIES NEXT. (See segment 1.1.3 for guidelines and examples.)

     Write any classification-level learning activities you have identified on the Learning Activity Worksheet and check the "Classification" column.

   - o CHECK LAST FOR MEMORY-LEVEL LEARNING ACTIVITIES. (See segment 1.1.4 for guidelines and examples.)

     Write any memory-level learning activities you have identified on the Learning Activity Worksheet and check the "Memory" column.

What you do next depends on the kind of learning activity you specified in the previous step. If you specified an <u>application-level</u> learning activity, continue with the section on the next page. The section on <u>classification-level</u> instruction begins on page . If you specified a <u>memory-level</u> learning activity, continue on page .

Application-level instruction (covered in Unit 2). Plan the generality and instances for application-level instruction as follows:

1. List any guidelines, general or specific, that will direct the student's activities as he produces the outcome specified in the learning activity.

   (See segment 2.1.1 for guidelines and examples.)

2. Review the guidelines and mentally work through the process the student is to learn. Identify all of the givens and list them.

   (See segment 2.1.2 for guidelines and examples.)

3. List the relevant variations, if any, for each given.

   (See segment 2.1.2 for guidelines and examples.)

4. Review the relevant variations for each given and cross out the ones that you do not intend to teach.

5. List the givens on an Instance Worksheet.

   - o List first all of the fixed givens, that is, those that have no relevant variations that you intend to teach.
   - o List next the variable givens, that is, those that do have relevant variations that you intend to teach.
   - o For each variable given, list its relevant variations and number them for easy reference.

6. With reference to the guidelines you have written and the list of givens on the Instance Worksheet, write an application generality on the Generality Worksheet.

   (See segment 2.1.3 for guidelines and examples.)

7. Specify a set of divergent examples on the Instance Worksheet.

   (See segment 2.1.4 for guidelines and examples.)

Based on your plan, prepare your application-level instructional segment to include an objective that presents the learning activity to the student, a generality that presents the instruction that the student will apply in order to produce the outcome specified in the objective, a generality support that presents other helpful or interesting information, examples that demonstrate the production of the outcome according to the information presented in the generality, and practice that requires that the student produce the outcome or a representation of the outcome.

1. Write a student objective that will direct the student in his learning activities.

2. Review the generality as you wrote it on the Generality Worksheet to be sure it is consistent with the objective. If it is not consistent, revise one or the other as needed.

3. Write a generality support that includes an application-level learning aid.

   (See segment 2.2.2 for general guidelines and examples for generality supports. See segment 2.2.3 for guidelines and examples relative to application-level learning aids.)

4. Prepare a set of divergent examples.

   (See segment 2.2.4 for guidelines and examples.)

5. Prepare a set of application-level practice problems.

   (See segment 2.2.5.)

Prepare a test for your application-level instruction as follows:

1. Select the test format or formats from among the following:

   - o True/False
   - o Matching
   - o Multiple Choice
   - o Completion/Fill-in
   - o Short Essay
   - o Simulation
   - o Actual Performance

   Do not use more than two or three formats in a single test.

   (See segment 2.3.1 for descriptions of the test formats.)

2. For each example type you specified on the Instance Worksheet assign an <u>actual</u> given for each given or relevant variation of a given that you listed on the Instance Worksheet.

3. Produce the givens in a form that is compatible with the format or formats you selected for presenting the test items.

4. Work through each item to produce not only the solution, but also likely wrong answers. Use the right answers for test feedback; the likely wrong answers are good distractors (incorrect response options) for objective test formats.

5. Prepare test items according to the format

Classification-level instruction. Plan the generality and instances for classification-level instruction as follows:

1. Make a list of the relevant attributes of the concept.

   - Divide the list into two parts: universal relevant attributes and non-universal relevant attributes.
   - Review the list of non-universal relevant attributes and identify groups of attributes that go together to define specific subclasses of the concept.
   - Rewrite the list of non-universal relevant attributes to show each subclass separately. The subclasses, taken together, must cover the entire concept class.

   (See segment 3.1.1 for guidelines and examples.)

2. Make a list of the useful irrelevant attributes.

   - Each useful irrelevant attribute may assume more than one state. For each useful irrelevant attribute, list the separate states it may assume and number them for reference. The set of states for an irrelevant attribute must cover the entire concept class.

   (See segment 3.1.2 for guidelines and examples.)

3. Write the attributes you've defined on the Instance Worksheet.

   - List separately each universal relevant attribute.
   - List the non-universal relevant attributes grouped by subclass.
   - List separately each useful irrelevant attribute, showing for each the numbered set of states it may assume.

4. Based on your list of attributes of the concept class, write a classification generality on the Generality Worksheet. Write it as you intend to present it to the students.

   (See segment 3.1.3 for guidelines and examples.)

5. Review the generality and underline with a red pencil, or in some other noticeable way, any terms or concepts that may be unfamiliar to the students.

6. Specify a set of divergent examples on the Instance Worksheet.

   (See segment 3.1.4 for guidelines and examples.)

7. Specify a set of matched non-examples on the Instance Worksheet.

   (See segment 3.1.4 for guidelines and examples.)

Based on your plan, prepare your classification-level instructional segment to include an objective that presents the learning activity to the student, a generality that gives the basis on which the student will make his classification, a generality support that presents other helpful or interesting information, examples of correctly made classification, and practice that gives the student an opportunity to make classifcations based on the information in the generality.

1. Write a student objective that will direct the student in his learning activities.

   (See segment 3.2.1 for guidelines and examples.)

2. Review the generality as you wrote it on the Generality Worksheet to be sure it is consistent with the student objective. If it is not consistent, revise one or the other as needed.

3. Write a generality support that includes classification learning aids.

   (See segment 3.2.2 for general guidelines on classification-level generality supports. See segment 3.2.3 for guidelines and examples relative to classification-level learning aids.)

4. Collect an instance pool that meets the example/non-example specification on your Instance Worksheet.

   (See segment 3.2.4 for guidelines and examples.)

5. Drawing from the instance pool, prepare a set of classification examples and non-examples.

   (See segment 3.2.5 for guidelines and examples.)

6. Add attribute isolation to the examples and non-examples.

   (See segment 3.2.6 for guidelines and examples.)

7. Drawing from the instance pool, prepare a set of classification-level practice items that meets your specifications.

   (See segment 3.2.7 for guidelines and examples.)

Prepare a test for your classification-level instruction as follows:

1. Select the test format or formats from among the following:

   - True/False
   - Matching
   - Multiple Choice

Do not use more than two or three formats for a single test.

(See segment 3.3.1 for descriptions of the test formats.)

2. With reference to the example/non-example specification on the Instance Worksheet, draw from the pool of instances an example/non-example see that contains a full range of divergent examples, and a non-example matched to each, with a reasonably balanced distribution of representation forms among the example/non-example types. Select one complete set of test instances as your test.

3. Decide which instances you will test with each of the test formats you selected. Rewrite the instances as test items according to the guidelines for the various test formats.

   (See segment 3.3.2 for guidelines and examples.)

4. For each example and non-example that you adapted for your test, prepare attribute isolation to be used as test feedback.

5. Sequence the test items so that each instance is unmatched and divergent from previous and subsequent items.

Memory-level instruction. Plan the generality and instances for memory-level instruction as follows:

1. Write on the Generality Worksheet the exact material that the student is to memorize. (See segment 4.1.1 for guidelines and examples.)

2. Determine if there are any alternate ways of representing the information to be memorized. (See segment 4.1.2)

3. If there are alternate methods of representation, decide which ones you will require of the student and list each separately on the Instance Worksheet.

4. Specify instances on the Instance Worksheet by marking one representation for each instance. (See segment 4.1 or guidelines and examples.)

Based on your plan, prepare your memory-level instructional segment to include an objective that presents the learning activity to the student, a generality that presents the information the student is to memorize, and a generality support that presents any other interesting or helpful information related to the topic. You may also include memory-level practice.

1. Write a student objective that will direct the student in his learning activities. (See segment 4.2.1 for guidelines and directions.)

2. Review the generality as you wrote it on the Generality Worksheet to be sure it is consistent with the student objective. If it is not consistent, revise one or the other as needed.

3. Write a generality support that includes memory aids. (See segment 4.2.2 for general guidelines and examples of generality supports. See segment 4.2.3 for guidelines and examples pertaining to memory aids.)

4. Prepare memory-level practice if you judge it appropriate. (See segment 4.2.4 for guidelines and examples.)

Prepare a test for your memory-level instruction as follows:

1. Decide whether to test the entire body of material memorized or just samples.

2. Select the test format or formats from among the following:

   - True/False
   - Matching
   - Multiple choice

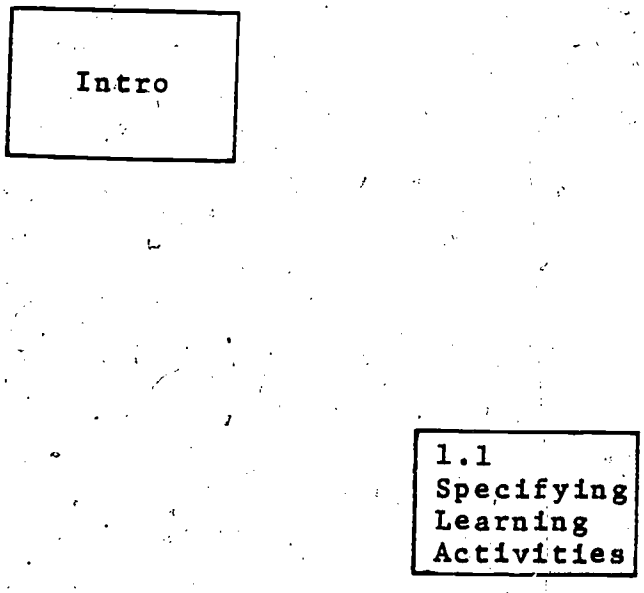
- o Completion or fill-in
- o Short Essay

Do not use more than two or three formats in a single test. (See segment 4.3.1 for descriptions of the test formats.)

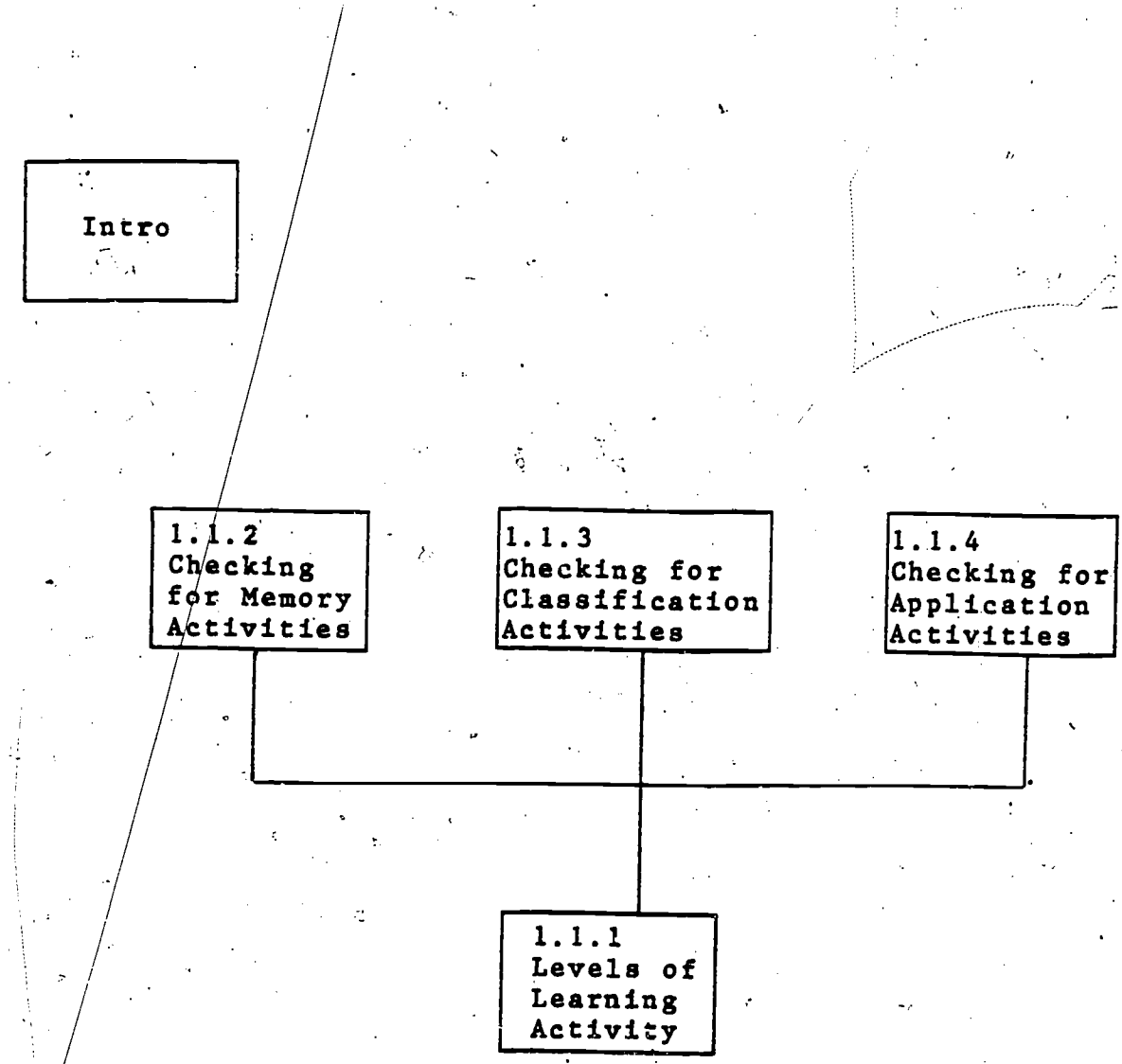
3. Prepare test items according to the format you've selected. Check your work. (See segment 4.3.2 for checking the adequacy of test items prepared according to the various formats.)

4. Proof-read carefully for typographical errors.

5. No item should contain a clue to any other item.



Unit 1 Map



Lesson 1.1 Map

1.1 Intro Specifying learning activities

Learning activity: Choose a topic and write a statement or several statements describing appropriate learning activities for teaching the topic.

1. Review your course and select a topic for development that:

   - o relates to an essential course outcome, and.....
   - o seems to give the students trouble.

2. Students will not learn a topic unless they actively work with it in some way. Specify meaningful student learning activities for your topic at the application, classification, or memory levels. (See segment 1.1.1 for the difference between the three levels.)

   - o CHECK FOR APPLICATION-LEVEL LEARNING ACTIVITIES FIRST. (See segment 1.1.2 for guidelines and examples.)

     Write any application-level learning activities you have identified on the Learning Activity Worksheet and check the "Application" column.

     (Note: A topic can be too broad to let you think easily about the problems you may face when developing instruction. If your topic seems too broad -- if you feel a bit overwhelmed as you try to specify learning activities for your topic -- break it into subtopics and consider each one separately.)

   - o CHECK FOR CLASSIFICATION-LEVEL LEARNING ACTIVITIES NEXT. (See segment 1.1.3 for guidelines and examples.)

     Write any classification-level learning activities you have identified on the Learning Acitivity Worksheet and check the "Classification" column.

   - o CHECK LAST FOR MEMORY-LEVEL LEARNING ACTIVITIES. (See segment 1.1.4 for guidelines and examples.)

     Write any memory-level learning activities you have identified on the Learning Activity Worksheet and check the "Memory" column.

1.1.1 Levels of learning activity

Learning activity: Given a statement of learning activity, identify whether the activity is at application, classification, or memory level.

Generality:

A learning activity is at the application level if:

- o An aspect of the topic is presented as a clearly defined outcome or end product of a set of activities or procedures.

  AND....

- o The student is asked to produce the outcome, or a representation of the outcome, by an actual or simulated performance of the activities by which such outcomes are produced.

A learning activity is at the classification level if:

- o An aspect of the topic is presented as a concept, class, or category to which instances or examples may be selected or produced and then presented to the students.

  AND....

- o The student is asked to classify, identify, or otherwise make known whether or not an object, event, or symbol, either real or represented, is an example or member of the concept class.

A learning activity is at the memory level if:

- o An aspect of the topic is presented that is expressed as a body of information that is fixed or invariant; that is, although the information may be paraphrased or represented in different ways, the basic information conveyed by all representations is the same.

  AND....

- o The student is asked to reproduce that basic invariant information from memory, or he is asked to recognize representations of the basic information when they are presented to him.

1.1.2 Checking for application-level learning activities

Learning activity: Given a topic, specify and appropriate application level learning activities.

~~Generality~~:

1. Consider your topic and ask yourself: "What do I want the student to be able to do when he has finished learning the material covered under this topic?"

   Do you want the student to be able to produce some outcome by applying the information he will learn? If the answer is yes, clearly specify what that outcome will be.

2. Write a statement that describes a learning activity in which the student is required to produce the outcome you've defined.

3. Once you've clearly framed the statement, judge its appropriateness according to the following criteria:

   - A student generally cannot produce the outcome well, if at all, prior to receiving the instruction.
   - The student has the time and either has or can be given the materials that he needs to perform the learning activity.
   - The learning activity contributes to the student's subsequent performance needs, either within the class or outside of it.
   - It is your judgment as the instructor that the activity "fits" -- that is, that it is consistent with your expectations and will be of real benefit to the student.

If your learning activity doesn't meet these criteria, especially the last, either revise it so that it does, or drop it.

1.1.3 Checking for classification-level learning activities.

Learning activity: Given a topic, specify any appropriate classification-level learning activities.

Guidelines:

1. Consider your topic and try to identify key concepts, classes, or categories to which a class or concept name may be assigned and from which you may select examples or instances to present to the students.

   (Note: If you've written an application activity for your topic, you already have a ready-made classification-level learning activity. Classes to be differentiated: situations where it is appropriate or essential to produce the outcome specified in the application-level learning activity, and situations where it is not.)

2. If you have identified such a key concept or class, write a statement that describes a learning activity in which the student classifies instances as either examples or non-examples of the concept, class, or category.

3. Once you've clearly framed the statement, judge its appropriateness according to the following criteria:

   - o A student generally cannot perform the classification prior to receiving the instruction.
   - o The learning activity contributes to the student's subsequent performance needs, either within the class or outside of it.
   - o It is your judgment as the instructor that the activity "fits" -- that is, that it is consistent with your expectations and will be of real benefit to the student.

   If your learning activity doesn't meet these criteria, especially the last, either revise it so that it does, or drop it.

1.1.4 Checking for memory-level learning activities.

Learning activity: Given a topic, specify any appropriate memory-level learning activities.

Guidelines:

1. Consider your topic and try to identify some aspect of it that is a body of information that is fixed and invariant. To be fixed or invariant means that, although the information may be paraphrased or represented in different ways, the basic information conveyed by all representations is the same.

   (Note 1: If you've written an application-level learning activity for your topic, you already have a ready-made memory level learning activity. Information to be memorized: Procedures, formulas, or guidelines for producing the specified outcome.)

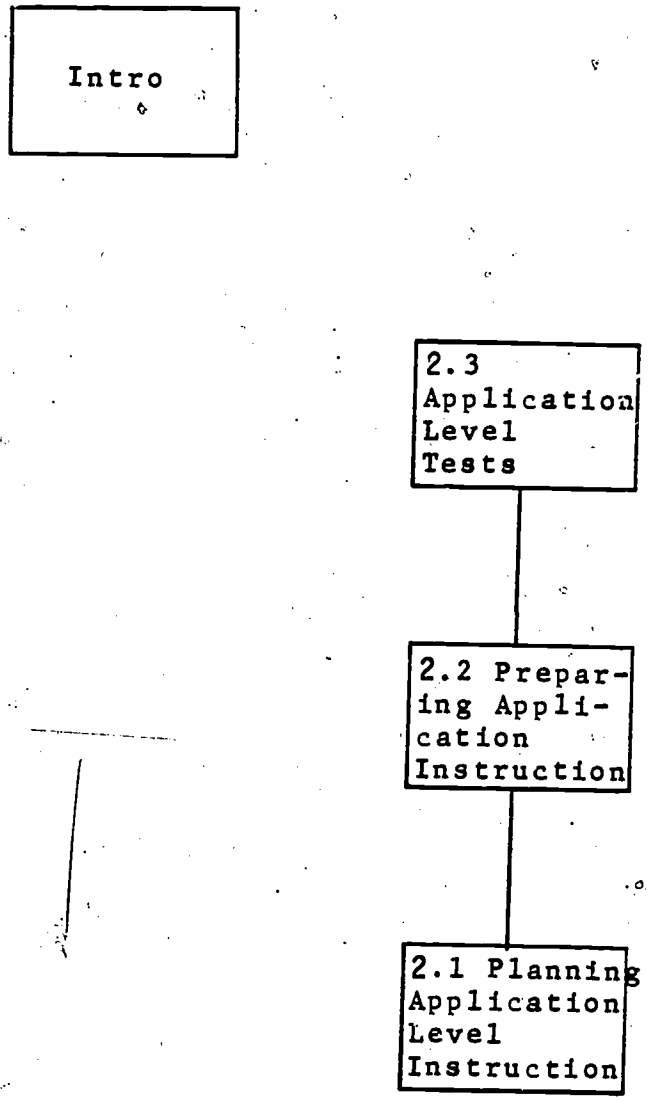
   (Note 2: If you've written a classification-level learning activity for your topic, you also have a ready-made memory-level learning activity. Information to be memorized: Attributes that characterize the concept, class, or category you've specified.

2. Write a statement that describes a learning activity in which the student reproduces the basic information from memory, or in which he recognizes representations of the basic information when they are presented to him.

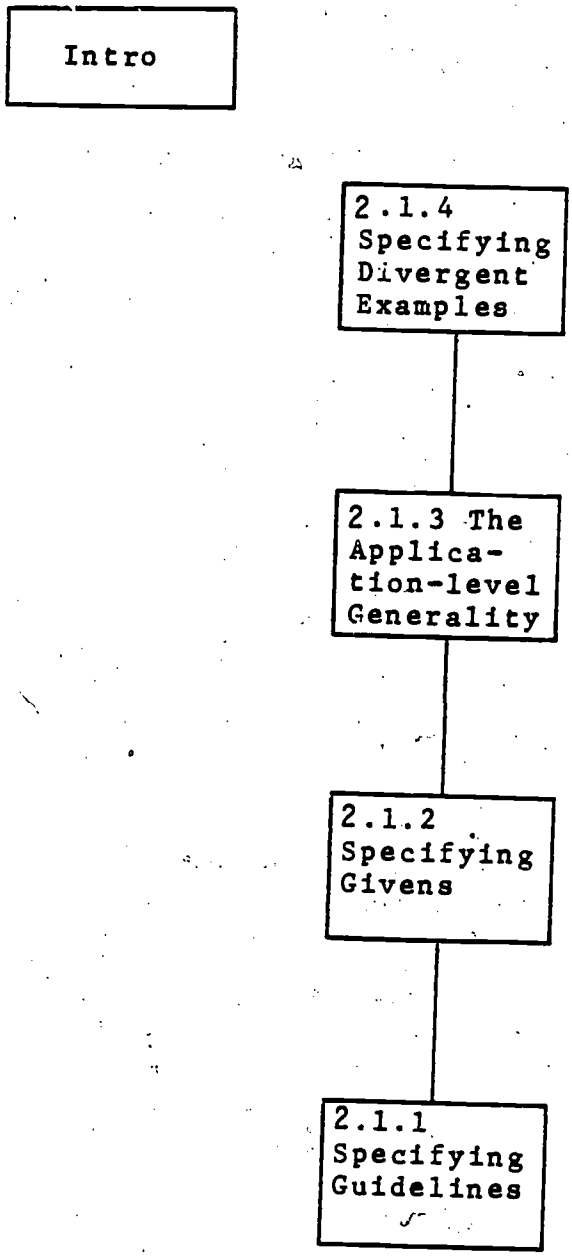
3. Once you've clearly framed the statement, judge its appropriateness according to the following criteria:

   - A student generally cannot reproduce or recognize the information prior to receiving the instruction.
   - The learning activity contributes to the student's performance needs, either within the class or outside of it.
   - It is your judgment as the instructor that the activity "fits" -- that is, that it is consistent with your expectations and will be of real benefit to the students.

If your learning activity doesn't meet these criteria, especially the last, either revise it so that it does, or drop it.


Intro
2.3 Application Level Tests
2.2 Preparing Application Instruction
2.1 Planning Application Level Instruction

Unit 2 Map



Lesson 2.1.Map

2.1 Intro Planning application-level instruction

Learning activity: Given a topic to be taught at application level, plan the generality and the instances for an instructional segment.

GENERALITY:

You may plan the generality and instances for a segment of application-level instruction by following these guidelines:

1. List any guidelines, general or specific, that will direct the student's activities as he produces the outcome specified in the learning activity.

   (See segment 2.1.1 for guidelines and examples.)

2. Review the guidelines and mentally work through the process the student is to learn. Identify all of the givens and list them.

   (See segment 2.1.2 for guidelines and examples.)

3. List the relevant variations, if any, for each given.

   (See segment 2.1.2 for guidelines and examples.)

4. Review the relevant variations for each given and cross out the ones that you do not intend to teach.

5. List the givens on an Instance Worksheet.

   - List first all of the fixed givens, that is, those that have no relevant variations that you intend to teach.
   - List next the variable givens, that is, those that do have relevant variations that you intend to teach.
   - For each variable given, list its relevant variations and number them for easy reference.

6. With reference to the guidelines you have written and the list of givens on the Instance Worksheet, write an application generality on the Generality Worksheet.

   (See segment 2.1.3 for guidelines and examples.)

7. Specify a set of divergent examples on the Instance Worksheet.

   (See segment 2.1.4 for guidelines and examples.)

2.1.1 Specifying guidelines

Learning activity: Given an outcome, specify information that can guide a student's productive activities.

Generality:

As you prepare guidelines for the student:

- List any information that can direct his outcome-oriented activities to make them more productive.
- Be as specific as the subject matter and common sense will allow, although your guidelines may range from step-by-step directions to suggestions and hints.
- Include only the guidelines, no examples.
- Write the guidelines in active voice and in second person, as you would speak to the student.

2.1.2 Specifying givens

Learning activity: Given guidelines for producing an outcome, specify the givens and any relevant variations.

Generality:

To find the <u>givens</u>, review the process in your mind and identify anything which:

- must be present if the outcome is to be produced.
- is not produced as part of the process.
- is not the person carrying out the process, nor one of his natural possessions or attributes.
- is not a general requirement for all functioning.

To find the <u>relevant variations</u> of the givens you have identified, review each given and ask yourself if it comes in different kinds or varieties which:

- require changes in process.

OR.....

- make the process either harder or easier by introducing or eliminating possible errors.

2.1.3 Writing the application-level generality

Learning activity: Given an outcome to be produced, a set of guidelines, and a set of givens with their relevant variations, if any, prepare an application-level generality.

Generality:

Write your application-level generality so that it:

- clearly sets forth the outcome to be produced.
- lists the givens and their relevant variations, if any.
- shows the changes in the process required by any relevant variations of the givens.
- contains no examples or nice-to-know material.
- is consistent with the student learning activity.

2.1.4 Specifying divergent application-level examples.

Learning activity: Given a list of givens and their relevant variations on an Instance Worksheet, specify a set of divergent examples.

Generality:

These guidelines should help you specify a complete set of divergent application examples based on the list of givens with their relevant variations that you wrote on the Instance Worksheet. Specify the examples in the columns on the right side of the sheet, one example per column.

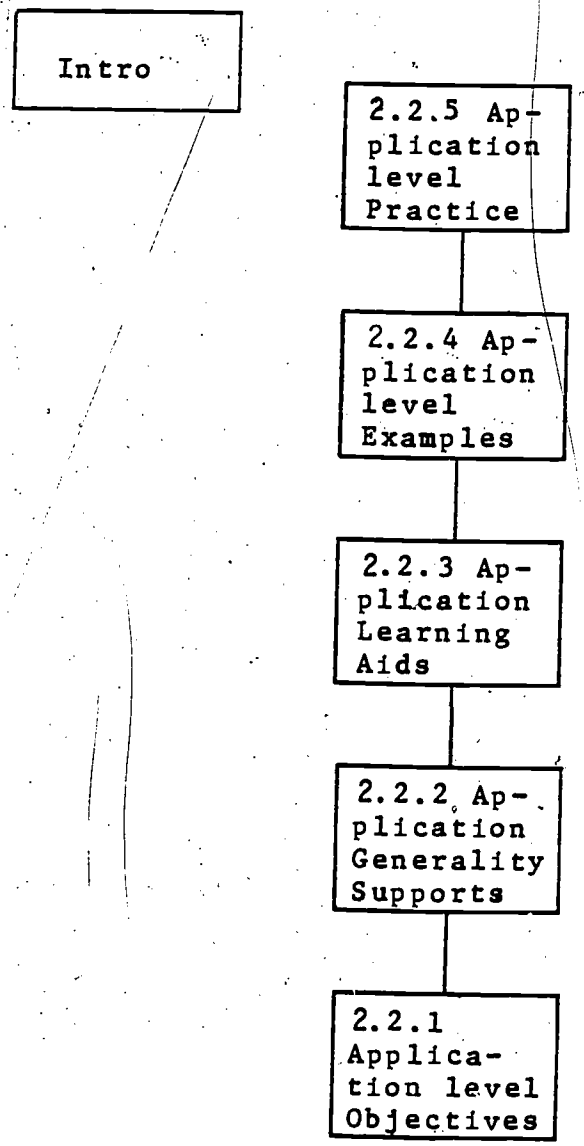
1. You specify the example by marking the givens that will characterize it. For the first example, place an "X" or a check (✓) in the column for each fixed given. Indicate with a reference number one of the variations for each variable given. You have now specified the first example.

2. Specify subsequent examples by marking, as before, all of the fixed givens and one of the variations for each variable given. Make each new example as different from the previous examples as possible.

3. Stop when you have checked at least once each relevant variation of all of the variable givens.

4. The set of examples is now complete in that each variation within the set has occurred at least once. However, other combinations of variations are still possible. Review the set of examples and specify other combinations of relevant variations of the givens if doing so will make the set seem more complete to you.

5. On the Instance Worksheet, specify for each example type the number of items you will collect or prepare for examples, for practice, for tests, and for replacement of poor items or preparation of alternate versions of the test. We recommend one for each category, or possibly two or three in the last category if you plan to prepare alternate versions of the test. However, as a general rule, the less clearly you can specify guidelines that will direct the student, the more examples and practice items you will need.



Lesson 2.2 Map

2.2 Intro Preparing application-level instruction

Learning activity: Write a manuscript for a segment of application-level instruction that contains all necessary instructional components, with the components clearly separated from, and consistent with, one another.

Generality:

Application-level instruction generally included an objective that presents the learning activity to the student, a generality that presents the instruction that the student will apply in order to produce the outcome specified in the objective, a generality support that presents other helpful or interesting information, examples that demonstrate the production of the outcome according to the information presented in the generality, and practice that requires that the student produce the outcome or a representation of the outcome.

To prepare application-level instruction based on your instructional specification:

1. Write a student objective that will direct the student in his learning activities.

   (See segment 2.2.1 for guidelines and examples.)

2. Review the generality as you wrote it on the Generality Worksheet to be sure it is consistent with the objective. If it is not consistent, revise one or the other as needed.

3. Write a generality support that includes an application-level learning aid.

   (See segment 2.2.2 for general guidelines and examples for generality supports. See segment 4.2.3 for guidelines and examples relative to application-level learning aids.)

4. Prepare a set of divergent examples.

   (See segment 2.2.4 for guidelines and examples.)

5. Prepare a set of application-level practice problems.

   (See segment 2.2.5.)

2.2.1 Application-level student objectives

Learning activity: Given specifications for a segment of application-level instruction, write an application-level student objective.

Generality:

With reference to your application-level instructional specifications, write your student objective so that it:

- tells the student unambiguously the outcome he is to produce, including any special features of the format in which he will present the final outcome.
- presents any non-obvious aspects of the conditions under which he will produce the outcome, including the givens with which he will work.
- tells the standards of mastery you expect him to achieve.

2.2.2 Application-level generality supports.

Learning activity: Given instructional specifications for a segment of application-level instruction and supplemental information about the topic, write an application-level generality support.

Generality:

You may use the generality support to present to the student any information you feel will be of interest or of help to him. Generality supports may contain information such as the following:

- Definition of unfamiliar terms and concepts.
- Warning about potential errors, especially these introduced by variations in the givens.
- Information showing how the generality relates to other parts of the instruction.
- Interesting nice-to-know information or other supporting explanation that is related to the generality.
- A flow-chart or prose algorithm.

2.2.3 Application-level learning aids

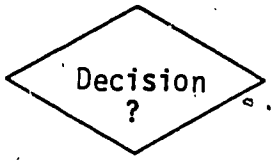
Learning activity: Given instructional specifications for a segment of application-level instruction, prepare a flow chart or, if appropriate, a prose algorithm.

Generality:

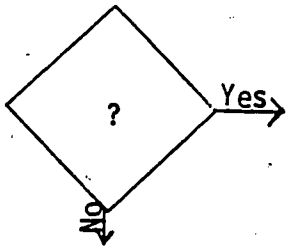
FLOW CHART

- A flow chart is a symbolic diagram outlining a student's path in applying a procedure to solve a problem or produce an outcome.
- The flow of the procedure from step to step is shown by arrows.
- Information about the procedure is written in box-like symbols which may take a variety of geometric shapes. Certain shapes generally represent certain functions. Here is a set of flow charting symbols you will probably find useful. There are books that give a complete set of flow charting conventions.

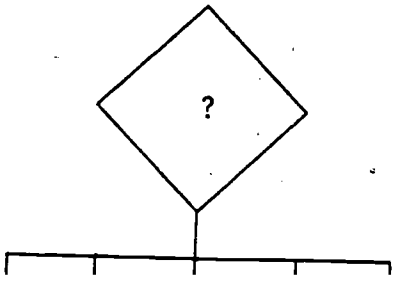
A single step in the procedure is written in a rectangular box.



Decision points which cause "branching" or changes in the pathway through the procedure are written in diamond-shaped symbols.

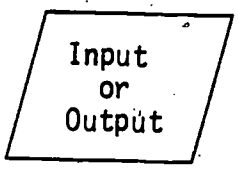


Decision points leading to two or three branches may use two or three of the free points of the diamond as branch-origins.

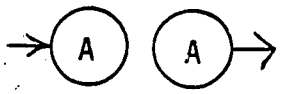


Here is a way of representing multiple branches.

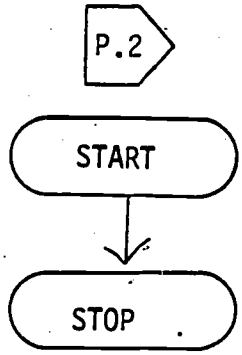
2.2.3 Application-level learning aids (page 2)



Inputs to, or outputs from, the procedure are written in a parallelogram.



A letter in a circle is used as an on-page connector when it is cumbersome to show the flow with an arrow. The same letter is used both where the arrow breaks off and where it begins again.



A "home-base" shaped symbol is used as an off-page connector. A number in the symbol shows the page where the flow continues.

An oval-shaped symbol is used to show where the procedure starts and stops.

You can buy a flow chart template to help you draw the symbols.

4. You can use flow charts to show "loops" in a procedure by turning the arrows back to an earlier part of the procedure. Make sure you make some provision for leaving the loop.

PROSE ALGORITHM

- A prose algorithm is a verbal statement of the steps to be followed in applying a procedure.
- Prose algorithms are useful when the process is represented in the generality as a symbolic formula.
- Write the prose algorithm in active voice and second person as you would talk to the student.

2.2.4 Preparing application-level examples.

Learning activity: Given a set of specifications on an Instance Worksheet for application instruction, prepare a set of divergent application examples.

Generality:

Prepare application examples as follows:

1. Decide how you will represent the examples to the students.

IF YOUR EXAMPLES WILL BE PRESENTED IN A LIVE PERFORMANCE BEFORE THE CLASS, DO THE FOLLOWING FOR EACH EXAMPLE YOU HAVE SPECIFIED ON YOUR INSTANCE WORKSHEET:

2. Define an actual given for each given or relevant variation of a given that you listed on the Instance Worksheet.

3. Prepare the givens in whatever form you need them for your in-class presentation.

4. Check your preparation to be sure that you have what you need to clearly demonstrate for the students the givens, the method of producing the outcome, and the outcome itself.

IF YOUR EXAMPLES WILL BE PREPARED BEFORE HAND AND PRESENTED IN SOME REPRESENTATION FORM OTHER THAN IN-CLASS PERFORMANCE, DO THE FOLLOWING FOR EACH EXAMPLE YOU SPECIFIED ON THE INSTANCE WORKSHEET:

2. Define an actual given for each given or relevant variation of a given that you listed on the Instance Worksheet.

3. Using those actual givens, work through the example, producing the outcome according to the guidelines.

   - Do not take shortcuts that would not be available to the students with their more limited knowledge.
   - Record all of your interim work in a way that is compatible with the way you decided to represent the example to the students.

4. Prepare the example in the representation form you selected showing the givens, your work through the process, and the outcome you produced.

2.2.5 Application-level practice

Learning activity: Given a set of specifications on an Instance Worksheet for application instruction, prepare a set of instructionally sound application-level practice items.

Generality:

You may prepare a set of practice items for application-level instruction as follows:

1. Decide on the format or formats you will use to present the practice items to the students. You may use any of the test formats for application instruction that are presented in lesson 4.3.

2. For each example type you specified on the Instance Worksheet, assign <u>actual</u> givens to the givens or relevant variations of givens that you listed on the Instance Worksheet.

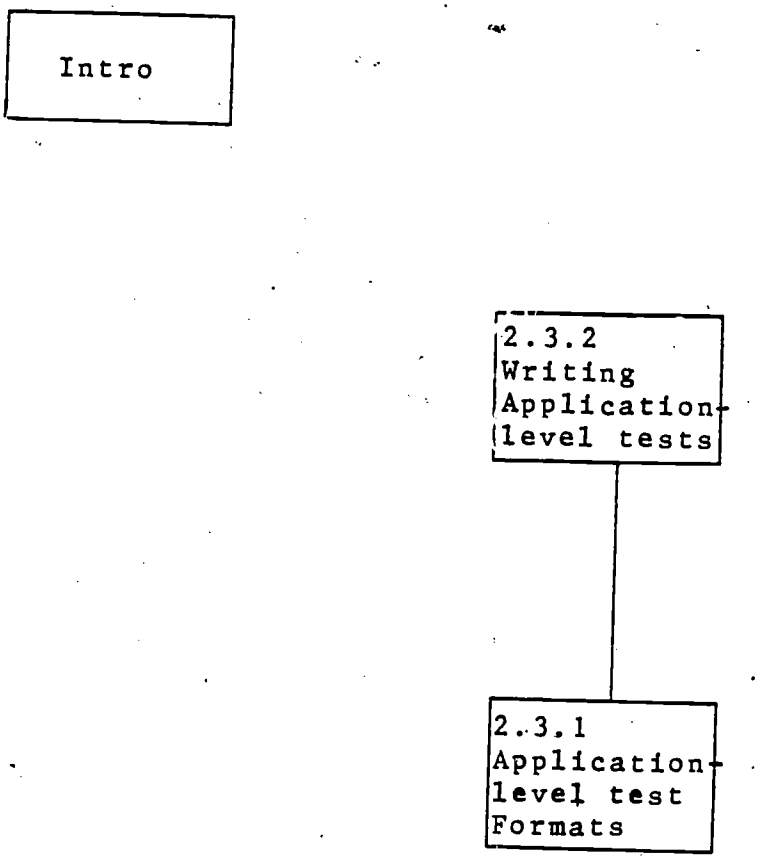
3. Produce the givens in a form that is compatible with the format you selected for presenting the practice items.

4. Work through each item to produce not only the solution, but also likely wrong answers. Use the right answer for feedback. The likely wrong answers are good distractors (incorrect response options) for objective practice formats.

5. Prepare the practice items according to the formats selected. (See segment 2.3.2) Make sure you have given the student everything he needs to produce the outcome.

6. If practice is written, prepare a feedback for each item. Present the feedback on the back of the practice page. Repeat the practice item on the feedback page.

7. Sequence the items so that the harder ones come last.



Lesson 2.3 Map

2.3 Intro Preparing application-level test

Learning Activity: Given instructional specifications for a segment of application-level instruction, prepare an application-level test.

Generality:

Here are some guidelines for preparing application-level tests:

1. Select the test format or formats from among the following:

   - o True/False
   - o Matching
   - o Multiple Choice
   - o Completion/Fill-in
   - o Short Essay
   - o Simulation
   - o Actual Performance

   Do not use more than two or three formats in a single test.

   (See segment 2.3.1 for descriptions of the test formats.)

2. For each example type you specified on the Instance Worksheet assign an actual given for each given or relevant variation of a given that you listed on the Instance Worksheet.

3. Produce the givens in a form that is compatible with the format or formats you selected for presenting the test items.

4. Work through each item to produce not only the solution, but also likely wrong answers. Use the right answers for test feedback; the likely wrong answers are good distractors (incorrect response options) for objective test formats.

5. Prepare test items according to the format selected.

   (See segment 2.3.2 for guidelines and examples.)

2.3.1 Application-level test formats

Learning activity: List and describe the test formats that may be used for testing application-level instruction.

Generality:

OBJECTIVF FORMATS FOR TESTING APPLICATION-LEVEL INSTRUCTION:

- o include True/False, Matching, and Multiple Choice items that ask the student to work through a problem in order to identify response options that present the correct answer.
- o include Completion or Fill-in items that ask the student to work a problem and write the answer.
- o are easy to respond to and therefore may often cover a broader range of problems in a single test than is possible with non-objective formats.
- o may require that you write many items for a single test.
- o may be scored quickly, accurately, and consistently.

ESSAY FORMATS FOR TESTING APPLICATION-LEVEL INSTRUCTION:

- o present a problem to the student and ask that he not only provide the answer, but also describe the processes by which he arrived at the answer.
- o let you cover only a few problems in a single test.
- o may favor fluent and articulate students, although they do so less than when testing instruction at the memory and classification levels.
- o take longer to grade than objective items, and they may be difficult to score accurately and consistently.

PERFORMANCE FORMATS FOR TESTING APPLICATION-LEVEL INSTRUCTION:

- o let the teacher observe not only the outcome of the student's activities, but also productive activities themselves.
- o include simulations in which the student engages in an activity which imitates key features of the process he will eventually be required to perform.
- o may be expensive in both time and materials, although simulations may be the only economical way to test certain performance-oriented instruction. In other situations, long-range economics may demand validation of learning through actual performance, regardless of cost.
- o are especially useful in testing activities involving psychomotor or interpersonal skills.

- may be difficult to grade accurately and consistently, depending upon how well you can observe specific features of the process and the outcome.

2.3.2 Writing application-level test items.

Learning activity: Given instructional specifications for a segment of application-level instruction, prepare test items in each of the test formats.

Generality:

TRUE/FALSE

Write a True/False item for testing application-level instruction so that it:

- o clearly presents all necessary givens and the exact nature of the outcome to be produced.
- o presents True/False statements that can be interpreted in only one way.
- o if false, is an outcome that would be produced by a likely error.
- o is easy to read.
- o is consistent with the student objective.

MATCHING

Write a Matching item for testing application-level instruction so that it:

- o clearly presents all necessary givens and the exact nature of the outcome to be produced.
- o presents response options that can be interpreted in only one way.
- o presents all problems in one column and all answers in the other.
- o has more responses than problems, or allows a single response to answer more than one problem.
- o includes among the distractors answers that would be produced by likely errors.
- o is easy to read.
- o is consistent with the student objective.

MULTIPLE CHOICE

Write a Multiple Choice item for testing application-level instruction so that it:

- clearly presents all givens and the exact nature of the outcome to be produced.
- presents a single problem in the stem.
- presents response options that can be interpreted in only one way.
- uses distractors, or incorrect response options, that would be produced by a likely error.
- uses "All of the above" as a response option only when other combinations of distractors are also presented as choices.
- is easy to read.
- is consistent with the student objective.

COMPLETION or FILL-IN

Write a Completion or Fill-in item for testing application-level instruction so that it:

- clearly presents all givens and the exact nature of the outcome to be produced.
- is easy to read.
- is consistent with the student objective.

ESSAY

Write an Essay item for testing application-level instruction so that it:

- clearly presents all givens and the exact nature of the outcome to be produced.
- tells the student that he must not only produce the specified outcome, but also describe the process by which he arrived at the outcome.
- is easy to read.
- is consistent with the student objective.

SIMULATION

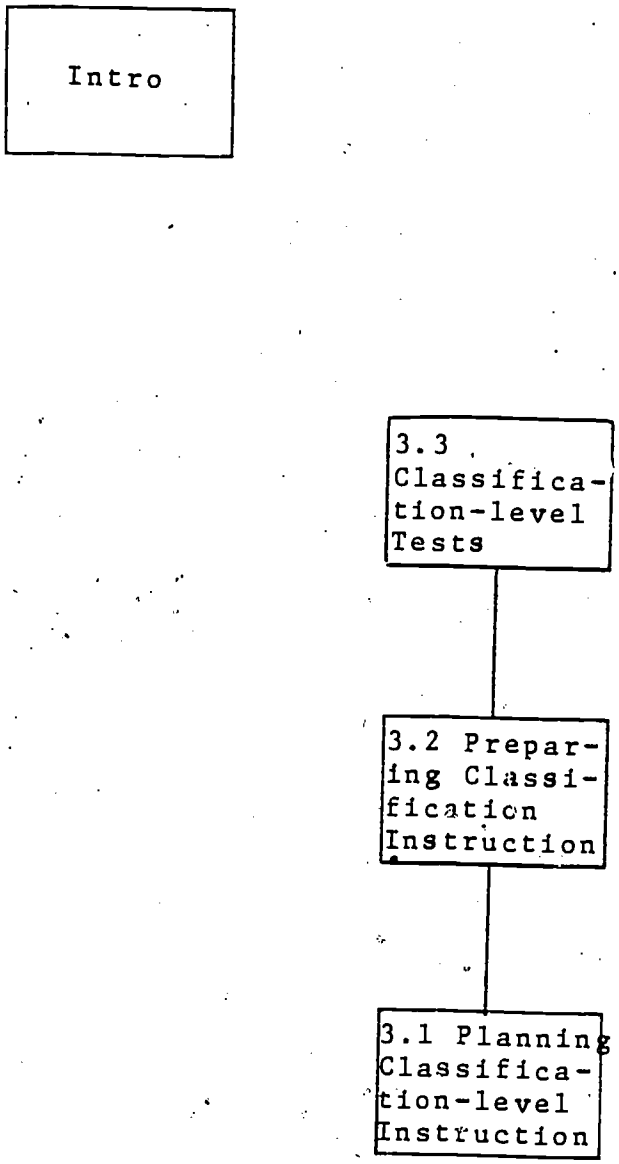
Write a Simulation item for testing application-level instruction so that it:

- o clearly presents all necessary givens and the exact nature of the outcome to be produced.
- o is enough like the actual performance the student will eventually be required to make to let the teacher make valid inferences about the student's performance in the actual situation.
- o allows clear observation by the teacher of the essesntial features of the simulated performance.
- o is consistent with the student objective.

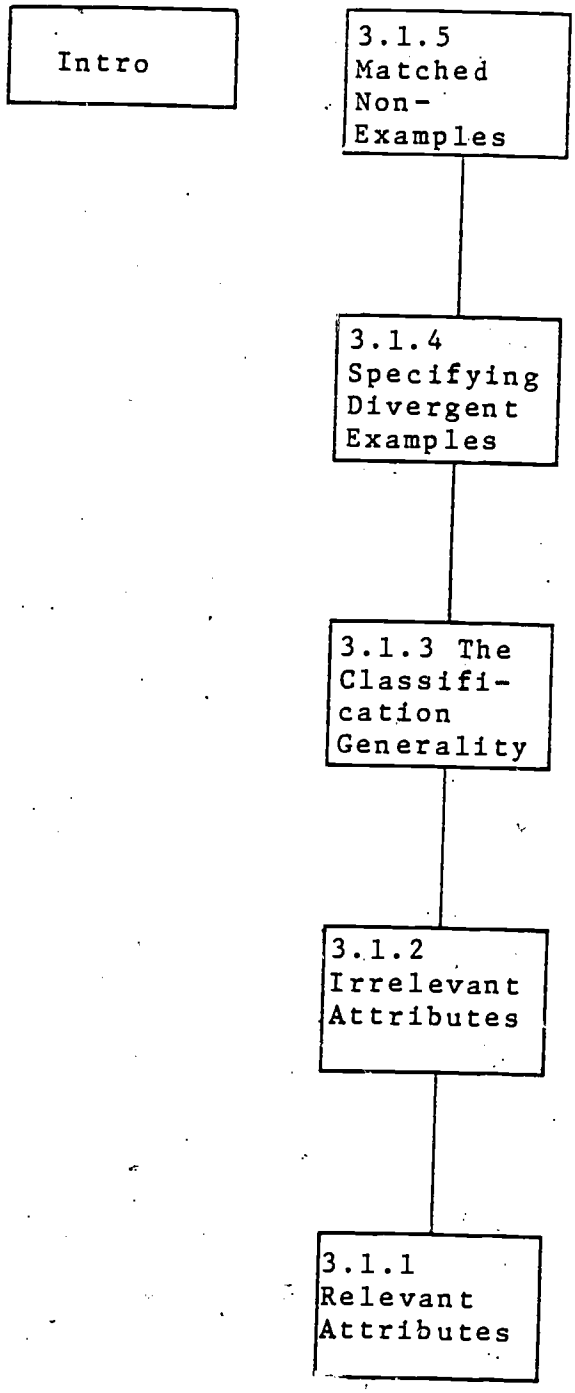
ACTUAL PERFORMANCE

Write an Actual Performance item for testing application-level instruction so that it:

- o clearly presents all necessary givens and the exact nature of the outcome to be produced.
- o allows clear observation by the teacher of the essential features of the performance.
- o is consistent with the student objective.


Intro
3.3 Classification-level Tests
3.2 Preparing Classification Instruction
3.1 Planning Classification-level Instruction

Unit 3 Map



Lesson 3.1 Map

3.1 Intro Planning classification-level instruction

Learning activity: Given a topic to be taught at classification level, plan an instructional segment.

Generality:

1. Make a list of the relevant attributes of the concept.

   - o Divide the list into two parts: universal relevant attributes and non-universal relevant attributes.
   - o Review the list of non-universal relevant attributes and identify groups of attributes that go together to define specific subclasses of the concept.
   - o Rewrite the list of non-universal relevant attributes to show each subclass separately. The subclasses, taken together, must cover the entire concept class.

   (See segment 3.1.1 for guidelines and examples.)

2. Make a list of the useful irrelevant attributes.

   - o Each useful irrelevant attribute may assume more than one state. For each useful irrelevant attribute, list the separate states it may assume and number them for reference. The set of states for an irrelevant attribute must cover the entire concept class.

   (See segment 3.1.2 for guidelines and examples.)

3. Write the attributes you've defined on the Instance Worksheet.

   - o List separately each universal relevant attribute.
   - o List the non-universal relevant attributes grouped by subclass.
   - o List separately each useful irrelevant attribute, showing for each the numbered set of states it may assume.

4. Based on your list of attributes of the concept class, write a classification generality on the Generality Worksheet. Write it as you intend to present it to the students.

   (See segment 3.1.3 for guidelines and examples.)

5. Review the generality and underline with a red pencil, or in some other noticeable way, any terms or concepts that may be unfamiliar to the students.

6. Specify a set of divergent examples on the instance Worksheet.

   (See segment 3.1.4 for guidelines and examples.)

7. Specify a set of matched non-examples on the Instance Worksheet. (See segment 3.1.5 for guidelines and examples.)

3.1.1 Relevant attributes

Learning activity: Given a definition of a concept class, with supporting explanation and examples, list the universal and the non-universal relevant attributes, Group the non-universal relevant attributes by subclass.

Generality:

An attribute of a concept class is relevant if:

- its presence must be observed in order to correctly classify at least one member of the concept class.

A relevant attribute is universal if:

- it characterizes all members of the concept class.

A relevant attribute is non-universal if:

- it characterizes some, but not all members of the concepts class.
- either alone, or in combination with other non-universal relevant attributes, it defines a specific subclass within the concept class.

3.1.2 Irrelevant attributes

Learning activity: Given a definition of a concept class, list the useful irrelevant attributes, showing the set of states each may assume within the concept class.

An attribute of a concept class is irrelevant if:

- observation of its presence does not make an essential contribution to the correct classification of any of the members of the concept class, even though it may characterize any or all of them.

An irrelevant attribute is useful if:

- there tends to be a mental association between the irrelevant attribute and the concept class that creates the possibility that the student may think that the irrelevant attribute is relevant.

  OR.....

- there are two or more states of the attribute that introduce noticeable variety into an example set.

3.1.3 Writing the classification generality.

Learning activity: Given a list of relevant attributes of a concept class, write them in a generality in the way you would present them to students.

Generality:

Write your generality so that it:

- o presents the concept name.
- o presents all universal relevant attributes.
- o presents all non-universal relevant attributes and clearly shows the complete set of subclasses within the concept class.
- o is consistent with the student learning activity.
- o contains no nice-to-know material.

3.1.4 Specifying divergent examples.

Learning activity: Given a statement of relevant and irrelevant attributes on an Instance Worksheet, specify a set of divergent examples.

Generality:

These guidelines should help you specify a complete set of divergent examples based on the list of relevant and irrelevant attributes you wrote on the Instance Worksheet. Specify the examples in the columns on the right side of the sheet, one example per column.

1. You specify an example by marking the attributes that are to characterize it. For the first example, place an "X" or a check (✓) in the column for <u>each</u> universal relevant attribute and for <u>one</u> of the non-universal relevant attribute sets (each set defines a subclass). Indicate with a reference number <u>one</u> <u>state</u> for each of the useful irrelevant attributes. You have now specified the first example.

2. Specify subsequent examples by marking, as before, all of the universal relevant attributes, one of the non-universal relevant attribute sets, and one state for each of the useful irrelevant attributes. However, make each new example as different from the previous examples as possible.

3. Stop when you have checked <u>at least</u> twice each non-universal relevant attribute set and each state of each useful irrelevant attribute.

4. The set of examples is now reasonably complete since the student will see each variation within the set at least twice. However, other combinations of variations may still be possible. Review the set of examples and specify other combinations of attribute variations if doing so will make the set seem more complete to you.

5. On the Instance Worksheet, specify for each example type the number of items you will collect or prepare for examples, practice, tests, and for replacement of poor items or preparation of alternate versions of the test. We recommend one for each category, or possibly two or three in the last category if you plan to prepare alternate versions of the test.

3.1.5 Specifying matched non-examples.

Learning activity: Given the specifications for a set of divergent examples on an Instance Worksheet, specify a set of non-examples matched to the example set.

Generality:

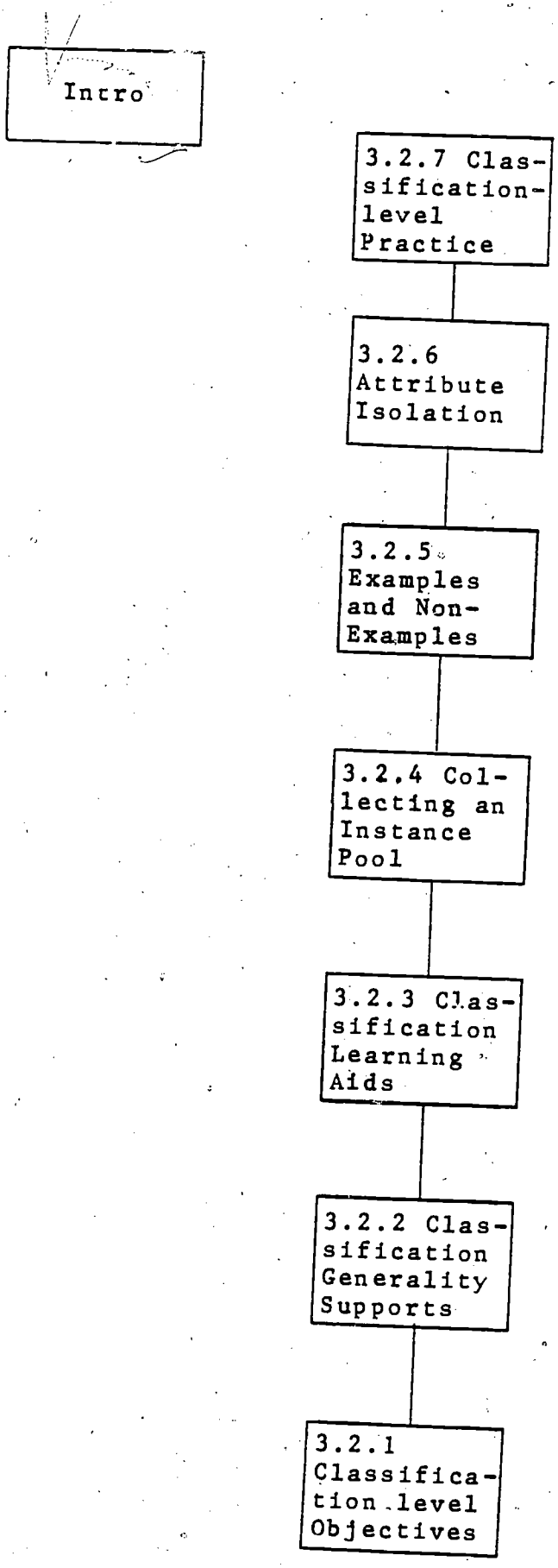
A non-example does not belong to the concept class because it is missing one or more of the relevant attributes. A non-example is matched to an example when it is identical to the example in everything except the missing relevant attribute or attributes. For each example, specify a matched non-example as follows:

1. Consider each relevant attribute in turn and decide whether an instance prepared without that relevant attribute would be meaningful.

2. If none of the non-examples based on a single missing relevant attribute seems meaningful to you, consider non-examples that are missing a combination of relevant attributes.

3. When you have determined which critical attribute or attributes you will delete to form your matched non-example, circle the corresponding marks in the example column of the Instance Worksheet. Use a red pencil or some other noticeable marking if possible.

4. Vary the missing relevant attribute from non-example to non-example.



Lesson 3.2 Map

3.2 Intro Preparing classification-level instruction

Learning activity: Write a manuscript for a segment of classification level instruction that contains all necessary instructional components, with the components clearly separated from, and consistent with, one another.

Generality:

Classification-level instruction generally includes an objective that presents the learning activity to the student, a generality that gives the basis on which the student will make his classifications, a generality support that presents other helpful or interesting information, examples of correctly made classifications, and practice that gives the student an opportunity to make classifications based on the information in the generality.

To prepare classification-level instruction based on your instructional specifications:

1. Write a student objective that will direct the student in his learning activities.

   (See segment 3.2.1 for guidelines and examples.)

2. Review the generality as you wrote it on the Generality Worksheet to be sure it is consistent with the student objective. If it is not consistent, revise one or the other as needed.

3. Write a generality support that includes classification learning aids.

   (See segment 3.2.2 for general guidelines on classification-level generality supports. See segment 3.2.3 for guidelines and examples relative to classification-level learning aids.)

4. Collect an instance pool that meets the example/non-example specifications on your Instance Worksheet.

   (See segment 3.2.4 for guidelines and examples.)

5. Drawing from the instance pool, prepare a set of classification examples and non-examples.

   (See segment 3.2.5 for guidelines and examples.)

6. Add attribute isolation to the examples and non-examples.

   (See segment 3.2.6 for guidelines and examples.)

7. Drawing from the instance pool, prepare a set of classification-level practice items that meets your specifications.

   (See segment 3.2.7 for guidelines and examples.)

3.2.1 Classification-level student objectives

Learning activity: Given specifications for a segment of classification-level instruction, write a classification-level instruction, write a classification-level student objective.

Generality:

With reference to your classification-level instructional specifications, write your student objective so that it:

- tells the student the concept name and how he will demonstrate his learning.
- lets the student know how the examples will be presented to him and any other non-obvious aspects of the conditions under which he will perform the classification activities.
- states the standards of mastery the student will be expected to achieve.

3.2.2 Classification-level generality supports.

Learning activity: Given instructional specifications for a segment of classification-level instruction and supplemental information about the topic, write a classification-level generality support.

Generality:

You may use the generality support to present to the student any information you feel will be of interest or of help to him. Generality supports may contain information such as the following:

- o Definitions of unfamiliar terms and concepts.
- o Warning about particularly confusing useful irrelevant attributes.
- o Information showing how the generality relates to other parts of the instruction.
- o Interesting nice-to-know information or other supporting explanation that is related to the generality.
- o A classification algorithm or a classification checklist.

(See segment 3.2.3 for guidelines and examples.)

3.2.3 Classification-level learning aids.

Learning activity: Given instructional specifications for a segment of classification-level instruction, prepare a classification algorithm or a classification checklist.

Generality:

CLASSIFICATION CHECKLIST

- A classification checklist is a series of questions, each one asking the student to observe the presence or absence in an example of one of the critical attributes listed in the generality. There is one question for each attribute.
- Phrase the questions so that a positive answer consistently means that the attribute is present in the example, and a negative answer means that it is not.
- The checklist should make apparent to the student the combination or combinations of answers that will enable him to correctly identify examples as members of the concept class. In other words, all of the subclass relationships within the concept class should be clear.

CLASSIFICATION ALGORITHM

- A classification algorithm expresses step-by-step all of the discriminations the student must make in order to correctly classify examples of the concept class.
- Classification algorithms can express not only a linear series of questions, but they may also include branching and more complex decision relationships. For this reason, classification algorithms are especially helpful when there is a complex set of subclass relationships within the relevant attributes of the concept class.
- Classification algorithms may be expressed as prose algorithms or in flow chart form.

3.2.4 Collecting an instance pool

Learning activity: Given the specifications for a set of divergent examples and matched non-examples, collect an instance pool that meets the example specifications.

Generality:

You may prepare an instance pool as follows:

1. Determine the form or forms of representation you will use for presenting the instances to the students.

   - Any concept class may be represented by a variety of instance forms, ranging from verbal descriptions, through pictures and other simple models, to elaborate simulations, and finally to the actual members of the concept class. Be creative in imagining a wide variety of possible representations for instances from the concept class you are considering.

   - Select at least two ways of representing instances to your students; more if practical.

   - The representations you choose must be able to unambiguously display to the student the relevant attributes of the concept class.

2. Using the example/non-example specifications as a guide, collect instances in the representations you have chosen.

   - You may collect already existing instances from other sources, or you may construct your own.

   - As you collect each instance, refer to the example and non-example types you specified on the Instance Worksheet. Keep a running tally of the number of instances you have collected in each type. Label each instance according to its type using the letters that head the example columns on the Instance Worksheet.

   - Keep going until you have collected the number of instances you specified on the Instance Worksheet.

   - Try to maintain a reasonably balanced distribution of the representation forms among the instances specified.

   - You will find that some types and representations are hard to find or produce. Just do your best to collect as complete a set as possible.

   - The complete set of instances will be the basis from which you will develop your examples and non-examples, and your practice and test items.

3.2.5 Classification examples and non-examples

Learning activity: Given a set of example/non-example specifications and a corresponding pool of instances, assemble an instructionally sound example/non-example set.

Generality:

You may prepare an example/non-example set for classification instruction as follows:

1. With reference to the specifications on the Instance Worksheet, draw from the pool of instances an example/non-example set that contains the full range of divergent examples and a non-example matched to each, with a reasonably balanced distribution of representation forms among the example/non-example types.

2. Organize the set into pairs of examples and their matching non-examples.

3. Sequence the example/non-example pairs in a way that maximizes the example-example divergence from pair to pair.

3.2.6 Attribute isolation

Learning activity: Given a classification generality and a set of examples and non-examples, add attribute isolation to the example/non-example set.

Generality:

Here are some guidelines to help you add attribute isolation to classification examples and non-examples:

- Attribute isolation is any device that directs the student's attention to the relevant attributes in an example or non-example, thereby clarifying the reasons for the item's classification.

- Do not add attribute isolation unless you judge that it really helps the student see why the particular instance is or is not a member of the concept class being taught.

- Use attribute isolation only to help students focus on the relevant attributes. Do not use attribute isolation to direct attention to words, such as the concept name or names of attributes, unless you simultaneously direct attention to the attributes themselves.

- Useful devices for attribute isolation include:
  Color or shading
  Exploded drawings
  Arrows, underlining, brackets, or other special symbols
  Labels
  Written or audio directions
  Simplified illustrations

3.2.7 Classification-level practice

Learning activity: Given a set of example/non-example specifications and a corresponding pool of instances, assemble an instructionally sound set of classification practice items.

Generality:

You may prepare a set of practice items for classification-level instruction as follows:

1. With reference to the example/non-example specifications on the Instance Worksheet that you prepared, draw from the pool of instances an example/non-example set that contains a full range of divergent examples, and a non-example matched to each, with a reasonably balanced distribution of representation forms among the example/non-example types.

2. Re-write each example and non-example as a practice item that presents the unlabeled instance to the student and asks that he classify it as either a member or a non-member of the concept set. Use any of the testing formats for testing classification instruction.

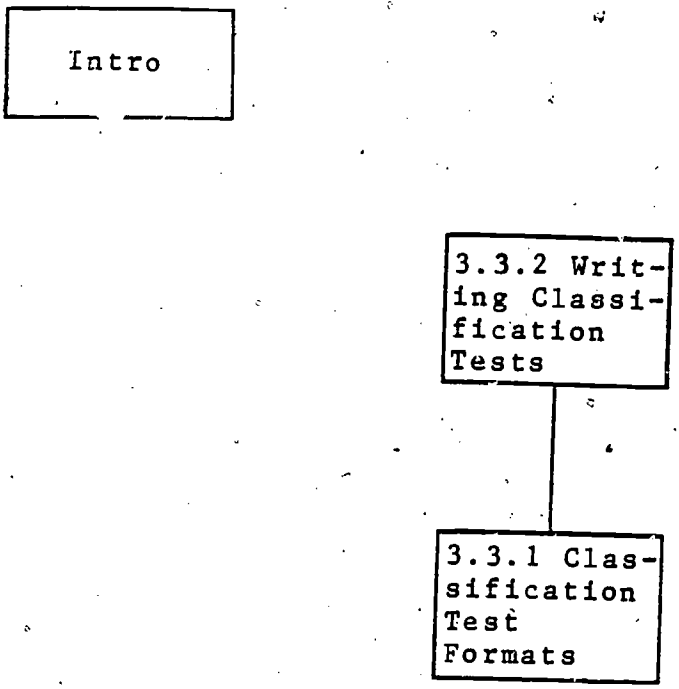
   (See segments 3.3.1 and 3.3.2 for information regarding test formats.)

3. For each example and non-example that you adapted to practice format, prepare attribute isolation to be used as practice feedback.

   (See segment 3.2.6.)

4. Pair each practice item and its attribute isolation feedback, and sequence the items according to these principles:

   - Order the items so that each instance is unmatched to and divergent from the immediately previous and subsequent items.
   - If you can judge that there are differences in the difficulty of the items, order them so that the easier items are presented first.
   - Place the practice item on the front of the page and the attribute isolation feedback on the back.



Lesson 3.3 Map

3.3 Intro Preparing classification-level tests

Learning activity: Given instructional specifications for classification-level instruction, prepare a classification-level test.

Generality:

Here are some guidelines for preparing classification-level tests:

1. Select the test format or formats from among the following:

   - True/False
   - Matching
   - Multiple Choice
   - Completion/Fill-in
   - Short Essay

   Do not use more than two or three formats for a single test.

   (See segment 3.3.1 for descriptions of the test formats.)

2. With reference to the example/non-example specifications on the Instance Worksheet, draw from the pool of instances an example/non-example set that contains a full range of divergent examples, and a non-example matched to each, with a reasonably balanced distribution of representation forms among the example/non-example types. Select one complete set of test instances as your test.

3. Decide which instances you will test with each of the test formats you selected. Rewrite the instances as test items according to the guidelines for the various test formats.

   (See segment 3.3.2 for guidelines and examples.)

4. For each example and non-example that you adapted for your test, prepare attribute isolation to be used as test feedback.

5. Sequence the test items so that each instance is unmatched and divergent from previous and subsequent items.

3.3.1 Classification-level test formats

Learning activity: List and describe the test formats that may be used for testing classification-level instruction.

Generality:

OBJECTIVE FORMATS FOR TESTING CLASSIFICATION-LEVEL INSTRUCTION:

- o include True/False, Matching, and Multiple Choice items that present to the student an unlabeled instance and ask that he identify response options that indicate the correct classification of the instance.
- o include Completion or Fill-in items that present to the student an unlabeled instance and ask that he supply the concept name that correctly classifies the instance.
- o are easy for the student to respond to and therefore may cover a broad range of concept classes in a single test.
- o require that you write many items for a single test.
- o may be scored quickly, accurately, and consistently.

ESSAY FORMATS FOR TESTING CLASSIFICATION-LEVEL INSTRUCTION:

- o present to the student an unlabeled instance and ask that he classify the instance and justify the classification.
- o generally let you cover only a few concept classes in a single test.
- o favor fluent and articulate students, but they encourage all students to organize their thinking and to express themselves clearly.
- o require that you write only a few instances for a single test.
- o take longer to grade than objective items, and they may be difficult to score accurately and consistently.

3.3.2 Writing classification-level test items

Learning activity: Given instructional specifications for a segment of classification-level instruction, prepare test items in each of the test formats.

Generality:

TRUE/FALSE

Write a True/False item for testing classification-level instruction so that it:

- o Clearly presents all relevant and useful irrelevant attributes that define the instance.
- o presents true/false statements that can be interpreted in only one way.
- o if false, is a likely undergeneralization, overgeneralization, or misconception of the concept class.
- o is easy to read.
- o is consistent with the student objective.

MATCHING

Write a Matching item for testing classification-level instruction so that it:

- o clearly presents all of the relevant and useful irrelevant attributes of the instances.
- o presents response options that can be interpreted in only one way.
- o does not "give away" correct answers, nor eliminate incorrect ones, by the way it is worded.
- o presents all instances in one column and all response options in the other.
- o has more responses than instances, or allows a single response to answer more than one instance.
- o is easy to read.
- o is consistent with the student objective.

MULTIPLE CHOICE

Write a Multiple Choice item for testing classification-level instruction so that it:

- o clearly presents all relevantand useful irrelevant attributes of the instance.
- o presents in the stem a single instance or a single concept class.
- o avoids negatively stated stems that might be misinterpreted if the student misreads the negation.
- o presents response options that can be interpreted in only one way.
- o uses distractors, or incorrect response options, that are likely overgeneralizations, undergeneralizations, or misconceptions of the concept class.
- o uses "All of the above" as a response option only when other combinations of distractors are also presented as possible choices.

COMPLETION OR FILL-IN

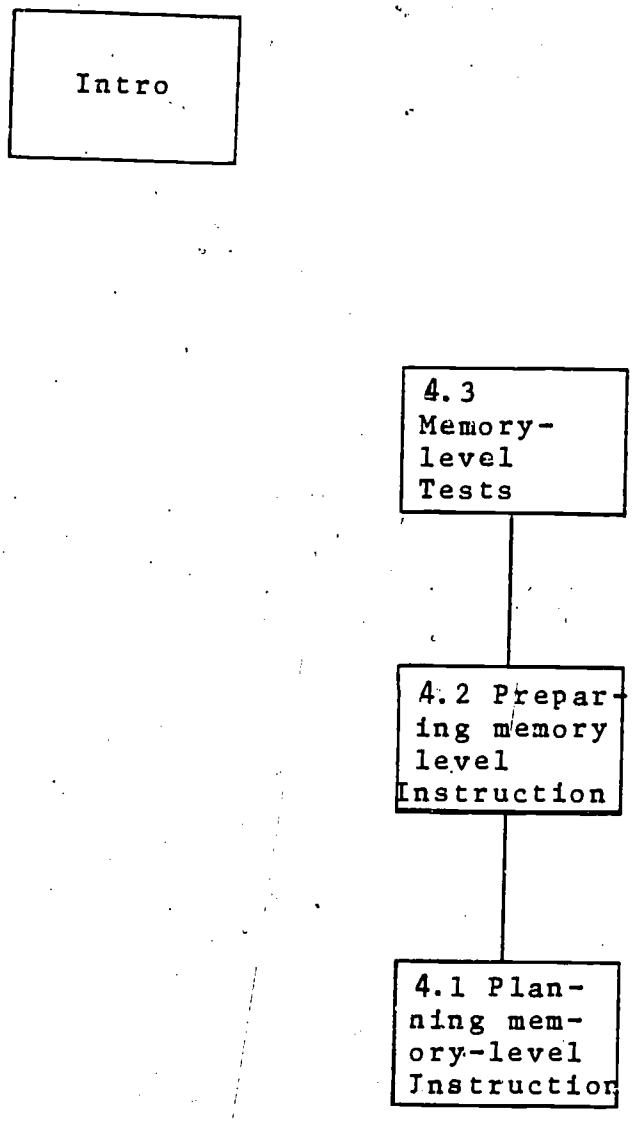
Write a Completion or Fill-in item for testing classification-level instruction so that it:

- o clearly presents all of the relevant and useful irrelevant attributes of the instance.
- o contains enough in the original statement so that the student clearly understands what is being left out.
- o does not "give away" correct answers, nor eliminate incorrect ones, by the way it is worded.
- o is easy to read.
- o is consistent with the student objective.

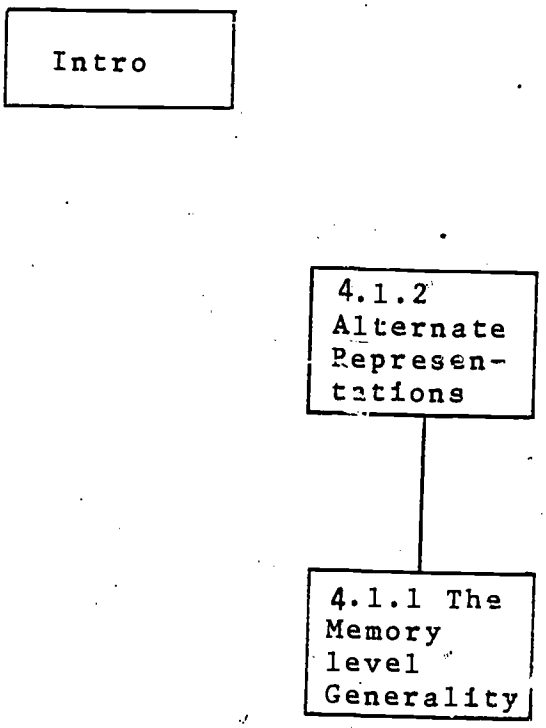
ESSAY

Write an Essay item for testing classifcation-level instruction so that it:

- o clearly presents the relevant and useful irrelevant attributes of the instance.
- o states clearly and unambiguously the classification the student is required to make.
- o tells the student that he must justify his response.
- o is easy to read.
- o is consistent with the student objective.


Intro
4.3 Memory-level Tests
4.2 Preparing memory level Instruction
4.1 Planning memory-level Instruction

Unit 4 Map



Lesson 4.1 Map

4.1. Intro Planning memory-level instruction

Learning activity: Given a topic to be taught at memory level, plan an instructional segment.

Generality:

These guidelines should help you plan instruction for a topic that requires memory-level learning activities.

1. Write on the Generality Worksheet the exact material that the student is to memorize. (See segment 4.1.1 for guidelines and examples.)

2. Determine if there are any alternate ways of representing the information to be memorized. (See segment 4.1.2).

3. If there are alternate methods of representation, decide which ones you will require of the student and list each separately on the Instance Worksheet.

4. Specify instances on the Instance Worksheet by marking one representation for each instance. (See segment 4.1.2 for guidelines and examples.)

4.1.1 Writing the memory-level generality.

Learning activity: Given information to be memorized, write a memory-level generality as you would present it to the students.

Generality:

Write your memory-level generality so that it:

- Clearly sets forth the material to be memorized in the clearest, most straight-forward representation.
- Gives a name for the material to be memorized.
- Is consistent with the student learning activity.
- Contains no nice-to-know material.

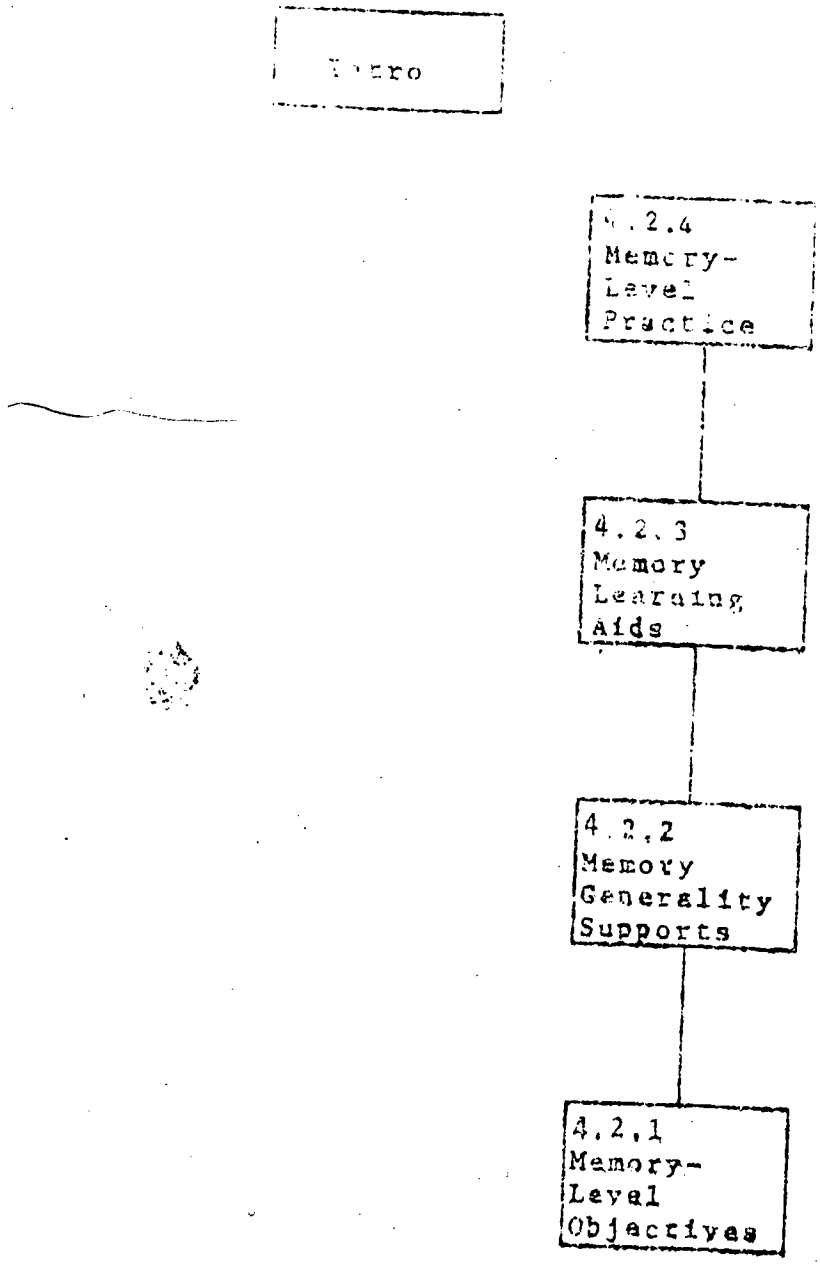
4.1.2 Specifying alternative representations

Learning activity: Given a body of information to be memorized, specify alternate ways in which the information may be represented for instructional purposes.

Generality:

Information to be memorized does not change -- it is fixed or invariant. However, the same information may often be represented in more than one way. When determining alternate representations for a body of information, consider the following:

- Does the information contain numerical, spatial, temporal, sequential, or comparative relationships that might be represented graphically?
- If the information in the generality is graphic or symbolic, may it be expressed in prose?
- You may be able to think of representations not covered by the two questions above. In any case, be sure that the information conveyed in all representations is the same.



Lesson 4.2 Map

4.2 Intro Preparing memory-level instruction

Learning activity: Write a manuscript for a segment of memory-level instruction that contains all necessary instructional components, with the components clearly separated from and consistent with one another.

Generality:

Memory-level instruction generally includes an objective that presents the learning activity to the student, a generality that presents the information the student is to memorize, and a generality support that presents any other interesting or helpful information related to the topic. You may also include memory-level practice.

To prepare memory-level instruction based on your instructional specifications:

1. Write a student objective that will direct the student in his learning activities. (See segment 4.2.1 for guidelines and directions.)

2. Review the generality as you wrote it on the Generality Worksheet to be sure it is consistent with the student objective. If it is not consistent, revise one or the other as needed.

3. Write a generality support that includes memory aids. (See segment 4.2.2 for general guidelines and examples of generality supports. See segment 4.2.3 for guidelines and examples pertaining to memory aids.)

4. Prepare memory-level practice if you judge it appropriate. (See segment 4.2.4 for guidelines and examples.)

4.2.1 Memory-level student objectives

Learning activity: Given specifications for a segment of memory-level instruction, write a memory-level student objective.

With reference to your memory-level instructional specifications, write your student objective so that it:

- o tells the student unambiguously what you expect him to memorize and how he will demonstrate his learning.
- o presents any non-obvious features of the conditions under which he will demonstrate his learning.
- o tells the standards of mastery that you expect him to achieve, including whether recall must be verbatim.

4.2.2 Memory-level generality supports

Learning activity: Given instructional specifications for a segment of memory-level instruction and supplemental information about the topic, write a memory-level generality support.

Generality:

You may use the generality support to present to the student any information you feel will be of interest or of help to him. Generality supports may contain information such as the following:

- Definitions of unfamiliar terms and concepts.
- Information showing how the generality relates to other parts of the instruction.
- A rationale for the generality.
- Interesting nice-to-know information, or other supporting explanation that is related to the generality.
- A mnemonic or chunking memory aid, or a combination of the two. (See segment 4.2.3 for guidelines and examples.)

4.2.3 Memory aids

Learning activity: Given instructional specifications for a segment of memory-level instruction, prepare a mnemonic memory aid, an chunking memory aid, or a combination of the two.

Generality:

A memory aid represents key aspects of the structure of a body of information in a way that is easy to remember. Two of the most common memory aids use (1) mnemonics, and (2) chunking.

MNEMONICS

For content that can be represented as a list, prepare a mnemonic as follows:

1. For each item on the list, select a critical word that can represent to you the entire content of the item.

2. Construct a meaningful sentence, or even several sentences, that incorporate the words that represent the items on the list.

   OR....

   Construct a word that incorporates the first letter of each of the representative words.

3. Make your mnemonic word or sentence vivid through exaggeration.

4. If the order of the items on the list is important, be sure to retain it in your mnmonic.

For content in which two items must be remembered in association with one another, prepare a mnemonic as follows:

1. Decide which of the two items the student already understands and which he is trying to memorize.

2. For each item, generate a word or short phrase that is easily associated with the item through rhyme or in some other easily remembered way, and can be easily visualized.

3. Write a meaningful, but vivid, sentence that associates the two mnemonic words. (If you're writing a series of such mnemonic sentences, be consistent about the order in which you place the item to be memorized (either always before or always after the item the student already understands).

4. Create a vivid mental image of the two items in association: draw it if you can.

4.2.3 Memory aids (page 2)

CHUNKING

1 Chunking involves breaking a list of items to be memorized into smaller groups or "chunks" of items that are associated or categorizable together in some clearly specified way.

2. Chunking may be nested; that is, you may chunk within chunks.

3. You may use mnemonics to remember the items within chunks, or to remember the chunking categories themselves.

4.2.4 Memory-level practice

Learning activity: Given instructional specifications for a segment of memory-level instruction, prepare memory-level practice.

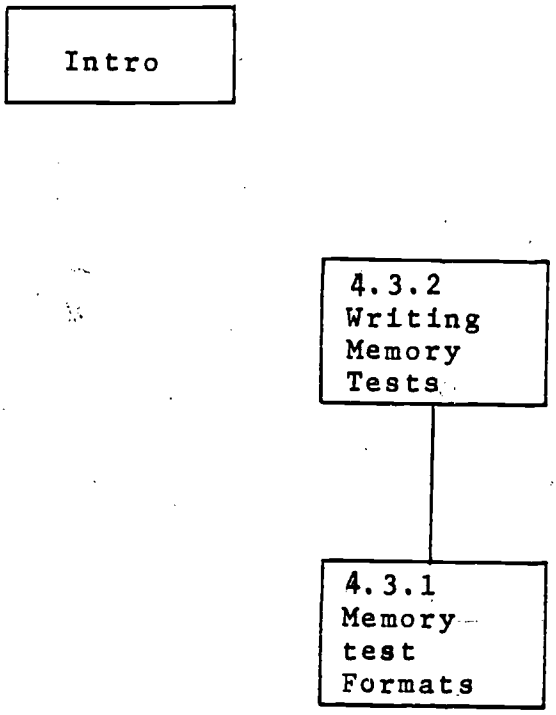
Generality:

Memory level practice may vary widely depending upon the subject matter and your preferences. Here are some suggestions:

1. In some cases, your memory-level practice will be no more than instructions to the student that he drill the material on his own.

2. You may want to suggest some techniques for memorizing .

3. Where the information to be memorized may be represented in more than one way, you may want to direct the student to memorize the information as it is presented in the generality, and then to "practice" what he has learned by either representing the information, or recognizing representations of the information, in alternate representation formats.



Lesson 4.3 Map

4.3 Intro Preparing memory-level tests

Learning activity: Given instructional specifications for memory-level instruction, prepare a memory-level test. .

Generality:

Here are some guidelines for preparing memory-level tests:

1. Decide whether to test the entire body of material memorized or just samples.

2. Select the test format or formats from among the following:

   - o True/False
   - o Matching
   - o Multiple choice
   - o Completion or Fill-in
   - o Short essay

   Do not use more than two or three formats in a single test. (See segment 4.3.1 for descriptions of the test formats.)

3. Prepare test items according to the format you've selected. Check your work. (See segment 4.3.2 for checking the adequacy of test items prepared according to the various formats.)

4. Proofread carefully for typographical errors.

5. No item should contain a clue to any other item.

4.3.1 Memory-level test formats

Learning activity: List and describe the test formats that may be used for testing memory-level instruction.

Generality:

OBJECTIVE FORMATS FOR TESTING MEMORY-LEVEL INSTRUCTION:

- o include True/False, Matching, and Multiple Choice items that ask that the student recognize from memory the correct answer from among a set of options that is presented to him.
- o include Completion or Fill-in items that ask that the student supply from memory an answer that is limited to a single word or phrase.
- o includes Verbatim Recall items that ask that the student reproduce from memory an exact representation of a body of information.
- o are useful when you want to cover a broad area of knowledge in a single test.
- o require that you write many items for a single test.
- o may be scored quickly, accurately, and consistently.

ESSAY FORMATS FOR TESTING MEMORY-LEVEL INSTRUCTION:

- o require that the student reproduce the material specified in the generality in narrative form and in his own words.
- o generally let you cover only a few points in a single test.
- o favor fluent and articulate students, but they encourage all students to organize their thinking and to express themselves clearly.
- o require that you write only a few questions.
- o are time-consuming to grade and difficult to score accurately and consistently.

4.3.2 Writing memory-level test items.

Learning activity: Given instructional specifications for a segment of memory-level instruction, prepare test items in each of the test formats.

Generality:

TRUE/FALSE

Write a True/False item for testing memory-level instruction so that it:

- o presents true/false statements that can be interpreted in only one way.
- o bases truth or falseness on the main idea of the statement, not on trivial details.
- o avoids absolutes such as "never" and "always."
- o avoids highly detailed or highly qualified statements.
- o if false, is a likely misconception.
- o is easy to read.
- o is consistent with the student objective.

MATCHING

Write a Matching item for testing memory-level instruction so that it:

- o states unambiguously what information is required.
- o presents response options that can be interpreted in only one way.
- o does not mix non-parallel items in the individual columns.
- o has more responses than questions, or allows a single response to answer more than one question.
- o does not "give away" correct answers, nor eliminate incorrect ones, by the way it is worded.
- o avoids highly detailed or highly qualified response items.
- o is easy to read.
- o is consistent with the student objective.

4.3.2 Writing memory-level test items. (page 2)

MULTIPLE CHOICE

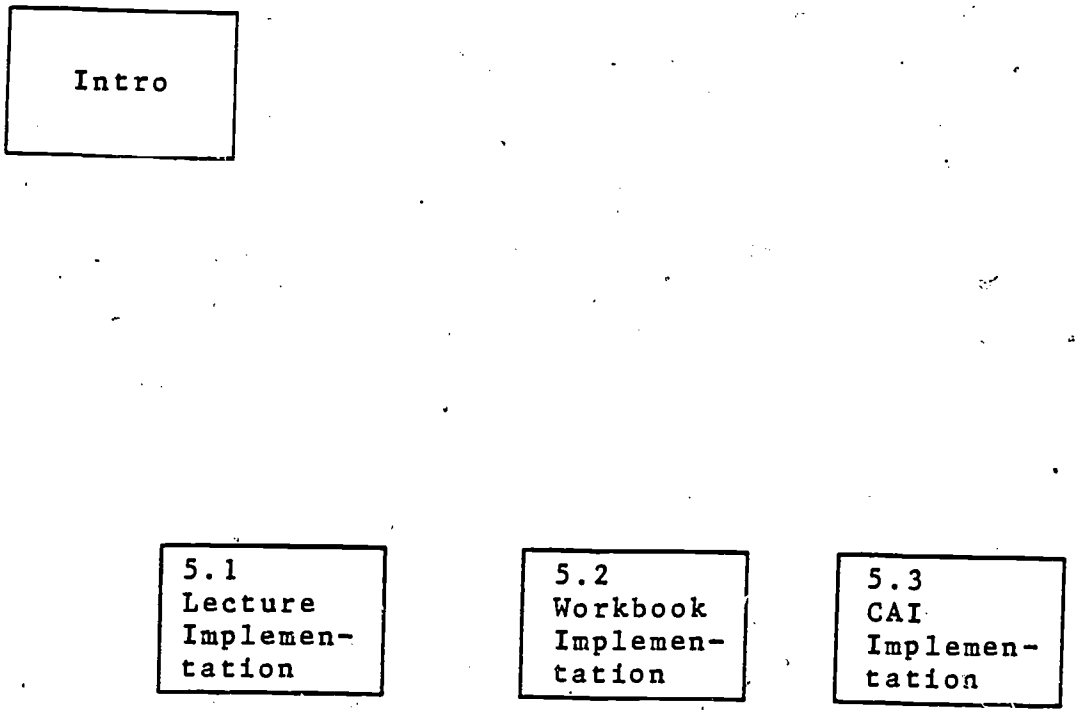
Write a Multiple Choice item for testing memory-level instruction so that it:

- o clearly and unambiguously states a single problem in the stem.
- o presents the stem so that it is grammatically consistent with all of the choices.
- o avoids negatively stated stems that might be misinterpreted if the student misreads the negation.
- o presents response options that may be interpreted in only one way.
- o uses distractors, or incorrect response options, that are likely misconceptions.
- o avoids highly detailed or highly qualified response options.
- o has no choices that are noticeably shorter or longer than others.
- o uses "all of the above" as a response option only when other combinations of distractors are also presented as possible choices.
- o is easy to read.
- o is consistent with the student objective.

ESSAY

Write an Essay item for testing memory-level instruction so that it:

- o states clearly and unambiguously what information the student is required to present.
- o indicates that the student is to express his answer in his own words.
- o specifies any requirements regarding format.
- o is easy to read.
- o is consistent with the student objectives.


Intro
5.1 Lecture Implementation
5.2 Workbook Implementation
5.3 CAI Implementation

Unit 5 Map