DOCUMENT RESUME

ED 289 495 IR 012 962

AUTHOR Ellington, Henry
TITLE How To Design Programmed Learning Materials. Teaching and Learning in Higher Education, 12.
INSTITUTION Scottish Central Institutions Committee for Educational Development.
SPONS AGENCY Robert Gordon's Inst. of Technology, Aberdeen (Scotland).
PUB DATE 87
NOTE 27p.; For a related guide, see IR 012 961.
PUB TYPE Guides - Non-Classroom Use (055) -- Information Analyses (070)

EDRS PRICE MF01/PC02 Plus Postage.
DESCRIPTORS Foreign Countries; Higher Education; *Material Development; *Production Techniques; *Programed Instruction; *Programed Instructional Materials; Teacher Developed Materials
IDENTIFIERS *Printed Materials; Skinner (B F)

ABSTRACT

A sequel to the booklet "How to Produce Printed and Duplicated Materials," this booklet begins by providing an overview of programmed learning. This introduction shows how the field has developed since the work of B. F. Skinner in the 1950s and explains what is generally meant by the term "programmed learning" today. Guidelines for planning and writing textual programmed materials are then presented, which deal in turn with all the various stages in the process: (1) determining the objectives; (2) choosing the content; (3) choosing a suitable programming model; (4) designing the program; and (5) writing the individual frames. Part of a typical programmed text is provided as an example, and a list of four annotated references recommended for further reading is included. (MES)

This booklet was first published internally in Robert Gordon's Institute of Technology, Aberdeen as part of the Institute's staff development programme. It was written by Dr Henry Ellington of RGIT's Educational Technology Unit.

CICED gratefully acknowledges the co-operation of RGIT and the author in the publication of the present edition of the booklet.

How to Design Programmed Learning Materials

Introduction

This booklet is a sequel to booklet number 11 in the series ("How to produce printed and duplicated materials"), which gave general guidance on how to design and produce paper-based instructional materials of all types. The present booklet deals in more detail with the design of one particular class of paper-based materials, namely, self-study materials of the *programmed learning* type.

To put the booklet into context, we will start by taking a general look at programmed learning, showing how the field has developed since the pioneering work of Skinner in the 1950's and explaining what is generally understood by the term 'programmed learning' today. We will then show how to set about planning and writing textual programmed materials, dealing in turn with all the various stages - determining the objectives, choosing the content, choosing a suitable programming model, designing the programme and writing the individual frames. The final section of the booklet will present part of a typical programmed text as an example for readers to study.

What programmed learning is

Since there is often some confusion over what exactly is meant by the term 'programmed learning', we will begin by discussing the early history of programmed learning and showing how the interpretation of the term has evolved over the years.

The origins of programmed learning

Programmed learning had its roots in *behavioural psychology*, the new 'scientific' school of psychology that was pioneered by the American psychologist B F Skinner during the 1950's. Behavioural psychology was based on what is generally referred to as the *stimulus response (S-R)* model of behaviour, i.e. the theory that every thought or action displayed by an organism comes about directly as a result of the organism receiving an external or internal *stimulus* of some sort. According to the early behavioural psychologists, the learning process was no exception, taking place via a linked series of stimulus-response mechanisms, or *S-R bonds*. Thus, they believed that the best way to promote learning was to create conditions that would be conducive to the formation of firmly-established S-R bonds that would cause the learner, when exposed to suitable stimuli, to exhibit the particular behaviour that it was desired that he or she should learn.

One feature of the original stimulus-response model of learning was the concept of *successive reinforcement*, the idea that a desired S-R bond would become firmly established more quickly if the learner was supplied with immediate *feedback* in the form of the correct answer immediately after attempting a particular learning task such as answering a question. Another feature of Skinner's original theory was that each successive stimulus-response step should be small enough to ensure that the learner was nearly always correct in his or her response, thus maximising the effectiveness of the reinforcement process.

Linear programmed learning

The first application of Skinner's research to the classroom situation came in the form of the *linear programmed learning* systems that he developed in the late 1950's. In this type of programmed learning, the subject matter was broken down into a sequence of small *steps* (or *frames*) that followed logically upon one another, with each of the steps representing only a very small part of the concept or skill being taught. In order to reward the learner, and so, in turn, reinforce the learning process, each step contained a certain amount of information and required the student to respond to a question about the information, while the small size of the step practically guaranteed the correctness of the desired response. Immediate feedback on the correctness (or otherwise) of the response was designed to provide suitable reinforcement, and, in principle, it was virtually impossible to take a step without having successfully completed the preceding ones. There was thus only one possible path which a student could take through the frames - hence the name *linear* programmed learning.

Branching programmed learning

In the early 1960's, a new form of programmed learning known as *branching programmed learning* was developed, largely on a result of the work of Norman Crowder. This involved the use of several possible paths through the sequence of frames, with so-called *remedial frames* and *remedial loops* being included in order to correct misconceptions identified from student responses to individual steps, which did not necessarily have to be as small as those advocated by Skinner. Thus, the topic to be studied was taught in a number of alternative ways in such a branching programme, depending on the performance of the learner. This, it was hoped, would avoid the inflexibility and learner boredom that characterised many early programmes of the linear variety. Such branching programmes now form the basis of much of *computer-based learning*, since the modern digital computer constitutes the ideal delivery system for

adaptive self-instructional programmes of this type (see booklet number 9 in this series - "How computers can be used in tertiary education").

The modern interpretation of the term 'programmed learning'

The modern interpretation of the term 'programmed learning' is very much broader and more flexible that that which prevailed during the early phases of the programmed learning movement. Small steps are no longer considered necessary, nor is immediate reinforcement, and, although most programmes are still described as either 'linear' or 'branching' , such programmes can now have a much wider range of structures than was ever the case in the past. Another feature of modern programmed learning systems is the great variety of instructional media and delivery systems that they can involve. Indeed, virtually all the media that are identified in booklet number 10 in this series, "A review of the different types of instructional materials available to teachers and lecturers", are now used in this way. See, for example, the various self-instructional systems described in "How to produce audio materials", "How to produce linked audio and still visual materials", and "How to produce computer-based learning materials", booklets numbers 15, 16 and 18 in the suite).

There is, however, something of a difference of opinion among educational technologists as to what exactly the term 'programmed learning' now means. To some, the term denotes a *process* - a systematic approach to instructional development that is practically identical to the educational technology-based approach that is described in the first booklet in the suite ("Educational objectives"). Others regard this interpretation as being far too wide, and use the term to denote particular types of instructional *systems* or *materials*. According to this interpretation, an instructional system or set of instructional materials should have the term 'programmed learning' applied to it if it has the following characteristics:

(1) it is *self-contained;*

(2) it is designed for use in *individualised learning* situations;

(3) it has well-defined *objectives* which are made clear to the user;

(4) it *structures* the work carried out by the learner;

(5) it allows learners to work *at their own pace*;

(6) it provides the learner with *feedback* on progress and achievement.

It is probably fair to say that the second interpretation is now gaining general acceptance, and is the one that we shall adopt in this booklet.

How to write textual programmes

In "A guide to the use of individualised learning techniques", we discussed the use of textual programmed materials in individualised learning and identified some of the situations in which the use of such materials might be appropriate. As we saw, textual programmed learning materials are best suited for teaching the basic facts and principles of a subject, i.e. for teaching towards objectives of the *lower cognitive* variety. Let us therefore assume that a teacher or lecturer has identified an area of the curriculum which he or she feels could usefully be taught by means of a textual programme of some sort, and look at just how he or she should set about planning and writing a suitable programme.

Finding out whether it is necessary to write your own programme

Writing programmed learning materials is a time-consuming business, so, before starting work on writing a programme for a particular purpose, it is very much in your interest to carry out a *thorough* search of existing materials to see

(a) whether a programme capable of meeting your needs has already been written by someone else,

or

(b) whether an existing programme could be adapted to meet your particular needs.

Such a search can be started by referring to the 1978/79 "International Yearbook of Educational and Instructional Technology" (see 'Further Reading' section), which lists all the instructional programmes that were available at the time it was published (not all that many have been produced since). More up to date information can be obtained from the Programmed Learning Library at Moray House College of Education, Edinburgh, which holds copies of most programmed learning materials that are generally available. If you do find an existing programme that could be used as it stands or adapted to meet your needs, the article by Hodge referred to in the 'Further Reading' section should help you to make effective use of same; this offers guidance on how to choose, use and adapt other people's programmes.

Establishing the objectives for the programme

Assuming that you have *not* succeeded in finding an existing programme that could be used for the purpose you have in mind and have decided to write your own programme, your first task should be to establish a clearly-defined set of *objectives* for this programme. General guidance on how to do this can be found in booklet number 1 in the series, "Educational objectives", but the following specific points should also be borne in mind.

- Do not be over-ambitious, particularly if you have not written a textual programme before; limit yourself to a few simple objectives of the *lower-cognitive type*.
- Write these objectives in *behavioural* form, making them as clear and explicit as possible; if you find that your objectives do not lend themselves to such treatment, they are probably not suited to a programmed learning approach.

Choosing the detailed content for your programme

Once you have established the objectives for your programme, your next step should be to decide on the *detailed content*. This will involve determining the key points that your programme should cover, something that can be done using the "must know", "should know", "could know" approach described on page 5 of "How to produce printed and duplicated materials". This should also help you to establish an embyonic structure for your programme.

Choosing a programming model

One of the key decisions that anyone writing programmed learning materials has to make is the choice of programming model. In other words, should the programme be essentially *linear* in structure, or should it employ some form of *branching*? Let us look at some of the possible structures that are available within these two basic paradigms.

Simple linear programmes

These have the form shown in figure 1, the learner starting at frame 1 and working systematically through all the following frames until the end of the programme is reached.

Figure 1: a basic linear programme

Such programmes are relatively simple to write, but make no allowance for the fact that some learners are capable of mastering material much more quickly than the 'average learner' for whom the programme is probably designed, whereas others are much slower, requiring exposure to far more frames than the average learner before they are capable of mastering a particular set of material.

Linear programmes with 'washahead' and 'washback'

One way of overcoming the above problems is to incorporate '*washahead*' and '*washback*' into a simple linear programme, as shown in figure 2.

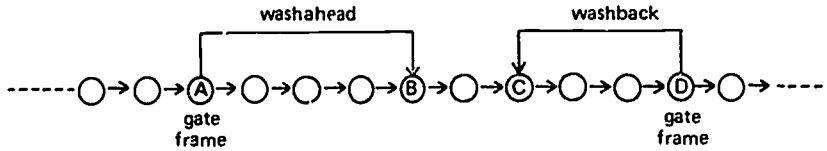


Figure 2: a linear programme incorporating washahead and washback

The process known as 'washahead' enables fast learners to miss out some of the frames in a linear programme, by, for example, moving directly from frame A to frame B in the example shown in figure 2. 'Washback' is the opposite process, and enables slow learners to be made to repeat part of a programme if they have not succeeded in mastering the material. Both processes are controlled by means of so-called *gate frames* – frames that assess the progress that the learner has made and use this assessment to determine his or her subsequent route through the programme. In figure 2, the gate frames are frame A and frame D.

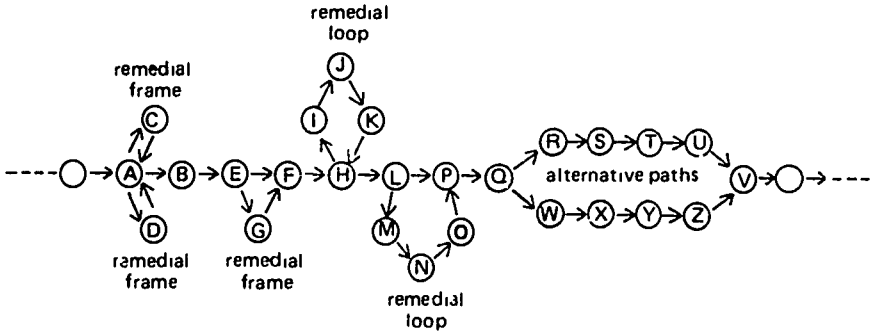
Simple branching programmes

Even when they incorporate washahead and washback, linear programmes make no allowance for the fact that different learners may require to be exposed to *different types of frame* in order to master particular material, since people not only *learn* in different ways, but also make different sorts of *mistakes*. It was this that led Crowder to develop *branching programmes* back in the early 1960's. Some of the features that can be incorporated into a linear programme in order to turn it into a simple branching programme are shown in figure 3.

One of the basic features of most branching programmes is the use of *remedial frames*, one method of using which is illustrated by frames A, B, C and D. Here, the *gate frame* for the group (frame A)

sets the learner a task of some sort (usually a multiple-choice question). If the answer is correct, the learner is simply directed to

Figure 3: some of the basic structural elements that can be built into branching programmes



the next frame of the programme proper (frame B); if the answer is wrong, however, the learner is directed to an appropriate *remedial frame* (either frame C or frame D in this case), and, after being told *why* the answer is wrong, is directed back to the gate frame to try again. Programmes which consist of a sequence of sets of frames of this type are said to have a *herringbone* structure, for obvious reasons (see figure 4).

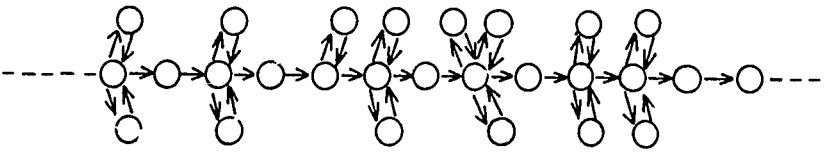


Figure 4: a branching programme with a herringbone structure

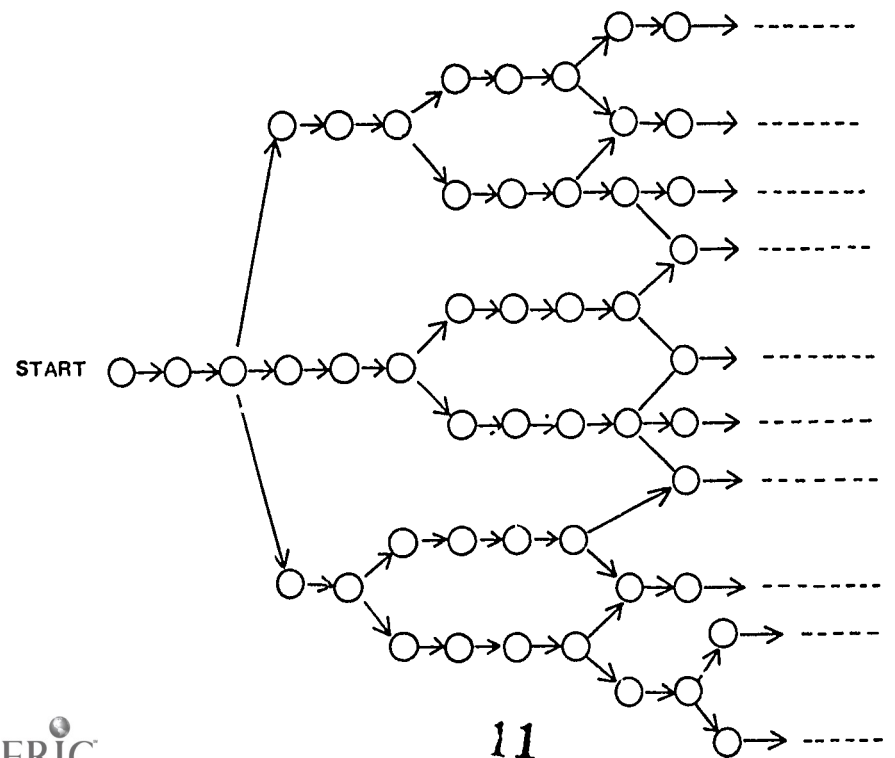
Another method of using remedial frames is illustrated by frames E, F and G. Here, the learner ends up on the same frame (frame F) whether he gets the gate frame question right or wrong. If the former, he moves directly to frame F from the gate frame (frame E); if the latter, he moves to frame F via a remedial frame (frame G) in which he is told why his answer was wrong and given any other remedial information thought necessary. Many programmers now prefer this type of remedial frame structure to the 'herringbone' structure described earlier, since they do not consider it good

practice to double the learner back onto a frame he has already failed to answer correctly.

In some cases, it may be necessary to expose a learner to a *sequence* of remedial frames if he gets a gate frame question wrong. Here, the same two basic approaches that were described above for single remedial frames can be adopted, directing the learner through a *remedial loop* that either leads back to the gate frame just failed (as in HIJKH) or on to the next frame in the programme proper (as in LMNOP).

Another feature that can be built into simple programmes is *alternative paths*, as shown in the sequences QRSTUV and QWXYZV. Here the learner is taught the same basic material in different ways depending on how he performs in the gate frame Q, eventually ending up on frame V in both cases. Use of such alternative paths enables the writer of a programme to make allowance for the fact that people have different learning styles, so that what might be the most effective way of helping one person to master a particular set of material might not be the best way for another.

Figure 5: the start of a typical fully-adaptive branching programme



Fully-adaptive branching programmes

The branching programmes described so far are still basically linear in structure in that all learners take more or less the same general path through the programme and all eventually end up in the same place. It is, however, possible to design much more complicated branching programmes in which the *overall direction* along which the learner is led is determined by how he or she performs. Needless to say, fully adaptive programmes of this type can become very complicated very quickly, as figure 5 clearly shows. For this reason, the fully adaptive structure is much more suitable for use in computer-based learning than in textual programmes.

Designing the programme

Once you have decided which programming style you want to use, it is necessary to draw up an *overall design* for your programme. This is best done using a flow diagram (or hierarchy of flow diagrams) that shows the overall structure of the programme and the individual frames of which it is to be composed.

The form that your programme will take will obviously depend on the nature of the material being covered and the programming style that you have decided to adopt. In the case of a simple linear programme designed to introduce the learner to a new concept (or related group of concepts) and help him to become competent in its (their) application, for example, a six-stage structure of the following type might well be adopted.

(1) An introductory section that explains the purpose of the programme.

(2) A review of any previously-studied material or other material with which the learner should be familiar before tackling the main section of the programme.

(3) A step-by-step presentation of the new material, together with appropriate illustrations.

(4) A 'consolidation' section in which the learner is given practice in applying the new material in unfamiliar situations.

(5) A summary of the main points covered.

(6) A diagnostic test to assess the extent to which the learner has mastered the material.

Such a structure can also be used for sub-sections within a larger programme.

The different types of frame that can be used in a programme

With regard to the basic 'building bricks' using which a programme is built – the individual *frames* – these can take a wide range of forms. The four most important types are listed below.

Teaching frames – frames whose main function is to present the learner with new information or knowledge, help him to re-structure knowledge he already possesses in some new way, review material previously covered, and so on; two of the most common types are *ruleg* frames and *egrule* frames (terms whose meaning will be explained in the next section).

Practice frames – frames whose main function is to help the learner to master newly-presented material by having him carry out simple exercises based on same.

Test frames (or *criterion frames*) – frames whose main function is to *test* the learner in some way, not merely provide him with practice; such frames are used both to provide the learner with feedback on his progress and to determine his subsequent path through the programme, test frames of the latter type being known as *gate frames*.

Response frames – frames which follow on from a test frame in a branching programme; such frames are of two types, namely *'right answer' response frames* (which confirm and reinforce a correct response before leading the learner on to the next stage of the work) and *'remedial' response frames* (which provide remedial material following an incorrect or otherwise unsatisfactory response).

Examples of all these different types of frame will be given in the next section.

Writing the frames

Once you have decided on the overall structure of your programme, it is obviously necessary to start writing the actual frames. In an introductory set of notes of this type, it is not possible to give detailed guidance on how to set about doing this (whole books have been written on the subject!) We will therefore merely offer some general hints on how to write frames of the four basic types listed above and look at illustrative examples of each.

Writing teaching frames

Here, the most obvious requirement is that the frame should present the material being covered in a clear, direct and unambiguous manner, using language that the learner will be able to understand without difficulty. The general guidelines on writing instructional

materials given on pages 4-8 of "How to produce printed and duplicated materials" should be followed, especially Rowntree's 'twelve hints for effective writing', which are sufficiently important to warrant repeating here:

1. Write like you talk.
2. Use the first person.
3. Use contradictions.
4. Talk directly to the reader.
5. Write about people, things and facts.
6. Use active verbs and personal subjects.
7. Use verbs rather than nouns and adjectives.
8. Use short sentences.
9. Use short paragraphs.
10. Use rhetorical questions.
11. Dramatise wherever possible.
12. Use illustrations, examples, case studies.

Let us now look at two of the most important types of teaching frames, *ruleg* frames and *egrule* frames.

Ruleg frames are so called because they start by stating a general rule or principle (the *rule* part of the frame) and then illustrate this by giving a specific example or set of examples (the *eg* part of the frame). A typical ruleg frame is shown below.

> To cube a number, you multiply it by itself twice.
> The cube of 2 is 2 x 2 x 2 = 8
> The cube of 3 is 3 x 3 x 3 = 27
> What is the cube of 4?

Note that the last example given in the frame has to be completed by the learner – a standard technique in programmed learning. The correct answer (64) would be given at the start of the next frame in the programme, thus providing the learner with immediate feedback and reinforcement. The ruleg principle can, of course, be used in *sequences* of frames within a programme as well as in individual frames.

Egrule frames (or *inductive* frames) are so called because they start by giving specific examples or instances of a rule or principle in a tion (the *eg* part of the frame) and then ask the learner to try to

work out what the rule or principle is by a process of induction (the *rule* part of the frame). A typical egrule frame is shown below

> When a uranium 238 nucleus (mass number 238, atomic number 92) undergoes alpha (α) decay, it changes into a thorium 234 nucleus (mass number 234, atomic number 90).
>
> When a radium 226 nucleus ($^{226}_{88}Ra$) undergoes α decay, it changes into a radon 222 nucleus ($^{222}_{86}Rn$).
>
> From these two examples, can you suggest a general rule for what happens to the *mass number* A and the *atomic number* Z of a nucleus that undergoes α decay?

As with the ruleg frame given earlier, the correct answer would be given in the next frame in the series, which would probably then reinforce and consolidate the material by providing further examples, some of which would have to be completed by the learner. Such a follow-up frame might take the following form:

> You should have seen from the examples given in the previous frame that the *mass number* A of a nucleus *falls* by *four units* during alpha decay and that the *atomic number* Z *falls* by *two units*. This, of course, is a direct consequence of the fact that the particle emitted during α decay (the α *particle*) contains *two protons* and *two neutrons*, and so has a mass number of 4 and an atomic number of 2. As we have seen, it is in fact a helium 4 nucleus ($^{4}_{2}He$).
>
> Thus, when plutonium 239 ($^{239}_{94}Pu$) undergoes decay, it changes into uranium 235 ($^{235}_{92}U$).
>
> Can you now use the above rule to work out the mass numbers and atomic numbers of the products of the following α decay processes and hence complete the right-hand-side symbol in each case?
>
> $^{232}_{90}Th \xrightarrow{\alpha \text{ decay}} {}^{?}_{?}Ra$
>
> $^{227}_{89}Ac \xrightarrow{\alpha \text{ decay}} {}^{?}_{?}Fr$
>
> $^{222}_{86}Rn \xrightarrow{\alpha \text{ decay}} {}^{?}_{?}Po$

Such a frame acts both as a *teaching frame* and as a *practice frame*, another common technique used in programmed learning. The correct answers ($^{228}_{88}Ra$), ($^{223}_{87}Fr$) and ($^{218}_{84}Po$)) would again be given at the start of the next frame in the programme.

Like the rule6 principle, the egrule principle can, of course, be used in *sequences* of frames within a programme as well as in individual frames.

Writing practice frames

The main object of incorporating practice frames in a programme is to help the learner to master a principle, procedure, etc. after this has been introduced or explained in earlier teaching frames. One example of such a frame – the follow-up frame to the egrule frame on α decay – has already been given.

Two techniques that are commonly used in writing practice frames are *prompting* and *fading*.

Prompting involves giving the learner clues of some sort in order to help ensure that he gives the correct response. There are two main types of prompt – *formal prompts* and *thematic prompts*.

Formal prompts help the learner by providing information about the *form* of the required response, as in

"The capital city of France is P _ _ _ _"

where the learner is 'told' that the answer is a five letter word that starts with P.

Another common type is the so-called *syntactic prompt*, where the clue is provided via the learner's (presumed) knowledge of grammar or syntax, as in

"The home of an eskimo is called an _____"

where the use of 'an' preceding the required response indicates that it begins with a vowel.

Thematic prompts, on the other hand, help the learner by providing or suggesting meaningful associations related to the meaning or *theme* of the required response. One common form is the *semantic prompt*, which is based on the roots or meanings of the words used, as in

"Conifers get their name from the _____-shaped seed cases that such trees produce"

Another is the *temporal prompt*, which makes use of information presented in the previous (or a recent) frame to help the learner give the required response. Thematic prompts can also be provided through graphic information, as (for example) when the learner is required to complete a drawing, diagram, etc.

It is recommended that the majority of prompts in a programme should be of the thematic rather than the formal type, since they are generally rather more effective as aids to learning.

Fading involves gradually removing the prompts in a sequence of practice frames in order to place a progressively greater onus on the learner. In a faded sequence of frames, the first frames are generally fairly strongly prompted in order to make it practically certain that the learner will give the correct response, while the final frames are generally completely unprompted, as are any *test frames* on the material that are included in the programme. (Indeed, test frames should *never* be prompted, for obvious reasons.)

Writing test frames and response frames

As we have seen, *test frames* are frames whose main purpose is to *assess* the learner rather than help him master material – either in order to provide him with feedback on the progress he is making or the extent to which he has succeeded in mastering the work of the programme, or in order to determine the path that he should subsequently take through the programme. In his book "Basically Branching" (see 'Further Reading' section), Derek Rowntree again gives valuable guidance on how to write such frames:

1. Test only the *crucial* point you've made in each frame.
2. Test one point at a time.
3. Make the student *use* the information you've given him.
4. Never ask questions that simply demand parrot-repetition.
5. Never force the student to guess.
6. Never write catch questions.
7. Make the student *do*, rather than talk about doing.
8. Ask your questions as directly as possible.
9. Don't use questions with only two real alternative answers.
10. Let your alternatives cover all reasonable sources of error.
11. Make sure your alternatives are all plausible.

A typical example of a test frame that acts as a *gate frame* in a branching programme is given below.

Frame 20

In the previous frames, we saw that the symbol b^n means "the number obtained by using the number b as a factor n times", so that 2^3 (for example) is equal to $2 \times 2 \times 2 = 8$.

We also saw that, in an expression of the form b^n, the number *b* is called the *base* and the number *n* is called the *exponent*.

Now here is a question on this material.

"If the base of an expression is 5 and the exponent is 2, what is the expression equal to?"

Pick an answer from the list given below, and move to the frame indicated.

Answers:
"25" (move to frame 21)
"32" (move to frame 22)
"I don't know" (move to frame 23)

As can be seen, this particular frame serves as a *review frame* (by recapitulating and summarising material previously covered), as a *test frame* (by assessing the extent to which the learner has mastered this material) and as a *gate frame* (by directing the learner to one of the three *response frames*, depending on how he performs in the test). Of these three response frames, the first (frame 21) would be the so-called *'right-answer' response frame*, while the others (frames 22 and 23) would be *'remedial' response frames*. The form that each of these might take in a sequence of the herringbone type is shown below.

The 'right answer' response frame

Frame 21

Your answer was: If the base of an expression is 5 and the exponent is 2, then the expression is equal to 25. You are CORRECT, since $5^2 = 5 \times 5 = 25$.

Well done!

Let us therefore move on to the next stage of this review.

(and so on)

The 'remedial' response frame designed to correct the specific misunderstanding that led to the incorrect response '32'

Frame 22

Your answer was: If the base of an expression is 5 and the exponent is 2, then the expression is equal to 32.

This is not correct. You have in fact got the terms *base* and *exponent* mixed up, using the number 2 as the base of your expression and the number 5 as the exponent, giving an answer equal to 2 x 2 x 2 x 2 x 2 = 32.

What you *should* have done is taken the number *5* as the base and the number 2 as the exponent. Return to frame 20 and try the question again.

The 'remedial' response frame designed to explain the material more clearly to a learner who has indicated that he has not understood it first time round by answering "I don't know".

Frame 23

Your answer was "I don't know", so we obviously failed to explain the material on which the question was based clearly enough for you to understand it properly. Let's see if we can do better now.

Some of the numbers in our number system are the product of *factors*. The number 15, for example, is the product of the numbers 5 and 3, since 5 x 3 = 15. We say that 5 and 3 are *factors* of 15.

Now there are some numbers that are the product of the *same* factor used a certain number of times. The number 9, for example, is the product of the number 3 used as a factor twice, i.e. 9 = 3 x 3. A number that is the product of the *same* factor used a certain number of times can be written in the form b^n, where b is called the *base* and represents the number that is used as the factor, and *n* is called the *exponent* and indicates the number of times the base should be used as a factor. The number 9, for example, can be written as 3^2, which simply means '3 used as a factor twice', or '3 x 3'. In this case, 3 is the *base* and 2 is the *exponent*. Now in the original question in frame 20, you were told that 5 is the *base* and 2 is the *exponent*. Go back to frame 20 and try the question again.

Some extremely useful general guidance on how to write 'right answer' and 'remedial' response frames of the type shown above is again given by Derek Rowntree in his book "Basically Branching".

Guidelines on how to write 'right answer' response frames

1. Repeat the student's answer with sufficient of the question to make it a complete meaningful statement.
2. Tell him he is correct.
3. If you think he's really earned it, reward him with a phrase like 'Very good'.
4. Tell him, briefly, why his answer is correct.
5. Lead smoothly into your new information, making clear to the student how it links up with what has gone before.
6. Make your new point simply and directly; be conversational, shun pedantry, and never forget that someone's got to read every word you write.
7. Scrutinise the frame for ambiguities and obscurities. Get rid of them.
8. Write a question that shows whether the student can *use* the new idea you've presented him with. (Never write questions testing ideas that are yet to come.)
9. Write sufficient alternative answers to cover all the logical sources of error.
10. Feel free to vary the form of questioning to suit your own purpose.

Guidelines on how to write 'remedial' response frames.

1. Repeat the student's answer, with sufficient of the question to make it a complete meaningful statement.
2. Tell the student immediately that his answer is incorrect, incomplete, unwise, too vague, or whatever.
3. Assume that the student has reached this frame because of your failure to communicate. Take full responsibility for correcting his error.
4. If possible, tell the student what faulty procedure led to his answer, and why it was wrong. Do this without the aid of sarcasm, impatience, or condescension.
5. Bring in new examples, more down-to-earth illustrations, more clear-cut cases: do whatever you must to help your student see the right answer. Don't tell him the right answer, but arrange things so that he now must see it himself.
6. Use sub-sequences whenever necessary, but don't let them much exceed ten right-answer frames in length.

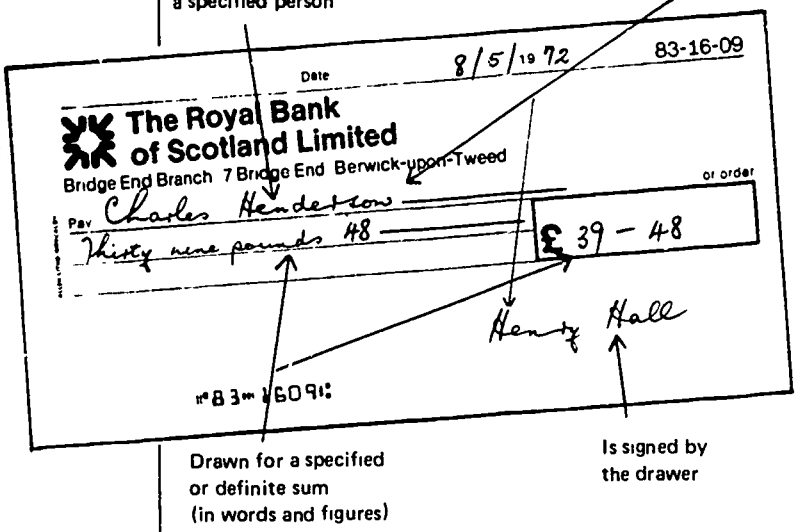
7. Make sure that your answer-alternatives leading into a sub-sequence are truly diagnostic of the student's need for it. Allow several points of entry: perhaps from two or three successive questions.
8. Finish every frame by telling the student how to reach the one he needs to see next.

An example of a programmed text

Readers who are seriously interested in writing programmed learning materials will find much more detailed guidance on how to set about this in the booklet by Megarry and the book by Rowntree that are listed in the 'Further Reading' section. In the meantime, we will end this booklet by giving an example of an actual programmed text – the first 11 frames of a linear programme that is used by the Royal Bank of Scotland to train staff how to deal with cheques when they come across them while working at the cash counter. The programme is presented in the form of a landscape A5 booklet held together by a spiral binder, with each frame occupying a single page and only one side of each sheet being printed on (except where supplementary material is presented along with a frame, as in the case of frame 3). Thus, the booklet takes the form of a 'mini-flip chart' – making it extremely easy to use. Detailed study of this material should help illustrate many of the general points made in these notes.

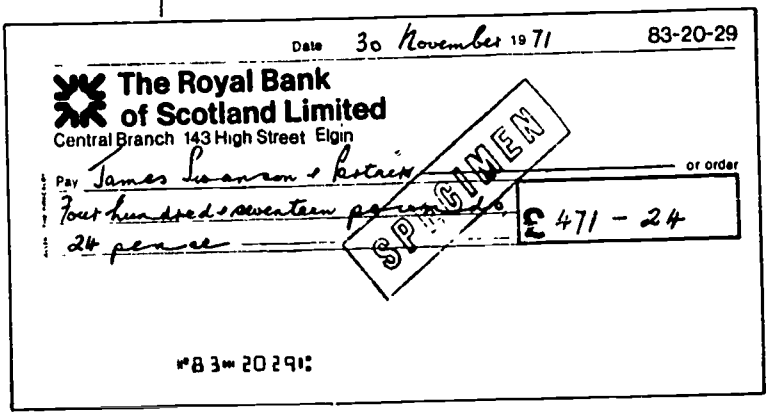
The start of a typical programmed text – the Royal Bank of Scotland's programme on 'Cheques'

First Introductory page	**To the student** This book has been designed as a self-instructional programme in order that you may learn the subject matter at your own pace, without any pressure from a lecturer or otherwise. You should work through the material at a time best suited to yourself and the branch schedule. Each page of the programme is termed a *frame*. In some of the frames you are asked a question, at which point you are required to write your answer on an answer sheet provided for this purpose. You then turn over to the next frame and find the response given at the top. This instant knowledge of your results is intended to interest and motivate you towards the material which follows. But you may ask "What if I give the wrong answer?" Well, we have found that this usually does not discourage students. Anyway, the programme is so geared that you will get the correct answer most of the time. If you "cheat" and look before writing your answer, you may still learn the material, but not nearly as well as when you honestly attempt an answer. At the termination of the programme, you are required to do the post-test. This is only to show your accountant and yourself how you fared with the material. If you have found difficulty with any of the material, it may prove worthwhile to re-read the text a week or so later.
Second Introductory Page	As this programme is rather heavy, and longer than the others in the package, it could, with some students, be split, i.e. you may find it worthwhile to have a break in the middle at some point foran hour or two. This is left to yourself to decide. In time, most of the cheque examples used will be out-of-date. Unless told to the contrary, you should assume that you are doing the programme in late May, 1972. The words *cancelled* and *specimen* appearing on any cheques shold be ignored. Now carry on and read the text. Writer – B H Latto
Third Introductory Page	**Cheques** **Objective:** – Once the programme is completed, a person about to go onto the cash as teller will be able to look for and identify any irregularities in cheques and be able to cash cheques with confidence.

Frame 1	**Cheques** Prior to your going onto cash, you will undoubtedly have dealt with innumerable cheques in the bank, whether this has been on the ledger or waste desk. On the telling table, you will come to deal more closely with cheques. Frequently, you will be giving out cash against them, so you will require to know exactly what a cheque is and be able to analyse it for discrepancies. As you may know, a cheque is primarily an order addressed by one person to another, his banker, instructing that a certain sum of money be paid or transferred to a third party.
Frame 2	As teller, you are in an important position regarding cheques. You are not just listing them or dealing with them in an impersonal manner. You are giving away the bank's cash against them. The bank may lose money if you give out cash indiscriminately. You therefore must be sure that the cheque is alright. You must know what you are about in this new position you are about to hold.
Page facing Frame 3	

Frame 3	There are various essentials which should be included in a cheque. Opposite, on a specimen cheque, four of these essentials are highlighted. These are as follows: 1. It should be in writing 2. It must be signed by the drawer 3. It must be drawn for a specified or certain sum of money 4. It is payable to the order of a specified person (or to the bearer). Now look opposite and study the cheque. What other point do you notice about it that is not mentioned?
Frame 4	It is dated – we shall discuss the date more fully later. Now let us take a look at some of these essentials one at a time and see what they mean to us.
Frame 5	1. It should be in writing. This means that the cheque may be made out in *any form* of writing, e.g. it may be in ink, typed, printed or handwritten. Do you think that one drawn in pencil would be acceptable?
Frame 6	Yes One drawn in pencil *would* be in order as it would still be in writing. Due to the ease, however, with which such cheques could be altered by an unauthorised person, the banks *discourage* customers from drawing them in *pencil*.
Frame 7	2. It must be signed by the drawer. This is straightforward. The cheque is not the order of the customer until it is signed by him or by someone duly authorised to do so on his behalf.
Frame 8	3. It must be drawn for a specified or certain sum in money. The sum in a cheque must be stated with certainty. You could not have a cheque, "pay John Brown £15 or £20". The sum to be paid in this case is not certain, i.e. it is not definite. The amount should be in writing and in figures, and the

	two should agree. If they differ, the bank cannot ascertain which amount is to be paid, and in such cases, the cheque should not be paid until the discrepancy is put right by the drawer. The amount should thus be ascertainable with certainty.
Frame 9	4. It is payable to the order of a specified person or to bearer. If you were presented with a cheque for cashing, the payee (i.e. the person or parties to whom the sum is to be paid) must be named specifically, and one must be in no doubt as to whom the money is to be paid. Alternatively, the cheque may be drawn payable to "bearer" (although this practice is rare today) which indicates that the bearer or holder of the cheque may be given the cash. As you can see, such a practice could be dangerous, as a thief could quite easily cash the cheque if he were the holder.
Frame 10	Now, with the specimen overleaf, examine it and then write down the essentials which are included in it. If there is any discrepancy in the cheque or any essential missing, state what.
Specimen (on separate page)	



Frame 11	1. It is in writing 2. It is payable to a specified person. 3. It is also dated. Essentials Wanting 1. It is not for a certain sum, as the amounts differ (£ in figures say 471; words say 417). 2. It is not signed. --- It is permissible for a customer to draw a cheque in favour of a fictitious person, e.g. "Pay Oliver Twist or order". In cases like these, the cheque immediately falls to be payable to the bearer, whoever he may be. In every case, the teller may find it difficult to decide whether or not the payee is a fictitious person. If in doubt, refer it to the accountant. In practice, one *rarely* encounters such cheques. (and so on)

Further Reading

1. *Programmed Learning - Writing a Programme*, by I Megarry; Jordanhill College of Education, Glasgow; 1978 (An inexpensive booklet that deals with all the material covered in this booklet in much greater detail, and also covers topics such as pilot testing and editing; highly recommended).

2. *Basically Branching*, by D Rowntree; Macdonald; 1966. (One of the most useful books ever written on programmed learning; an invaluable source to anyone seriously thinking about writing such materials).

3. *A guide for users of programmed material*, by P Hodge. In *Strategies for Programmed Instruction*, by J Hartley; Butterworths; 1972. (An extremely useful article that offers guidance on how to use and adapt other people's programmes).

4. *International Yearbook of Educational and Instructional Technology 1978/79*, edited by A Howe and A J Romiszowski; Kogan Page, London; 1978 (This contains a comprehensive catalogue of the programmed materials that were generally available in the UK at the time of publication; an invaluable source of reference to would-be users of such materials.)